Ontology-Based Integration of Business Intelligence¹

Longbing Cao¹, Chengqi Zhang¹, Jiming Liu²
¹University of Technology Sydney, Australia
²Hong Kong Baptist University, China
¹{lbciao, chengqi}@it.uts.edu.au, ²jiming@comp.hkbu.edu.hk

Abstract

The integration of Business Intelligence (BI) has been taken by business decision-maker as an effective means to enhance enterprise “soft power” and added value in the reconstruction and revolution of traditional industries. The existing solutions based on structural integration are to pack together data warehouse (DW), OLAP, data mining (DM) and reporting systems from different vendors. BI system users are finally delivered a reporting system in which reports, data models, dimensions and measures are predefined by the system designers. As a result of a survey in the US, 85% of DW projects based on the above solutions failed to meet their intended objectives. In this paper, we summarize our investigation on the integration of BI on the basis of semantic integration and structural interaction together. Ontology-based integration of BI is discussed for the semantic interoperability in integrating DW, OLAP and DM. A hybrid ontological structure is introduced which includes conceptual view, analytical view and physical view. These views are matched with user interfaces, DW and enterprise information systems, respectively. Relevant ontological engineering techniques are developed for ontology namespace, semantic relationships, and ontological transformation, mapping and query in this ontological space. Ours is a promising approach for business-oriented, adaptive and automatic integration of BI in the real world. Operational decision making experiments within a real telecom company have demonstrated that the use of a BI system results in such a decision being more flexible and also more potential.

Keywords: business intelligence, ontology, ontological engineering

1 Introduction

The integration of Business Intelligence (BI) [1, 2] aims at providing systematic analyses and decision supports by combining queries, reporting and analysis tools. In general, the process of constructing a BI system in industries is as follows: first, extracting, transforming and loading data dispersing in relevant enterprise information systems (EIS) into a data warehouse (DW) system [3] in terms of predefined subject-oriented data models; then, linking analytical engines such as predefined and ad hoc reporting systems, online analytical processing (OLAP) and/or data mining (DM) engines to the DW to present analytical reports. The above analysis-oriented process supports decision-making which cannot be served by existing EIS. Therefore, BI has been taken as an effective means to enhance enterprise “soft power” and added value in building and transforming traditional industries after a one-off substantive investment on infrastructure.

However, commercial BI solutions normally implement the BI integration via packing all BI-related packages available from vendors [1, 4, 5, 6, 7, 8, 9, 10] through structural integration based on schema and/or open data connectivity. This has resulted in the difficulty of deploying custom-designed and custom-built BI systems in the real world for several reasons. For instance, organizations developing BI systems request business-oriented capabilities for friendly personalization and interoperability based on business jargon other than technically predefined notations and syntax in existing solutions. Furthermore, existing systems cannot effectively adapt to a dizzying flurry of dynamic requests on user changes, analytical models, data sources update and enhancement requirements.

¹ This work has been sponsored by Australia Capital Markets CRC, and UTS Research Funds.
Some comments and feedbacks we captured from Chinese telecom customers [11] were: “We feel a BI system is not practical and flexible at this moment. We are delivered with thousands of reports designed by the vendor, but few are interesting. Actually we cannot perform real and dynamic analyses by ourselves as we like every day” (from a China Mobile manager). One of China’s Unicom salesman told us: “The current BI system is not friendly to me, I don’t know the meaning of those technical symbols in reports, I prefer to interact with my favorite business terms in a personalized portal in Chinese.” We can further list pages of relevant feedbacks. As a survey shows, in the US, 85% of DW projects failed to meet their intended objectives, and 40% didn’t even get off the ground [12].

Therefore, the integration of BI is actually a very challenging work for not only technical issues but also human involvement and business-oriented requirements. The weakness of existing BI solutions in dealing with the challenge to some degree results from the reality that existing work on warehousing mainly focuses on the structural integration [13] of data sources, warehouse and analyses. They lack semantic integration mechanisms which can assist in friendly and adaptive interaction and deployment of BI. A good strategy for an effective integration of BI is to combine the structural integration and the semantic integration.

In this paper, we investigate ontology-based integration of BI since ontology has great potential in addressing semantic integration [2, 14, 15, 16]. Ontology-based semantic integration involves domain knowledge, semantic understanding, aggregation, mapping, query and discovery of ontological items.

The basic idea of the ontology-based integration of BI is as follows: as well as the existing structural linkage such as schema-based information integration, we also design an ontology-based logical communication channel and architecture for integrating EIS, DW and reporting systems. Furthermore, we identify and develop relevant ontological engineering techniques including ontology namespace, semantic relationships, ontological transformation, mapping, discovery and query for combining and transforming ontological items across these systems. Finally, we advocate that a unified knowledge portal should be built to manage the universal integration of all BI packages, support ontology-oriented analysis, management and interaction user-friendly and adaptively.

