

# FCA assisted IF Channel Construction towards Formulating Conceptual Data Modeling

YANG WANG AND JUNKANG FENG

E-Business Research Institute

Business College, Beijing Union University, China

Database Research Group

School of Computing, University of Paisley

University of Paisley, Paisley, Scotland, UK, PA1 2BE

yang.wang @paisley.ac.uk

junkang.feng@paisley.ac.uk

*Abstract:* In this paper we explore how IF (Barwise-Seligman's information flow theory) [3] and FCA (Formal Concept Analysis) [20] may be used to help conceptual data modeling. A fundamental observation we make is that the process of conceptual data modeling is a distributed system in the sense of Barwise-Seligman's information flow theory, i.e., the Channel theory [3], and therefore the process of conceptual data modeling per se, the correctness of it, and thus the usefulness of the resultant database depends solely on the existence of a relevant information channel (hereafter IF channel) that takes the real world domain being modelled and the conceptual data model as components. Based upon this conviction, we explore how such an IF channel may be constructed and how it guides conceptual data modeling. The main findings through this work are as follows

- Using IF and FCA for conceptual data modeling, domain-dependent knowledge is still needed in establishing a set of original correspondences between objects in the real world and those in a conceptual data model. This constitutes the basis for further modeling work;
- IF and FCA help the modeling in that all those that are required by the existence of an IF channel guide the process of modeling and eventually determine the correctness of it. This includes:
  - Helping check the correctness of the domain-dependent knowledge reflected in the original set of correspondences;
  - Guiding the solicitation of knowledge about the domain from the domain expert throughout the process of modeling.

We use a simple example to illustrate our points.

*Key-Words:* Conceptual Modeling, Information Flow, Infomorphisms, Concept Lattice, FCA

## 1 Introduction

It is easily observable that conceptual data modeling for information systems is still conducted largely intuitively. Information systems are those computer based systems that are characterized as being data-intensive, transaction-oriented, with a substantial element of human-computer interaction [14]. Data modeling is a process of understanding the 'information aspect' of a domain of interest, for which an information system is developed, so it plays a vital role in the development of information systems. For example, Structured Systems Analysis and Design Methodology (SSADM) takes three perspectives in analyzing the 'current environment', namely DFD, LDS (Logical Data structure) and Entity life history [1], [19]. SSADM places a great emphasis on data modeling, as Weaver put it, the Logical Data Model 'is possibly the most important and ultimately the most rigorous product of an entire SSADM project', and the Logical Data Model 'is a vehicle for analyzing the logical structure of an organization's information' [19]. Another example is the Unified Modeling Language (UML) [18]. It uses a set of diagrams including use case, static structures, behaviour and implementation to ensure the quality of software development that meets the business requirements.

Conceptual data modeling is carried out by using high level (i.e., human level, in contrast to machine level) data models, widely-known ones of which include the entity-relationship data model [7], the enhanced entity-relationship data model [7], and the extended relational data model - RM/T [5].

A conceptual data modeling process is normally started with entity identification, which is carried out in a 'just do it' manner. Taking SSADM as an example, with SSADM version 4, when LSD is developed, the first step is entity identification, followed by relationship identification and so on. Obviously, without entities

identified, relationships have no ground to exist. With RM/T, the first step is to identify kernel entities. With the entity-relationship data model, a conceptual data modeling process starts with the identification of regular entities. So, the concept entity is vital to information systems development and conceptual data modeling, but its definition in the literature appears vague and general. It is taken as 'a 'thing' in the real world with an independent existence' [9]; or 'a real world object which is distinctly identifiable' [5] or 'any object or concept about which a system needs to hold information' [19]. These definitions often cause confusions among novices of information systems and databases, who feel that entities seem everywhere and difficult to know where to start and where to stop. Thus the above definitions of entity can hardly serve as an adequate guideline. On the other example, i.e., UML, although its class diagram has the great flexibility on either including implementation details or connecting the whole analysing process with entity relationship model, the evaluation of any 'product' of such an analysis still involves a great deal of intuition. Identified business rules are validated mainly by face-to-face communications despite the fact that some researcher tries to ease this step by remedying UML with fact-oriented approach such as object role modeling (ORM) [13].

