A Conceptual Model Driven Semantic Web – an Integrated Binary Approach [1]

(2004-09-15)

Stig Berild

(Santa Anna IT Research Institute AB)

[1] Rapport framtagen i samarbete mellan e-Society II-projektet (www.skriver.nu/esociety) vid Santa Anna IT Research Institute (www.santaanna.se) och Serviam-projektet (www.serviam.se) samt med stöd från Stiftelsen SISU och Vinnova.

Abstract

In general, approaches to Service Oriented Architecture (SOA) do not seem to pay enough attention to semantics when it comes to exchanged information. This is also true for exposed information about services (and related entities) to be explored and discovered in the web environment. This paper discusses a web architecture based on a binary modelling language and an extensive use of conceptual models. As can be expected, semantic expressiveness, ease of use and flexibility are some of the more noticeable characteristics. Furthermore, the architecture has been focused on creating a well integrated solution. Noticeable is also the very limited role played by XML. Interestingly enough, some of the supporting facilities were specified and implemented some thirty years ago.

1 Introduction

1.1 Background, history

Thirty five years ago a small group of enthusiasts started a research project at the Royal Institute of Technology, Stockholm under the leadership of Professor Janis Bubenko jr. The project was named CADIS (Computer Aided Design of Information Systems) to indicate what we intended to do – to develop a prototype tool to support the design of new applications. The prototype should be capable of storing the specification of a design during its different stages of refinement. Those types of tools were later on called CASE tools. It was early on discovered that those specifications needed a database solution capable of storing complex data structures. Furthermore, experimentation required extreme flexibility in data structuring. This required a powerful but at the same time easily managed structuring language. Also, schema support should just act as an optional feature.

No such dbms was around at the time. Inspired by the work of Langefors [1] and his concept of “elementary messages”, a new dbms was developed and implemented based on a binary relationship approach. The data elements in the database were stored in the form of triplets or basic statements (<subject>, <predicate>, <object>). The use of triplets is certainly not unique today – but it was at the time. As it turned out during the following years, the research project came to put their main efforts in developing this database solution and related features (query language, integrated programming language, multi user environment, transaction management, …) rather than emphasizing computer aided solutions. Results were presented at international conferences, e.g. [2] as well as in a published book [3].

The different versions of the dbms were called Cadis System 1 – 5 or shorter CS1 – CS5. One version of the system was later on commercialized under the name of DREAM in the 1980s. A number of applications, mainly operating complex data structures, came to be implemented with DREAM as the supporting database management system. However, according to the according to the commonly held view at the time databases should be relational. Because of that the development efforts had to be reduced after a while. Most resources were instead oriented towards supporting existing applications. By the way, some of them are still in active use today (e.g. some defence applications and auction house applications).

Binary modelling languages had their main period in the 1970s. The CADIS project, Abrial [4], Senko [5] and many others were all involved. With Chen [6] came an interest shift towards different types of Entity-Relationship Modelling Languages. Later on Object Databases in some way revived an interest in Binary Languages. However, only recently things changed dramatically. With the web came the need for metadata management and once again the need for flexible and complex data structures. Binary modelling was redeveloped in Resource Description Framework (RDF). RDF helped open up a new phase in modelling and in information management, not just for metadata but for data in general. Data wasn't any longer required to be managed in relational databases. The Binary Modelling Language and supporting technologies experienced a well deserved revival also as an alternative for structuring databases.

No longer is information just stored in databases. It is moving around on the web. As the next chapter will explain, information exchange may also – and preferably – be based on binary models. Information of different kinds - be it data, metadata, model data, or anything else – constitute the base ingredients that gives the web its purpose, meaning and soul. It is about knowledge sharing and cooperation in its most general sense.

Along came Service Oriented Architectures (SOA) with Web Services Architecture as the most well known alternative. It was time to combine binary modelling and modern advancements in web architectures into a fully integrated approach, the Integrated Binary Architecture (IBA).