The combination of the traditional structural integration and our proposed semantic integration leads not only to predefined but also user-friendly, adaptive and automatic interaction, integration, representation and management of BI [2, 16, 17]. This approach somehow supports seamless transition from a practical workspace into the virtual business-oriented analysis world that is expected by businesspersons. A collection of user-friendly supports can help users to modify, update, create or re-arrange ontological items and functionalities at different granularities on demand. This can benefit businesspersons in a business-oriented rather than technology-centered interaction. It also results in run-time capabilities that help business analysts to adapt to changing or new environment flexibly and adaptively in a user-friendly manner. These are not achievable through simply adding BI packages together based on structural integration.

In Section 2, we discuss business and technical complexities in integrating BI. Section 3 introduces a hybrid ontology architecture for ontology-based integration. Ontology multi-namespace and semantic relationships are discussed in Section 4. In Section 5, ontology mapping, transformation and query are described. In Section 6, a unified knowledge portal is advocated to organize the ontology-based integration of BI, an integrated BI system prototype and customer evaluation are also presented. We conclude this study in Section 7.

2 Complexities in integrating business intelligence

A BI system is normally built through packing all commercial BI-related products together. However, the process of integrating BI is actually quite complicated in the real world such as in the telecom industry. Complexities in this process may take the forms of openness, heterogeneity, distribution, evolution and emergence [18]. These are further embodied in, but not limited to, the following aspects and forms in terms of structural and semantic heterogeneity [19].

First, ubiquitous complexities coexist in the business environment, the underlying operational systems, and the interaction between business requirements and relevant information systems. For instance,

- multiple distributed BSS/OSS provide differentiated services such as billing, switches, accounting, customers services, operations and maintenances; whereas these systems usually are developed by multiple system integrators;
there are also many other related information systems such as customer relationship management (CRM), enterprise resource planning (ERP), office automation (OA), management information system (MIS) and external systems. These systems either take form as information resources or as the application and output of BI;

- hardware and software platforms and infrastructures involved are normally purchased from various vendors. The analysis and design specifications used for these systems are normally not based on a universally open standard;

- diversity of data structures are hidden in the above systems; these structures embed both structural and semantic heterogeneity;

- multiform business models and taxonomies that surround individual BSS/OSS, further lead to various operational workflows and information flows;

- colorful customer requests are put on varying localized and personalized services; and

- daily changes and emergent behaviors that come up from systems and operations.

Second, a predefined BI system cannot adapt to the daily emergent requirements of data and analyses in run time. To this end, it is sometimes helpful and necessary for users to undertake business analyses directly on EIS (usually on an operation data store rather than EIS). This can complement the normal analyses on top of DW when new changes are not reflected in existing data models.

Third, the incapability of the existing BI systems also results from internal differences and incompatibility in the following:

- specifications for system analysis and design,

- interpretations of some similar terms and relations,

- interoperability in system architecture and design patterns,

- supports for information integration, metadata management, and

- methodologies and concrete design methods used by different productivity systems, BI products vendors and BI system designers.

These make it very hard to construct an integrated BI system dealing with DW, OLAP and DM smoothly and transparently. Therefore, a BI system simply packing all the related components cannot make its objectives come into effect in a dynamic and live world.

Fourth, the semantic heterogeneity is ubiquitous and challenging in the data warehousing. Semantic heterogeneity may result from semantic conflicts such as confounding, scaling and naming conflicts [13], or ontological mismatches such as conceptualization and explication [19], scope and encoding of concepts [20], and model conventions, coverage and granularity, and paradigm [21].

Finally, business users of a BI system normally would like to interact with a one-stop workplace [2, 17]. They would expect to encompass all interactive interfaces and link all relevant underlying systems over a network. The one-stop portal should not only adapt to their existing business definitions, processes and rules, but also bring decision-makers with flexibility and intelligence to find both interestingness and scientific evidences hidden in heavy transactions. Therefore, it is significant for a BI system that it should support the dialogue between business-oriented and technology-centered worlds. This dialogue will assist customers to cross the great gulf between their daily concept-oriented business operations and processes, and the complicated physical views inside underlying DW and EIS.

Therefore, in the process of building a practical and productive BI system, the first and most important issue is to deal with the above structural and semantic heterogeneity and interoperability. This task actually gets involved in the whole process from business requirements arrangement, data preparation to intelligence discovery. This leads to the lexical and semantic integration and transformation of heterogeneous information dispersing in telecom business and operation systems. It also results in developing mechanisms for business-oriented interaction, and dynamic and adaptive integration of both DW and EIS as required.