Therefore, conceptual data modeling poses as a difficult task, like a 'black art', full of informality and uncertainty considering drawbacks of entity identification and the lack of effective support of validation methods. It results in our understanding of and methods for conceptual data modeling as a whole being unsatisfactory. This appears due to our understanding of data, information and information systems being still somewhat shallow and patchy even though twenty years have elapsed since Bubenko and Olive's observation in 1986 [4] that the field was immature.

Therefore any formulation and validation of conceptual data modeling for information systems with sound theoretical underpinnings would seem desirable, and the issue of whether and how this might be done is an interesting research question.

A fundamental observation we make is that the process of conceptual data modeling is a distributed system in the sense of Barwise-Seligman's information flow theory, i.e., the Channel theory [3], and therefore the process of conceptual data modeling per se, the correctness of it, and thus the usefulness of the resultant database depend solely on the existence of a relevant information channel (hereafter IF channel) that takes the real world domain being modelled and the conceptual data model as components. Based upon this conviction, we explore how such an IF channel may be constructed and how it guides conceptual data modeling. In this paper, some ideas on FCA assisted IF channel construction from the real world context [20] to some certain conceptual data model context will be presented. The thoughts are described in terms of two semiotic levels, i.e., the syntactic and the semantic level [17]. By the 'the syntactic level of the construction of an IF channel for conceptual data modeling', we mean that the syntactic rules for data modeling and for IF channel construction are used for a particular modeling task. This is one hand. On the other hand, 'the semantic level of the construction of an IF channel for conceptual data modeling' is concerned with how the correspondences between objects in the real world domain and those in a data model are established.

An IF channel for conceptual data modeling is built upon the understanding and knowledge about the syntactic rules of conceptual data modeling and IF channel construction of the person (i.e., the modeller) that carries out the modeling. We assume that the modeller has the ability to map the real world objects to a certain set of conceptual objects and data. Therefore, the IF channel is 'conceptual' in that it formulates something in his or her mind, and this process is not unlike the process of 'metaphor'.

How the syntactic rules for data modeling and for the construction of an IF channel are used for a particular modeling task would depend on the establishment of concrete correspondences between real world objects and objects within a data model, which are represented by data types and instances. These correspondences (mappings) constitute the semantic level of the construction of the IF channel. Moreover, semantic correspondences are presented by the formal structure of the IF channel and the conceptual data model, which comply with the syntactic rules for IF channels and data models. Thus the two levels are interrelated with and depend on each other. In the sections that follow, we will use an example to show our points.

## **2 The Syntactic Level of the IF Channel Construction**

On this level, the construction of an IF channel involves representing some objects (together with relations) in one context by using certain set of objects in a different context. Here, we are especially interested in how to establish a channel that connects real world objects in the 'source' context (Terminology 1) with modeling objects in the 'target' context (Following Barwise-Seligman's IF theory, the term 'source' refers to representations (Terminology 10), which in our case is the conceptual data model, and the term 'target' the real

world domain being modelled). As aforementioned, the modeling process is unlike the process of using metaphor, which is described as a ‘mapping from one situation to another which is governed by the constraints of structural consistency and one-to-one mapping’ [12]. Our work on the construction of an IF channel (Terminology 3) on this level is inspired by the findings of Old and Priss [15] from their work on an IF based formal model of metaphor. We use concept lattice (Terminology 5) to describe the relations within one context (the same as ‘classification’ (Terminology 2) [3]) and use infomorphisms (Terminology 3) to describe the information flow between different contexts. The IF channel conceptually links up the source and target contexts and facilitates information transmission between the target and the source. Infomorphisms that are established at some point of time in modeling between these contexts capture the correspondences between objects that are known at that point, which constitute an IF channel. Such a channel captures information flow between the involved contexts.