1.2 Purpose

This report proposes a new, simple yet powerful web architecture for storage, discovery, interpretation, and exchange of information based on an extensive use of Conceptual Models. It is called the *Integrated Binary Architecture (IBA)* where 'Binary' is intended to associate to the Binary Modelling Language defined within the CADIS project (see above). The language is used to specify Conceptual Models and to operate on binary data. The focus on conceptual modelling opens up new perspectives on data, metadata, and model data. It is shown how these, formerly often presented as quite different types of data, in fact may be approached and modelled using the same modelling technique and the same modelling language.

Simplicity has been shown to be one of the major drivers for success on the Internet. Simplicity is also believed to be an important property of the new architecture in order for it to gain widespread acknowledgment and use.

IBA is a general-purpose, integrated architecture built on solid ground. It furthermore establishes a flexible platform for future evolution steps toward more visionary perspectives such as an anticipated web supported eco society.

2 An Overview of the Integrated Binary Architecture (IBA)

2.1 The Binary Modelling Language

Suppose there is an interest in remembering or sharing some information about something real or imaginary for some purpose. This "something of interest" is often called the *Scope.* The scope in combination with a given Context, as seen by those having the information of interest, is often called the *Universe of Discourse – UoD*. The information about this UoD has to be expressed in some way according to some restricted language (in order for it to be manageable).

A number of different modelling languages are available to define Conceptual Models, each with its unique capabilities. For IBA a natural choice is the *Binary Modelling Language (BML)* defined by the CADIS project (see above). It exposes a combination of expressiveness and simplicity and is as such a perfect candidate for use on the web. The revival of binary relationships in RDF and its promotion by W3C establishes a solid ground for BML.

The basic construct for expressing facts or opinions about a UoD is the *Basic Statement* in which two entities are related by a binary relationship. "Smith lives in Idaho" is an example of a Basic Statement. 'Smith' and 'Idaho' are the entities (often referred to as subject entity and object entity). 'Lives in' is the relationship (often referred to as the predicate). A typical graphical notation is shown in figure 1.

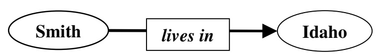


Fig. 1. A Basic Statement

When we talk about or exchange information about some subject (UoD) we usually need a number of Basic Statements to express all we want. These Basic Statements are related in different ways as they are all about the same UoD. See figure 2.

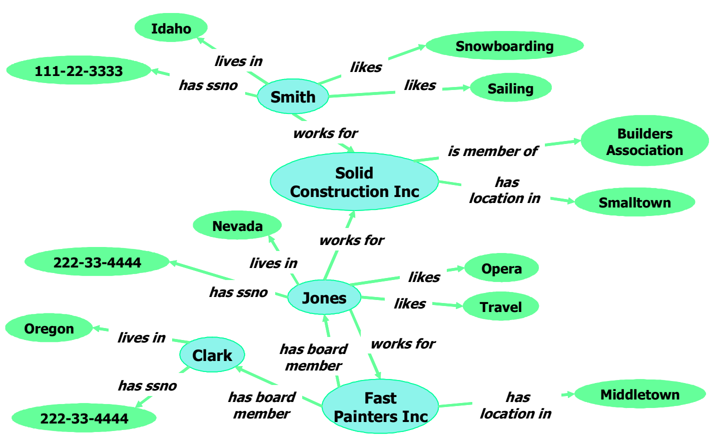


Fig. 2. A number of Basic Statements

An alternative way of expressing the same thing is by distinguishing between concepts and terms/symbols. Concepts are represented by an internally generated symbol with no meaning other as a common point of reference. In the 1970s they were usually called "surrogates" or "internals". Relationships are given more neutral names just to show another way of expressing Basic Statements. See figure 3.