3 Architecture for ontology-based integration

In this section, we propose a system architecture for the ontology-based integration of BI and also encompass ontological views for relevant problem domains such as DW, EIS and business-oriented human machine interaction. These actually form a hybrid ontological space. This space can support multiform analyses such as predefined and ad hoc reporting, OLAP and DM in a user-friendly and adaptive manner [2, 16].
3.1 Hybrid ontological space

A hybrid ontological architecture (as shown in Figure 1) is proposed to integrate user profiles, DW, OLAP, DM and underlying EIS. It aims to develop mechanisms for smooth mapping from user-defined keywords to metadata items in DW or physical attributes/entities dispersing in BSS/OSS.

A BI system can be categorized into three high-level layers from bottom up -- EIS physical layer, DW global analytical layer and top-level representation layer. Correspondingly, we build three ontological domains for them. They are: (i) EIS Ontologies enclosing EIS ontological items mapping to individual EIS, (ii) DW Ontologies embedding DW ontological items for models and metadata, and (iii) Business Ontologies consisting of business-oriented items for human-computer interaction. Additionally, a mediation layer is built for ontology mapping and query parsing intra or inter the above domains. The Figure 1 also outlines relevant functional components for the appropriate operation of the above relevant layers. These components deal with user interaction, data model, metadata management, query transformation and parsing, etc. To manage the operation of the hybrid integrated space, a collection of ontological engineering techniques are necessary. They include services for naming, representation, transformation, mediation, transport, directory, mapping and discovery of ontological items either intra or inter the domains.

![Figure 1. A hybrid ontological structure integrating business intelligence](image)

Figure 2 further illustrates relevant terminologies in individual layer and the interaction among the above three domains in an ontology-based BI system. Terminologies in existing BI system are mapped to ontological items. For instance, the original DW metadata items are transformed into its DW ontological items. With regard to the interaction, it shows the general linkage in existing BI solutions and ontology association between ontological domains for the integration of BI. The line with double arrows shows the general integration in BI systems. This is actually based on a two-step process. First, data dispersing in relevant EIS is extracted, transformed and loaded into DW according to predefined data model. Then, analytical engines such as OLAP or DM are linked to the DW to present reporting results to analysts. The analysts actually interact with the metadata items in the DW.

![Figure 2. Interaction in an ontology-based BI system](image)

On the other hand, the line without arrows shows the logical dialogue among related ontological domains. This indicates that the ontology-based business analysis can be performed directly on DW. In addition, it also supports interaction between ontological domains of Business and DW, or between Business and EIS if required. This facility cannot be served by existing BI solutions while it is expected by customers and complementary to existing DW-based analysis.
Furthermore, corresponding ontological engineering techniques [14, 16, 22] are required for the effective operation of the above ontology-based BI integration. These include the following: (i) building ontological namespaces and semantic relationships for organizing items in the above domains; (ii) mechanisms for ontological transformation and mapping intra or inter domains; (iii) services for ontological query and search in the warehousing. These aspects are main content of this paper. The rest of this section further briefly introduces the ontological domains in the above hybrid ontological structure.

3.2 Business ontology

Business ontology captures domain-specific concepts, entities and business rules from the business world. They specifically refer to those terms used daily by business analysts rather than the notations extracted directly from DW or EIS designed by technicians. There are two obvious features of the BO ontological items. The first is that a business concept can be interpreted as varying terms. The second one is that a business concept is usually presented in a native language spoken by business persons, but that may not be necessary as in English normally spoken by DW and reporting systems. For instance, the concept user_id is spoken as 用 户 ID by Chinese businesspersons.

A conceptual ontology base will be built for storing the business ontological items. It makes the business-oriented human-system interaction achievable, and this would be very welcome to business analysts. In an ontology base, items are organized as concept category directory (CCD). CCD is a hierarchical concept tree implementing the namespace of the problem domain, and defining and organizing all relevant terms and their relationships abstracted from business world. CCD consists of CCD entries. A CCD entry is an ontological record of the ontology base.

**DEFINITION 1.** CCD Entry: A CCD entry consists of a mandatory leading item (LI, an identifier) and optionally multiple substitute items (SI, recommended candidate concepts), and their values and constraints on them. The following formula presents the CCD entry structure encoded as key-value tuple (KVT) and key-property pair (KPP).

\[
\{\{\text{leading item, MO}\}: \{LI\_Value\}, \{\text{substitute items, OM}\}: \{SI\_Value1, SI\_Value2, \ldots \}\}\}
\]

LI and SI are used for dealing with semantic synonymous phenomenon commonly seen in enterprise systems and business life. As a result, a set of candidate concepts and expressions are included in CCD. These candidates benefit businesspersons who have favorite terms or work in a specific culture.