Old and Priss [15] introduce a definition of ‘relational infomorphisms’, which enables more complex modeling of metaphor and extends Barwise-Seligman’s information flow theory:

*Relational Infomorphisms: a pair of functions  $(f, f')$  which maps between two classifications,  $(A1, B1, R1)$  and  $(A2, B2, R2)$ , which have further relations  $R11, R12, \dots \subseteq A1 \times B1$  and  $R21, R22, \dots \subseteq A2 \times B2$ , such that for elements  $b1, b2 \in B2$  and for each pair of corresponding relations  $(R1i, R2i)$ :  $b1R2ib2 \iff f'(b1)R1if(b2)$ .*

Based on this definition, we observe that conceptual data modeling is seen as an IF channel that is constructed with relational infomorphisms that connect relations between two contexts. For example, in an example regarding ‘furniture’ in a room, there is an IF channel shown in figure 1.

In this diagram, a real world context, i.e., the room containing furniture (the room context), and a model context (the DB context), i.e., a DB conceptual structure with a table of data, are connected by an IF channel, which is constructed based upon people’s perceptions about the link between the two. Infomorphisms map each concept (Terminology 4) from the room context and the DB context to a corresponding concept of the IF channel (more precisely, the core of the IF channel). The core of the IF channel here is represented by a concept lattice [20], which describes the common abstract conceptual characteristics of both the room context and the DB context.

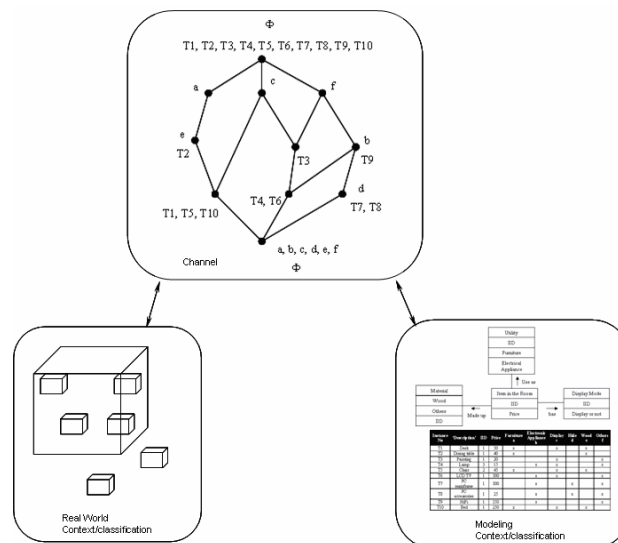


Fig.1 Modeling Procedure Represented by IF channel

The infomorphisms are relational infomorphisms because the concept relation hierarchies of both contexts are mapped. The concept relations of the room context are mapped into the concept hierarchy of the IF channel, which is in turn mapped into the DB context. It should be noted that these infomorphisms are not just instantiate functions from the core of the IF channel to each context. Infomorphisms preserve certain information but an IF channel also needs each context to provide additional structural consistency constraints [12]. Successfully achieving the syntactic level of an IF channel construction means that the source context and the target context are informationally related in a certain way through the understanding of the modeller. However, the syntactic level gives little concern to how these two contexts (or even more contexts) are informationally connected, and yet this relationship plays a central role in the whole process of modeling. This is the reason why we need to know in details how semantically the two contexts are connected by the channel.

### 3 The Semantic Level of the IF Channel Construction

On this level, we will show how concept lattice assists the finding of mappings for the IF channel. In the example, the room context consists of a set of real world objects including desk, dining table, painting, lamp, chair, LCD TV, PC mainframe, PC accessories, HiFi and bed. A modeller might model them with conceptual objects and relations in a database, i.e., a DB context. The DB context could be shown in figure 1 and table 1.