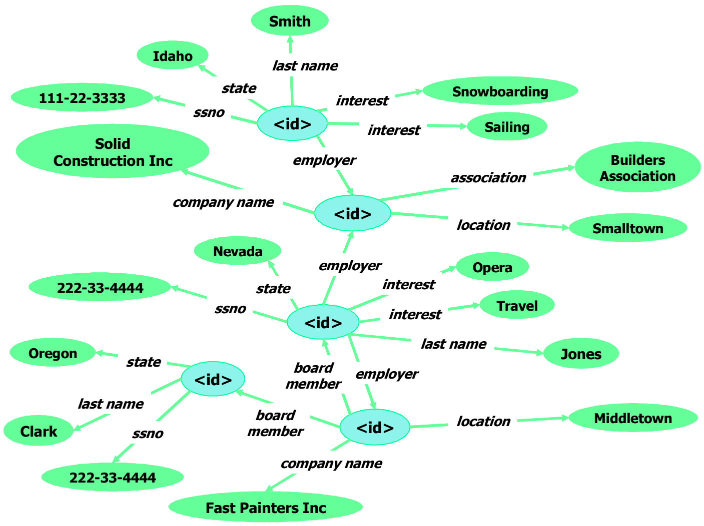
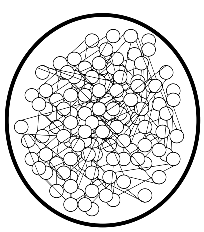


Fig. 3. The same content expressed in an alternative way

2.2 Information Clusters and Information Cluster Management

In most real cases the number of Basic Statements may be thousands or millions. For reasons, that soon will be explained, this set of statements is called an *Information Cluster (IC)* rather than a database. In the following, figure 4 a will be used to symbolize an IC.

In IBA an IC may be operated on via a *Binary Interface (BI),* three interface languages that take advantage of the capabilities the modelling language has to offer.



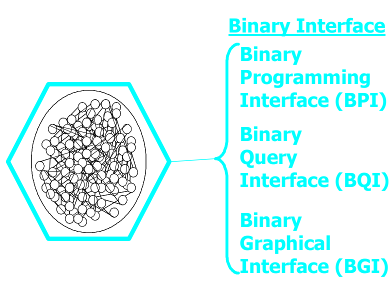


Fig. 4 a. Information Cluster **Fig. 4 b.** Binary Interface alternatives

A specific Retrieval is in fact a set of Basic Statements either forming a sequential stream or an IC in its own right, depending on the requester's intended use of the information.

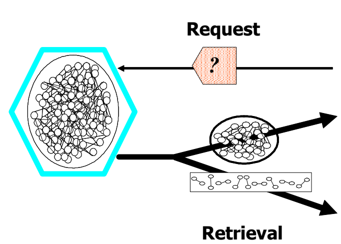


Fig. 5. Alternative output formats

No matter the alternative, the structure (complex or simple) is implicitly part of the set of retrieved Basic Statements in the same way as they form a structure in the originating IC. There is no need for transformation to any hierarchical structure (read: XML) as long as the receiver of the information is acting within IBA.

2.3 Information exchange

If the Retrieval is to be delivered from Party A to Party B, both acting according to IBA, the sequential stream is probably the most convenient for B to consume as it can be done by using one of the available BIs.

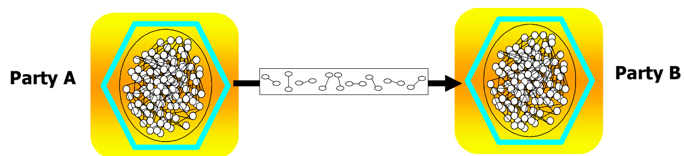


Fig. 6. Delivery as a set of Basic Statements

If, for some reason, the Retrieval is to be used as a new separate IC at Party B the delivery as an IC is a perfect choice since nothing extra at all needs to be performed by B. Just put the incoming IC at the right place and start work on it by using one of the available BIs.

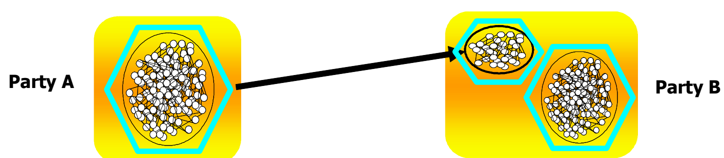


Fig. 7. Simple IC retrieval

In the future, it is to be expected that a delivery from A to B rarely will be direct. Instead a number of intermediates will be part of the delivery. This is already a fact in connection with the WS-Security standard where intermediates are supposed to act to control, confirm, add trust to, guarantee, mediate, …. the delivery and/or its content. Each such role needs to work with the information, maybe add some information as a result of its performed role and/or delete some other information of no value any longer for the continuing process.