The generation of CCD entries is as follows. First, an ontology namespace (as shown in Figure 3 in Section 4.1) [2] is built via extracting ontological items and their relationships from a specific domain. Then, each ontological item is formulated into some representation form. For instance, the following lines define the LI and SI of business ontological CCD entry service provider in telecom services in an informal way, where BO refers to business ontology, ‘substitute_to’ (addressed in Section 4.2) points out the semantic relationships between LI and SI items.

```java
;; Definition of Service Provider in business ontology CCD
(BO Service_Provider LI)
(substitute_to Service_Provider_Label
(Service_Provider_Name, Service_Provider_Nickname, Service_Provider_Description SI))
```

It is recommended and worthy of noting, when developing the BO items, that a natural business term rather than a notation be used to describe an attribute or entity in the real world. For instance, a business term ‘condition’ should be used as a BO item which is associated with an ontology query rather than the word ‘where’.

3.3 DW ontology

Since OLAP and DM analyses are mainly undertaken on top of DW, DW ontology wraps technical and business metadata items in terms of domain-specific business environment and IT infrastructure where a DW system is located. These metadata items are composed of set of elements (attributes, dimensions and measures) related to data model of DW, source data and its extraction, transformation and loading. They also encompass information about data connectivity, interaction and its rules between DW and data sources, and DW, OLAP and DM. Similar to the representation of ontological items in BO, DW ontology is organized as an analytical ontology directory (AOD). Its entries are
defined in the following to represent DW ontological items.

**Definition 2. AOD Entry:** It represents a domain-specific metadata item. An AOD entry captures business and technical metadata rather than business rules and concepts. They are extracted from the problem domain and transformed into ontological domain.

To cope with the complex interaction between DW and its related domains such as EIS and DM, AOD entries are presented in the following attributes and specifications.

- globally unique identifier(\textit{gui}),
- recommended global name(rgn),
- candidate substitute names(csn),
- parent object(po, top-level coupled concept name),
- child objects(co, low-level EIS instances),
- analytical locator(al, where to find this entry from the bottom EIS resources, including related connection string, schema, metadata of resources, and so forth),
- close associators(ca, including actions and relationships with other neighboring entries).

As a result, an AOD entry is presented by the above attributes in the following form:

$$\{\text{gui}, \text{MO}, \{\text{rgn, MO}, \{\text{csn, OM}\}, \{\text{po, MO}, \{\text{ca, MM}\}, \{\text{al, MO}\}, \{\text{ca, OM}\}\}\}$$

where MO, OM and MM are cardinality constraints which define the existent items of relevant attributes as zero, one, or many in either a mandatory or a optional way. They are further specified in Table 1. In addition, all entries are stored into DW ontology base in KVT and KPP.

<table>
<thead>
<tr>
<th>Property Notation</th>
<th>Notation Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>MO</td>
<td>Mandatory one</td>
</tr>
<tr>
<td>MM</td>
<td>Mandatory many</td>
</tr>
<tr>
<td>OO</td>
<td>Optional one</td>
</tr>
<tr>
<td>OM</td>
<td>Optional many</td>
</tr>
</tbody>
</table>

### 3.4 EIS ontology

EIS is normally composed of multiple enterprise information systems, for instance, BSS, OSS, MIS, ERP and OA in telecom companies. In these EIS, huge number of transactions and relevant information are lodged in tables and attributes. In order to present these elements, EIS ontology encompasses all physical entities and relationships distributed throughout the underlying individual EIS. For instance, ontological domains are defined for billing systems, accounting systems, network management systems and other information systems respectively. We build individual EIS ontological domains for each subsystem of EIS rather than set up one overall ontological space for all EIS components. This is due to the internal complexity and dynamics in combining varying EIS. In fact, it would be hard and unsuitable to pack and manage all ontological items from all EIS components into one namespace. There is an obvious multiplicity in EIS embodied on heterogeneous instances, attributes and relations. For instance, the keyword \textit{customer name} on user conceptual view may take forms as \textit{Customer Name}, \textit{Customer Label}, \textit{User Name}, \textit{User Label} etc. in EIS systems like billing, accounting, switch, and operation and maintenance systems.

### 4 Ontology namespace and semantic relationships

The above ontological architecture actually encloses a logical communication channel beyond the traditional structural linkage among reporting, DW and EIS. To make this channel actionable, an initial task is to organize and manage all ontological items via ontology namespaces [2] and semantic relationships [22, 23]. These are fundamental for the further interaction and management of ontological items across multiple ontological domains.

#### 4.1 Hybrid multi-namespace ontology management and representation
For a meaning to be shared without disagreement of ontological items, it needs to be supported by concepts that are expressed in precise terms. Also we need to explicitly declare and organize all concepts and relations in a systematic way. These are handled by ontology namespace and representation.

As the abovementioned complexities show, it would be very difficult to pack all ontological items from multiple domains into one ontological namespace. In light of this consideration, we set up a hybrid multi-namespace strategy [2] which tries to organize the heterogeneous and multiplicate ontological domains into an ordered but correlated ontological family for a specific problem domain. The idea of this hybrid multi-namespace strategy is exactly the same as the Universal Resource Locator namespace for managing the Web domain names.