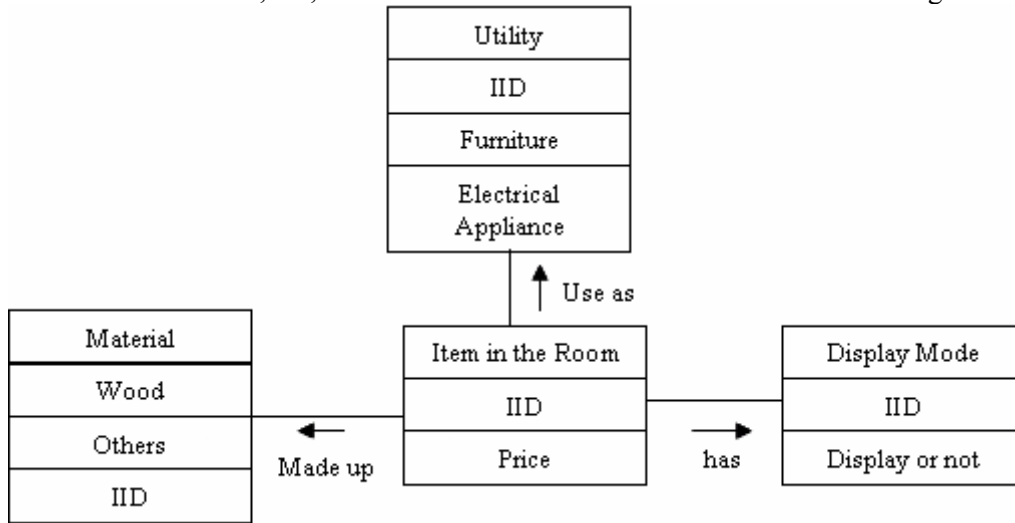


Fig.2 Conceptual Model of DB Context

Instance No	'Description'	IID	Price	Furniture a	Electronic Appliance b	Display c	Hide d	Wood e	Others f
T1	Desk	1	50	x		x		x	
T2	Dining table	1	40	x				x	
T3	Painting	1	20			x			x
T4	Lamp	3	15		x	x			x
T5	Chair	2	45	x		x		x	
T6	LCD TV	1	300		x	x			x
T7	PC mainframe	1	300		x		x		x
T8	PC accessories	1	25		x		x		x
T9	HiFi	1	350		x				x
T10	Bed	1	250	x		x		x	

Table 1. The DB context Table

Through the analysis on the syntactic level, we know that due to the prior knowledge of modeling of the modeller, he/she might model those identified characteristics of each real world objects as some corresponding attributes of the DB. The mapping is one-to-one. We take attributes 'Furniture', 'Electronic Appliance', 'Display', 'Hide', 'Wood' and 'Others' as an example, and we would have the source and target contexts defined as follows:

Source Context - Room Context  $A_R(A1, B1, R1)$ :

Tokens A1 – real world objects: desk, dining table, painting, lamp, chair, LCD TV, PC mainframe, PC accessories, HiFi and bed,

Types B1 – characteristics for classifying tokens (in modeller's mind): Furniture, Electronic Appliance, Display, Hide, Wood, and Others,

Relations R1 – Relations between token A1 and B1.

Target Context – DB Context  $A_{DB}(A2, B2, R2)$ :

Tokens A2 – DB instances: T1, T2, T3, T4, T5, T6, T7, T8, T9, T10,

Types B2 – DB attributes: ‘Furniture’, ‘Electronic Appliance’, ‘Display’, ‘Hide’, ‘Wood’ and ‘Others’,  
 Relation R2 – ‘belongs to’ relation in DB.

To find the informational relationship between the objects in the two contexts, following the rules of the IF channels, we need to set up infomorphisms that connect each context with the core of the IF channel. On the type level, the correspondences between some types of the two contexts are established based upon the perception of the modeller that is involved in the modeling process per se or in the use of the modeling. Such perception is indeed found on the syntactic level of IF channel.