Keeping the delivery as an IC between each processing step means a big leap in performance and simplicity. The information is ready to be processed at each intermediate's convenience via a BI without any pre or post transformations.

The IC alternative is also a good choice if B as the endpoint of the delivery wants to check and perhaps only select pieces of the incoming information.

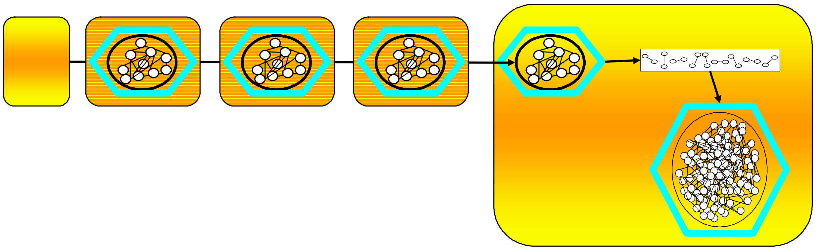


Fig. 8. Multi-step delivery

Also, a process may involve several parties as well as multiple process steps. During the execution of the process there is often a need to keep the state of task to be performed. Each process step acts based on this state as well as makes the expected changes to the state according to its role in the overall process. If those steps are executed by digital services based on IBA they may all work directly on one and the same IC delivered between the services and representing the actual state of the process. Simple and powerful.

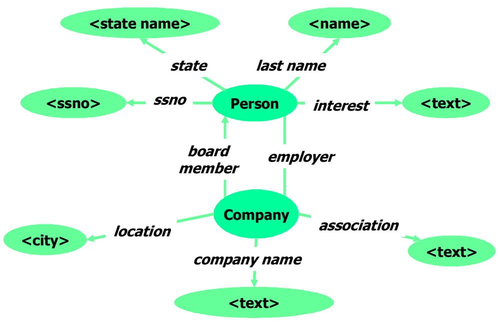
Obviously, also other exchange formats may have to be supported as options. This is for instance true to and from a Party working with some other structuring language (e.g. XML). Mapping between the exposed binary format and the exchange format of XML is supported. IBA also includes facilities (report generators) capable of mapping BI outputs to structured information readable for humans. Generally, those types of facilities are expected to be offered by vendors specializing in different needs and contexts.

2.4 Conceptual Model based Management

So far the content of an IC could be anything – any Basic Statements - about the UoD of interest, without any restrictions. This is fine and preferable in some situations. In others, probably in most cases where information is to be exchanged on a professional basis, there is a need to restrict this freedom to a limited set of allowed types of Basic Statements. This is done by specifying a *Conceptual Model.* A Conceptual Model is an abstraction of something real or imagined (UoD) of interest. The Conceptual Model is in this context a representation of the semantics of the included concepts - concepts used to give meaning to different combinations of information about some identified UoD of interest that parties choose to keep or to exchange with others. Database applications have a long tradition in using implementation dependent schemas to manage the data in databases.

IBA is centred on an extensive use of Conceptual Models. The Conceptual Model is in fact one of the focal points of the architecture.

A simplified Conceptual Model for the IC in figure 3 is shown in figure 9 a. An alternative graphical notation including more of the restrictions that can be identified from the UoD, or for some other reason chosen to be included, is shown in figure 9 b. The asterisk is indicating that ssno can be used as a unique reference to a Person. In the general case a combination of attributes and/or relationships is needed to form such a reference. A Company may have at least 3 and at most 30 board members while a Person may act as a board member in 0 or more (M) Companies. And so forth.



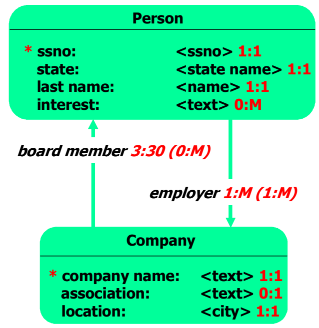


Fig. 9 a. Conceptal Model; free format **Fig. 9 b.** Compact format

A Conceptual Model is well suited to be used as a schema for managing the content of an IC (figure 10). Compare with a relational database approach (in parentheses).