For every specific ontological domain, we build one or more ontological namespaces for them on demand. However, since all these ontological domains belong to one problem domain (related to a root domain name), so we link all of them into a loosely coupled global-level ontological namespace via some second top-level domain names. For a telecom BI system, this strategy is instantiated as the following namespaces.

- A unified namespace is built for organizing and managing ontological items in the BO domain and is closely related to user profiles, business process and requirements in a problem domain. As mentioned before, this namespace may include elements in some local language other than English.
- For a DW domain, all DW ontological items are managed in one namespace.
- To manage the multiplicity and heterogeneity of EIS systems, ontological items in each sub-EIS system can be stored into one specific EIS namespace. For instance, different namespaces can be designed for telecom billing systems, accounting systems and balance systems respectively.
- A root and its second top-level domain names are built for the combination of all the above domains.

Taking the telecom BI system of China telecom industries as an instance, the namespace must be expandable for coping with the reality and complexity in six national-wide telecom operators. They provide specific and somehow overlapped services using different technologies, specifications and metrics in the individual BSS/OSS and business process. However, almost every operator has set up province-wide branches that hold individual BSS and OSS developed by self-nominated integrators. To manage ontologies in these operators, as shown in Figure 3, all of them belong to the root telecom. Under telecom, there are six second roots divided for all national-wide operators. For instance, telecom:cm is the root of China Mobile. Then under these second roots, we can organize ontologies in terms of business types or BSS/OSS system types. For instance, Figure 3 further shows an excerpt of the abstracted ontological items for telecom voice services which are the most common business in telecom. It lists some ontological items in billing (cm:billing) and balance (cm:balance) systems under telecom:cm.

![Diagram](image-url)

Figure 3. Hybrid multi-namespace ontology management

Our further work is about how to present ontologies in the above namespace. We refer to design strategies used in the specifications of XML namespace [24], RDF [25], and OWL namespace [26]. Our strategy [16] for expressing ontologies is as follows: (i) using XML specifications to declare ontology concepts, (ii) following the rules, constraints, defaulting, overriding, scoping used by XML
namespace elements and attributes, and (iii) adopting rules and functions from OWL and RDF for the representation on demand. For instance, the following lines declare an ontological item for a local voice call. It is a subclass of class voice (subclass is a semantic relationship which is introduced in Section 4.2), the label for it is local_voice in English and in Chinese.

```xml
<owl:Class rdf:ID="LocalVoice">
  <rdfs:subClassOf rdf:resource="&voice;"/>
  <rdfs:label xml:lang="en">local_voice</rdfs:label>
  <rdfs:label xml:lang="zh">本地电话</rdfs:label>
...
</owl:Class>
```

### 4.2 Semantic relationships

In nature, semantic heterogeneity results from the semantic difference (D) between the natural meaning (M) and its conceptualization (represented by some notations (N) such as terms) of a real world entity or concept (C). So, we obtain a tuple as $D := (C, M, N)$. Different combinatory contexts among $N$ incur differentiated $M$, and further lead to varying degrees of $D$. Table 2 lists some combinatory scenarios between items of $N$ and relevant relations between $M$ in terms of semantics.

| $M_1$ ? $M_2$ | $N_1 = N_2$ || $N_1 \napprox N_2$ | $N_1 \not= N_2$ |
|---------------|--------------|-----------------|-----------------|
| $M_1 \equiv M_2$ | Equivalence, similarity | Synonyms, encoding, conventions, paradigms, scaling |
| $M_1 \supseteq M_2$ | Scope, coverage, granularity | Generalization, specialization |
| $\emptyset = M_1 \cap M_2$ | Naming conflict, homonymy | Disjointness, antonyms |
| $\emptyset \not= M_1 \cap M_2 <_{\text{min}}(M_1, M_2)$ | Scope, coverage, granularity | Overlapping |
| $M_1 \ni M_2$ | Naming, encoding | Instantiation |

These combinatory scenarios furthermore result in varied semantic relationships among items of $C$. Semantic relationships [16, 22] refer to semantic dependencies between ontological items in a domain or across domains. We identify seven types of semantic relationships to organize the semantic dependency existing in the context of ontology-based warehousing. They are Aggregation, Association, Disjointness, Generalization, Instantiation, Overlap and Substitution, respectively. The corresponding predicates for them are part_of, associate_with, disjoint_to, is_a, instance_of, overlap_to and substitute_to. The following are definitions of these relationships where the symbols $O$ and $o$ refer to an ontological class and its instance.

- **part_of** ($O_1$, $O_2$): two ontological classes $O_1$ and $O_2$, where $O_2$ is part/member of $O_1$, or $O_1$ is made of $O_2$
- **associate_with** ($O_1$, $O_2$): it represents a relationship between $O_1$ and $O_2$ which cannot be specified by any of the above six; but $O_1$ and $O_2$ are associated with some linkage defined by users e.g. for business requirement
- **disjoint_to** ($O_1$, $O_2$): it stands for that $O_2$ is independent of $O_1$
- **is_a** ($O_1$, $O_2$): the relationship between $O_2$ and $O_1$ is subtype/supertype or as subsumption, i.e. $O_2$ is a $O_1$, or $O_1$ is a kind of $O_2$. The is_a sometimes is also called as subclass_of
- **instance_of** ($O$, $o$): two ontological elements $O$ and $o$, $o$ is an instance of the ontological class $O$
- **overlap_to** ($O_1$, $O_2$): it represents that there is something shared by both $O_1$ and $O_2$; but the share percentage is not high enough for one to be substituted by another
- **substitute_to** ($O_1$, $O_2$): $O_1$ and $O_2$ are identical or to a large degree similar; in this case $O_2$ can be substituted by $O_1$
5 Ontology mapping, transformation and query

Ontological mapping is necessary for the ontology integration. In this section, we first introduce an ontological mapping structure [16] for the matchmaking of ontology across domains in an integrated BI system. Then the ontological transformation and mapping rules [22] are briefly discussed, as it supports the aggregation and matching of ontological items intra or inter domains.

5.1 Mapping structure across multi-namespace

In Section 4.1, we propose a hybrid multi-namespace strategy for naming and managing ontological items in an ontology-based BI system. The proper work of such a system would then greatly depend on the ontology mapping across these namespaces. In order to design an appropriate mapping structure, let’s first review a generic BI analysis process in a BI niche.

First, business analysts select an analytical subject and a method according to their interestingness. They further specify target dimensions and measures by selecting or typing in favorite keywords that are candidate items in the keyword items. The keyword items are transformed to relevant BO items in the BO. For instance, the keyword “User Name” may be transformed to “Customer Name” in BO. Then, if DW is specified as the target analysis space, the “Customer Name” is further matched with a DWO item, for instance, {...Customer_Name...}, in the DWO. It further is directed to a metadata item as {...customerservices...Customer_Id...} in the DW. In some case, a business support system, for instance, the billing system, may be selected by users to do further analysis that cannot be supported by models in DW. In that case, the “Customer Name” in BO will be directed to ontology domain of billing system through routing by the universal resource identifier (URI) items of EIS. The ontology concept in billing domain will be transformed to relevant attributes in billing system. Finally, the resulting query reports are returned to user interface in user-defined keywords.

Figure 4 sketches the relevant matching relations in the above process. For each item in BO, one-to-many relevant keyword items may be specified by different users. One BO item will be matched with one DWO item in the DWO, but there may be one-to-many EIS URI items which identify all linkages to underlying specific EIS ontology domains.

To undertake the above analytical process, semantic aggregation and transformation [22] are essential. Doing this leads to developing the corresponding mechanisms for (i) aggregation of semantic relationships, (ii) aggregation of ontological items, and (iii) transformation of one ontology item to another involved in either intra or inter domains. To this end, we focus on building semantic mapping rules [22], so that we can deploy these rules to aggregate and transform semantic relationships and ontological items across namespaces.

5.2 Rule-based ontology aggregation and transformation

For the semantic aggregation and ontological transformation of ontological items, there are three basic mapping types in warehousing including concept-to-concept mapping (e.g. resident-home), attribute-to-concept mapping (id-home), and attribute-to-attribute mapping (local_fee-local_call) as
shown in Figure 5. A fundamental work for supporting the above mapping is to develop mapping rules for (i) aggregating ontological items, and (ii) transforming an ontology item to another involved in either one ontological domain or multiple domains.

Fig. 5 Basic mapping types

Developing formal rules for ontological aggregation and transformation is vital towards the well-founded and automated interoperability of ontologies across domains. We mainly focus on developing formal rules to support transitivity, additivity and antisymmetry in transforming and reducing semantic relationships and ontological items. By contrast, the aggregation of semantic relationships is more basic for further transformation of ontological items. It can simplify the combination of semantic relationships, and find the final reduced semantic relationship.

For all the seven semantic relationships discussed in Section 4.2, we develop relevant rules [22] supporting the semantic aggregation of the above semantic relationships. For instance, the following rule 1 shows an excerpt for some cases.

**RULE 1** Let ontological items $A$, $B$ and $C$ be associated by the Generalization relationship, then

- $\forall (is\_a(A, B) \land is\_a(B, C)) \Rightarrow is\_a(A, C)$
- $\forall (is\_a(A, B) \land is\_a(A, C)) \Rightarrow is\_a(A, (B \land C))$
- $\forall (is\_a(A, B) \land is\_a(B, A)) \Rightarrow substitute\_to(A, B)$

These sample rules show that rule-based aggregation and transformation make it easier to understand the relations between multiple ontologies and integrate semantic relationships.