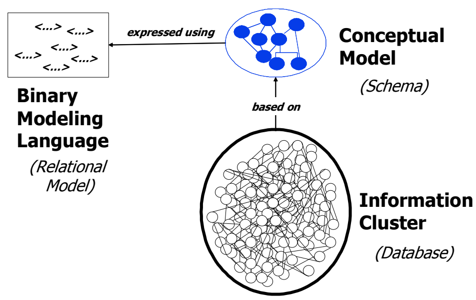


Fig. 10. Conceptual Model controlled Information Cluster content

2.5 Conceptual Model Based Exchange

In an open environment the sender and receiver supposedly have their own opinions about things expressed in *Internal Models*.

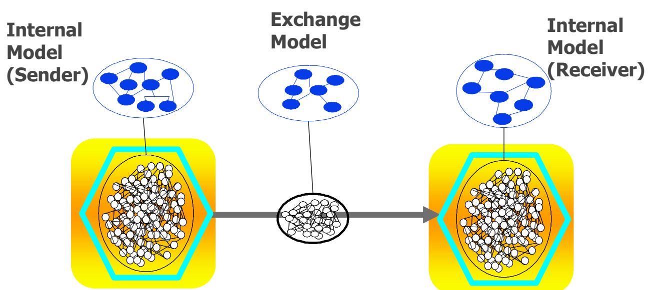


Fig. 11. Conceptual Model controlled information exchange

Therefore, when exchanging information they have to agree on a Conceptual Model for the semantics just for the information to be exchanged. If this model is different from the internal models, which usually is believed to be the case, mappings to and from this *Exchange Model* have to be performed.

Now something interesting. Conceptual Models might in their own rights be UoDs for those working with models. Those Conceptual Models are also expressed in BML and as such forming ICs. This is perfectly fine. The concepts in BML are here used to create abstractions of Conceptual Models. Such an abstraction may act as a Conceptual Model for working with Conceptual Model information. The bonus effect is an exposure of a Conceptual Model as any other type of information to be operated on by the BI with or without a restricting Conceptual Model (schema). See figure 12.

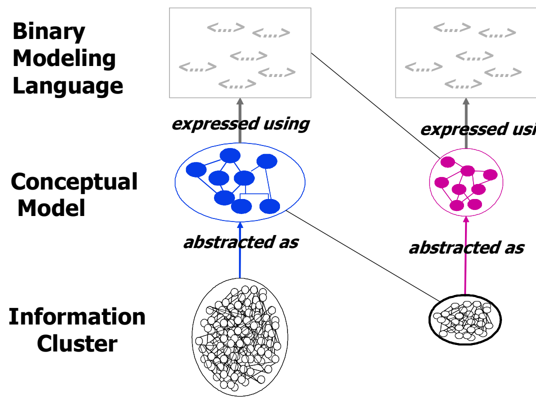


Fig. 12. Different abstraction levels – the same approach

By the way, this opens up possibilities to exchange information about almost anything without the need for predefined Exchange Models – and still with interpretable semantics at hand. Just send the Conceptual Model together with the information. The receiving Party collects both types of information by using the same IBA facilities.

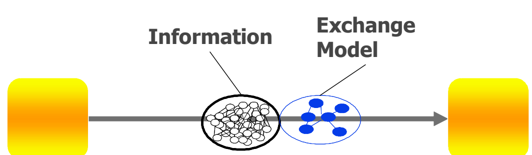


Fig. 13. Information exchange; about anything; in dynamic communities

Figure 14 adds another important specification level to the information exchange. Given an Exchange Model that expresses a general UoD, agreed upon among parties to be of interest for information exchange, many types of information deliveries (messages) are possible. Some are of a permanent type, others more temporary of interest. Each specific type is in IBA called a *Content Definition* (cf. Document Type Definition – DTD in XML). A Content Definition defines its semantics and structure by reference to the agreed Exchange Model. Each Content Definition is also expressed as a set of Basic Statements forming an IC in its own right (marked in green).