Furthermore, we need to develop semantic rules for aggregating and transforming ontological items in one or across multiple namespaces, so that a BI analysis triggered in BO domain can be performed on target namespace such as DWO or EIS.

The rule-based aggregation of ontological items is to develop semantic rules that reduce and integrate ontological items by analyzing logic and semantic relationships associated with items. The transformation between ontological items could be a mapping from an arbitrary keyword to its relevant items in BO domain, or from BO to another domain such as DWO or one of EIS0 domain. Rules for these aggregation and mapping of ontological items are developed [22]. For instance, the following Rule 2 shows an aggregation of ontological items. Rule 3 presents an example of transformation of ontological items. In practice, a set of similar rules [22] are developed and stored into rule knowledge base for automatic implementation of the ontology-based integration of BI.

**RULE 2**

- $\forall (A \land B), \exists B :: is\_a(A, B)$
  $\Rightarrow B$, the resulting output is $B$

**RULE 3**

- $\forall C, C_r (\neq \emptyset), \exists (is\_a(O, C) \land is\_a(O, C)) \Rightarrow O$

### 5.3 Ontology-based query

As discussed in Section 5.1, the ontology mapping is a multi-step process. Similarly, ontology-based query [2] is composed of multiple operations in some query language. After user requests are typed into an analytical portal, first they will be transformed to BO items in order to find the relevant ontology concepts in user profile. The BO items will further be matched with DWO items in DWO domain. And then they will be mapped to metadata items in DW. If EIS is queried, the BO items will be matched with URI routing items, and then routed to concrete attributes existing in a specific information system.

An ontology-based query would be executed as the following steps.

Step1: User log on to knowledge-based business intelligence portal;
Step2: Locate analytical subject or special subject;
Step3: Locate target analytical space (in the DW or EIS);
Step4: Locate analytical method;
Step5: Input/specify key terms in user personification;
Step6: Recursively query key words in user profile;
Step7: Recursively search ontology concepts in ontology space of user profile according to key words;
Step8: Recursively search ontology concepts in ontology space of DW and EIS;
Step9: Map to corresponding metadata items in DW or physical attributes in EIS;
Step10: Data transformation and integration to generate target analytical report;
Step11: Return to user interface and present in user-defined terms.

To support the above steps, many ontology services will have to be developed. For instance, services for ontology mapping and the aggregation and transformation of semantic relationships and ontological items as discussed in Section 5.1 and 5.2 will be involved in the ontology query. These services implement automatic mapping, aggregation, transformation, and query parsing between concepts in BO and ontological items in DWO or EISO.

The above ontology service-based query would benefit from some interactive modules [16] enclosing relevant service supports. For instance, the following job can be supported in an interactive component. Users select metadata items and ontology items, define relations (by predicates) between ontology items and metadata items, set limit conditions (cardinality properties) for relevant items and attributes, and group and/or sort them where appropriate. Then, a query parsing rule can be generated automatically. After a query rule is generated, users can check it, and make some modifications if necessary before submitting it.

In addition, ontology search services are also required in a BI system integrating user profiles, DW, and EIS into a knowledge portal. There are probably three types of search services [22]. They are following:

- Search ontological items in BO correlated to user-defined terms.
- Search ontological items in another ontology domain correlated to items in BO. For instance, looking for a corresponding item in DWO that is matched with a source item in BO.
- Search ontological items in DWO or EISO that is mapped to the user-defined terms.

6 Unified knowledge portal

We believe that a one-stop analytical portal is helpful for a businessperson-BI system interaction [17, 2]. This one-stop entry needs to support the following features.

- It links and captures any required data and information flow existing in the relevant EIS;
- It fulfills online four-level of analyses such as predefined and ad hoc reporting, and OLAP and DM;
- More importantly, it performs business-oriented user interfaces in the whole interaction phase including requests and responses, businesspersons access elements in either DW or EIS at a favorite granularity of business concept as they like;
- It undertakes adaptive capability to multiform of business terms in different production systems, and to emergent updates in either DW or EIS;
- It embeds supports that make automatic aggregation, transformation and mapping achievable. This leads to an relatively automatic ontology-based integration of BI;
- In addition, unified security certification and audit are supported beyond the existing mechanisms in underlying DW or EIS;
- Finally it provides user-friendly interfaces and services for human participation if required. This mainly results in the involvement of domain experts if no one else can solve the problem.

An ontology-based BI system with strong supports to the abovementioned features is called as a Unified Knowledge Portal (UKP). Through this UKP, business intelligence emerges from the underlying EIS, DW and OLAP and DM systems as user-friendly, adaptive and automatic. In this UKP, measures, dimensions, attributes and members are not presented by the notation specifications specified in the DW, rather they are presented as business concepts in a flexible and efficient manner. These are well known to users. To users, complexities of multiple heterogeneous EIS resources, ETL process, ODS, DW, reporting presentation, and mapping from business concepts to underlying
physical entities are shielded under the one-stop interface. Business analysts can easily launch analysis and observations without worries of underlying symbolization, authorization and information management from DW to bottom EIS.