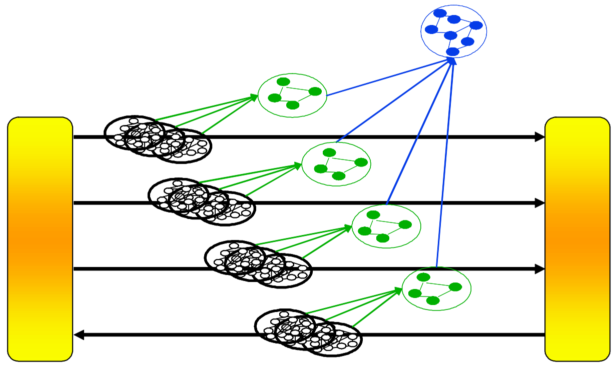


Fig. 14. Delivery Content Definition

In all, the UoD Information, Content Definitions, Internal Models, and Exchange Models are all defined as Information Clusters. Which make them all accessible via the Binary Interface (BI). The Binary Modelling Language (BML) in combination with the BI means uniformity. Those ICs, all expressed in BML and including bridging entity references, mean integration.

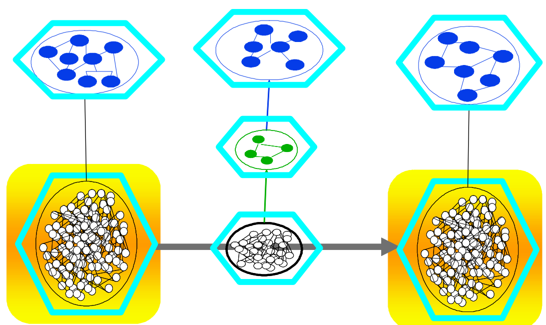


Fig. 15. The Binary Interface to be used at any level

Please note that so far only exchange between two parties has been discussed. Obviously, in the general case, the same approach is applicable for any group of parties that may agree on an Exchange Model and a set of Content Definitions. Or just as well, without such agreements (according to figure 13) when that suits the conditions at hand.

3 Information exposure

Parties are expected to keep their valuable information as internal assets. As a consequence, the information they are willing to open up to external parties has to be carefully evaluated so that this exposure always reflects the interest of the party from an overall business perspective. Furthermore, that externally available information probably has to be derived in more or less complex ways from internal databases and applications and explicitly structured for exposure. This structure is expressed in an *Exposure Model* (figure 16 a). Alternatively, a Party may choose to expose its information divided into several different subject areas where each area is represented by its own Exposure Model (figure 16 b).

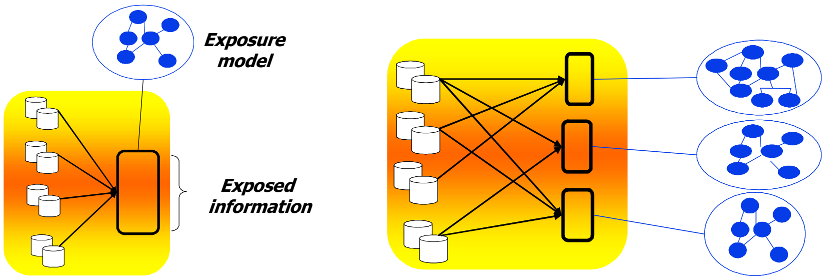


Fig. 16 a. Information exposure **Fig 16 b.** Multi subject information exposure

Those Exposure Models as UoDs are all preferably modelled according to the concepts of BML as shown in figure 12 above.

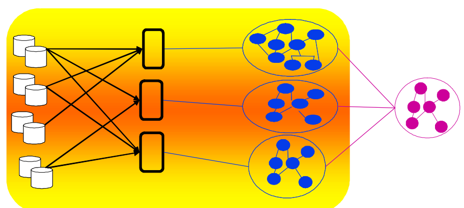


Fig. 17. Using a BML Conceptual Model for retrieval of model information

From a global perspective each Party has the responsibility for its own exposure (blue models in figure 18). It is expected that Parties sometimes form communities of common interest. When it comes to sharing information among members, this common interest is supposedly manifested by one or more Exposure Model agreed by the members (red and brown models in figure 18).