Following the ideas in this paper, we have built some prototype systems [16, 27] in the real world. XReport2 [27] is a business analysis and reporting system built on the theory of ontology-based integration of data warehousing. It supports user-defined business-oriented keywords and ontology concepts and constraints, and generates analytical reports in user personification automatically and manually.

We have it tested it in China Netcom International (CNI). CNI used to use Brio as their reporting and analysis systems [9] for almost two years. In this period, they met with some key challenges including user unfriendliness, inflexibility, and incapability for supporting dynamic and transparent data access from underlying EIS. Our XReport was invited by CNI to have a test in the real environment. The evaluation environment for XReport is as follows. The main business involved was Customer Services of CNI. Two involved business supporting systems were run on two Linux-based IBM server xSeries 206, with 512M memory and 3.0GHz Intel® Pentium 4™ 1* Processor. IBM UDB 7.2 was used as database. XReport was installed on xSeries 305 with 2.8GHz processor and 512M memory, SQL Server 2000 was installed as database. IBM DB2 client was installed for connecting to the production database servers.

The task of this evaluation is to online generate periodically aggregated reports on request of analysts. XReport generated around 100 aggregated tables by summarizing different measures in terms of user interested branches, products, categories, sub-categories, and date period from more than 10,000 operational tables of customer services related systems in CNI. In addition, this test also evaluated the performance of user friendliness, efficiency, and distribution and manageable ability.

For user friendliness, they checked whether our system could support business terms-oriented interaction between analysts and underlying customer services support systems. With efficiency, they wanted to see whether it supported data extraction, transformation and aggregation through multiple steps from a huge number of operational records. With regard to distribution and manageable ability, they tested whether the system supports distributed requirements of business analyses from multiple branches of CNI, and whether personalized user profiles like business-based report templates, reports, tasks, and so forth were supported.

After the test in the Operation & Share Center in CNI in March 2004, the General Engineer and the Manager of the Center gave the following feedback and comments about the test: [27] “different from the Brio-based reporting and analysis system, XReport can generate business-oriented reports on request of business analytical requirements. However, the Brio-based analytical system can only create reports based on data model predefined in the data warehouse, ..., as a result, it is difficult for it to generate resulting reports in the format and with the content as required by business and operations...”

7 Conclusions and future work

Integration and implementation of business intelligence has become a popular solution for promoting scientific and efficient decision making in many industries. However, the existing products and solutions mainly focus and rely on structural integration. This has shown to be ineffective in handling ubiquitous complexities in the real world, especially semantic heterogeneity. The work on semantic interoperability in the data warehousing environment is very challenging but necessary. The challenge comes from the multiform semantic conflicts and the inconsistency of multiple domains.

In this paper, we have proposed ontology-based approach to explicitly deal with semantic heterogeneity which is widespread in integrating BI. A series of ontological engineering techniques have been developed to make the semantic integration and interoperability realizable. An ontology-based integrated BI system encompassing mechanisms for both of them present a much stronger power for information integration user-friendly and flexibly from the following aspects.

- A BI system cannot be built by simply packing DW, OLAP and DM on the basis of a DW, rather through an internal ontology-based linkage and communication channel. This hybrid ontology space sets up some internal mechanisms for supporting ontology mapping, query

---

2 Sponsored by China Innovation Fund for Small Technology Based Firms 04C26211100957.
parsing among business concepts, analysis models and physical entities matching with the existing subsystems of DW and/or EIS.

- A series of relevant ontological engineering techniques make the ontology-based BI system achievable in the real world. These include ontology naming, representation, semantic relationships, ontology mapping, aggregation and transformation and query intra or inter the relevant ontological domains.
- A collection of semantic rules have been designed to make it possible for the automatic aggregation and mapping of semantic relationships and ontological items, and mapping and query of ontological items intra or inter domains.
- A unified knowledge portal that includes a set of user-friendly supports is recommended to help users to modify, update, create or re-arrange ontologies and functionalities on different levels in terms of specific problem domain and changing requirements. Thus, users can arrange their own ontology base and ontology namespace as required adaptively without modifying the match relations.

We have built some real-world prototype systems in terms of the above theory. Our experiments and customer feedbacks have shown that the ontology-based warehousing and BI system are valuable for supporting more user-friendly, flexible, adaptive and automatic integration and analysis of business intelligence in the real world. This work is worthy of further systematic research and development in real-world environment on top of existing DW, OLAP, DM and reporting systems.

References

[8] www.cognos.com
[10] www.microstrategy.com
[22] L.B. Cao, et. al., Ontology Transformation in Multiple Domains. 17th Australian Joint Conference on AI, G.J. Web and Xinghao Yu (Eds.): LNAI 3339, pp. 985–990, 2004