Not unlikely, this global society of exposing Parties will include mediators and brokers of different kinds searching for existing information within their specialty fields, in turn exposing this information or giving advice about it to interested “customers” (lined Parties in figure 18)

Presumably, interested Parties would prefer to have some standard way to inspect the offers of others on this web. If the Exposure Models are all formed by the use of BML, the Conceptual Model of BML could be used as a common Model Exposure Model. If that model in turn is expressed using BML – which by the way is the obvious choice – the BI could be used by all Parties when looking for interesting types of information everywhere else on the web. A simple yet powerful solution!

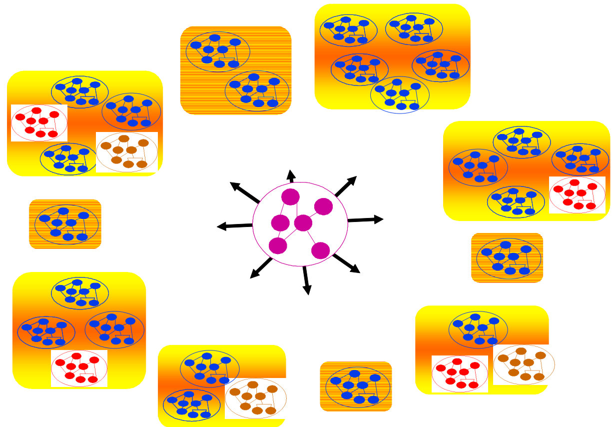


Fig. 18. A global perspective

So far only information has been exposed. Why not expose everything else that might be externally of interest and available in the same way? For instance, all types of information that the Web Services Framework document about Services in their WSDL and UDDI standards might easily be modelled in Binary Conceptual Models. (In fact a much more attractive alternative than to use XML for the same thing.) Other Resources (files, documents, books, pictures, videos, music, …) of different kinds are expected to fill the web as well. And to be available in free or restricted ways. They can be described directly in Exposure Models but also in alternative ways for specific purposes through Metadata. Metadata is also used to inform about other digital and non digital products the Party is selling, renting, building, …. And so forth.

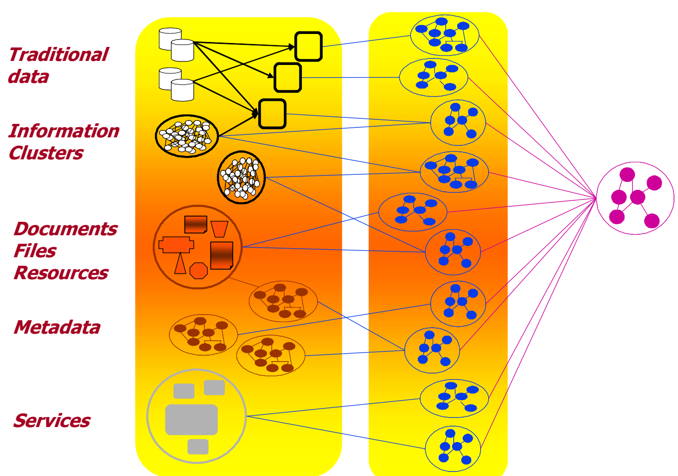


Fig. 19. A full range of exposure information

Add to that some general information about the Party and its business in the form of a *Generic Model* and we get a thorough picture of the Party and what it has to offer in different contexts.

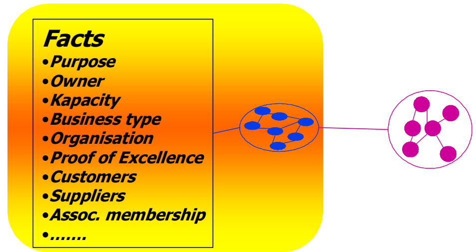
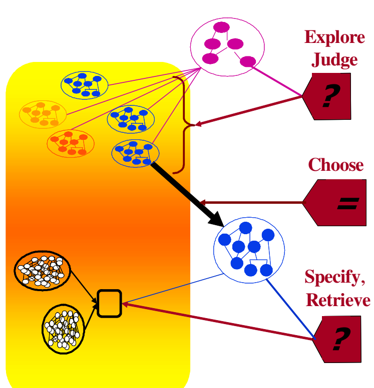


Fig. 20. General Party information

Suppose there is a Party looking for some information of interest. The Party probably starts by contacting some broker to get hints on good places to search for the information of interest. Picking one of the places, the Party contacts its Generic Model to find out more about what information subjects that is being offered. Each subject points at an Exposure Model and available Content Definitions thus making it possible to perform a more thorough content analysis. If the Exposure Model is found to be a model standardized by some consortia that the Party is well aware of and understand, the next step is just to start using the Exposure Model and its available Content Definitions for retrieval of information of interest. If the Exposure Model is not known to the Party it has a choice to try to understand its concepts by analyzing the model information in more detail. By doing so it hopefully gets an enough conception of the type of information the Exposure Model is modelling to make a decision if it is worthwhile to search the content or not.

Figure 21 shows some of the steps where – surprise – not only the models but also the information of interest is in binary form.

To follow up, figure 22 shows where the BI is available as interface. Almost everywhere! Add to this the capabilities shown in figure 15 above and the flavour of the Integrated Binary Architecture is at hand.



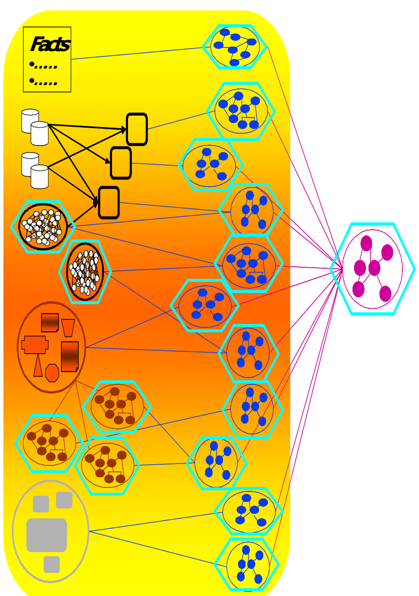


Fig. 21. Approaching Exposed information

Fig. 22. The Binary Interface for every type of information retrieval

4 Finally

IBA is a new approach to information management, exchange, discovery, interpretation, and sharing - all nicely integrated into a unified architecture. The architecture is well suited for the future Semantic Web and as such believed to have qualities not found in XML-based counterparts. However, it is recognized that IBA, as is stands today, by no means is any final solution. It doesn't even intend to take a big leap forward. IBA is just a natural evolution of ideas, specifications and implementations having their origins decades back.

In all, IBA is based on some obvious and surprisingly simple facts and ideas that support an integrated architecture for efficient and flexible storage, retrieval, interpretability, and exchange of information. This capability is a natural fit for almost any number of application areas. IBA is a foundation and generic platform on top of any available transport and message passing layers. It is expected that different types of facilities dynamically will evolve on top of IBA in support for specific requirements and applications areas (and built by any interested parties).

References

1. Langefors, B.: Theoretical Analysis of Information Systems. Studentlitteratur, Lund, Sweden (1966, 1973)
2. Berild, S., Nachmens, S.: CS4: A Tool for Database Design by Infological Simulation. In: *Proceedings of Very Large Data Base Conference (VLDB1977), Tokyo* (1977)
3. Janning, M., Nachmens, S., Berild, S.: CS4 – An introduction to Associative Data Bases and the CS4 System. Studentlitteratur, Lund, Sweden (1979)
4. Abrial, J.R.: Data Semantics. In: Klimbie & Koffeman (eds): Data Management Systems, North-Holland (1974)
5. Senko, M.E.: Conceptual Schema, Abstract Data Structures, Enterprise Descriptions. In: Morlet & Ribbens (eds): International Computing Symposium, North-Holland (1977)
6. Chen, P.P.: The Entity-Relationship Model – Toward a Unified View of Data. In: ACM Transactions on Database Systems (1976)