Although the modeler may carry out the modeling process incrementally, she/he understands and can use the types of the real world domain and the attributes in a database as belonging to a theoretical domain of discourse T. This is mathematically captured by a classification A with T as its type set. For example, a, b, c, d, e, f are types of A, and they are translated (mapped) to types of the source context and the target context respectively. This is part of type level function of their respective infomorphisms.

$$\begin{aligned}
 g_R^*(a) &= \text{Furniture}; g_{DB}^*(a) = \text{‘Furniture’}; \\
 g_R^*(b) &= \text{Electronic Appliance}; \\
 g_{DB}^*(b) &= \text{‘Electronic Appliance’}; \\
 g_R^*(c) &= \text{Display}; g_{DB}^*(c) = \text{‘Display’}; \\
 g_R^*(d) &= \text{Hide}; g_{DB}^*(d) = \text{‘Hide’}; \\
 g_R^*(e) &= \text{Wood}; g_{DB}^*(e) = \text{‘Wood’}; \\
 g_R^*(f) &= \text{Others}; g_{DB}^*(f) = \text{‘Others’}.
 \end{aligned}$$

The types are associated with a classification. Such a classification, say A, is holding the rules in terms of partial alignments, which govern the way of the two contexts being related. Classification A would not add any additional semantic information as it does not comprise any IF theory (namely no constraints showing on it). Therefore, with types defined as a, b, c, d, e, f, we generate all the possible tokens for the classification of A. Note that as all the possible tokens are generated, there are no embedded constraints available on this classification, that is, no constraints are placed on the possibilities of the existence of the tokens in terms of how they may belong to types (Terminology 9).

	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>	<b>e</b>	<b>f</b>
n0	0	0	0	0	0	0
n1	0	0	0	0	0	1
...	...	...	...	...	...	...
n9	0	0	1	0	0	1
...	...	...	...	...	...	...
n17	0	1	0	0	0	1
...	...	...	...	...	...	...
n21	0	1	0	1	0	1
...	...	...	...	...	...	...
n25	0	1	1	0	0	1
...	...	...	...	...	...	...
n34	1	0	0	0	1	0
...	...	...	...	...	...	...
n42	1	0	1	0	1	0
...	...	...	...	...	...	...
n63	1	1	1	1	1	1

Table2. The IF Channel Classification

With types a, b, c, d, e, f, the classification A has totally 64 tokens as shown in above table. To satisfy the fundamental property of infomorphisms (Terminology 3) [3], the token level functions of the infomorphisms,  $g_R$  and  $g_{DB}$ , have to be:

$$\begin{aligned}
&g_R^*(\text{desk}) = n42; g_{DB}^*(T1) = n42; \\
&g_R^*(\text{dining table}) = n34; g_{DB}^*(T2) = n34; \\
&g_R^*(\text{painting}) = n9; g_{DB}^*(T3) = n9; \\
&g_R^*(\text{lamp}) = n25; g_{DB}^*(T4) = n25; \\
&g_R^*(\text{chair}) = n42; g_{DB}^*(T5) = n42; \\
&g_R^*(\text{LCD TV}) = n25; g_{DB}^*(T6) = n25; \\
&g_R^*(\text{PC mainframe}) = n21; g_{DB}^*(T7) = n21; \\
&g_R^*(\text{PC accessories}) = n21; g_{DB}^*(T8) = n21; \\
&g_R^*(\text{HiFi}) = n17; g_{DB}^*(T9) = n17; \\
&g_R^*(\text{bed}) = n42; g_{DB}^*(T10) = n42;
\end{aligned}$$

After achieving classification A, we can find the desired IF channel accordingly. This includes constructing the IF channel classification, i.e., the classification that is the core of the channel (Terminology 7) and infomorphisms:  $f_R: VA_R \rightleftarrows C$  and  $f_{DB}: VA_{DB} \rightleftarrows C$ , where  $VA_R$  and  $VA_{DB}$  are the distinctive power of  $A_R$  and  $A_{DB}$  respectively. The distributed system is shown as follows.

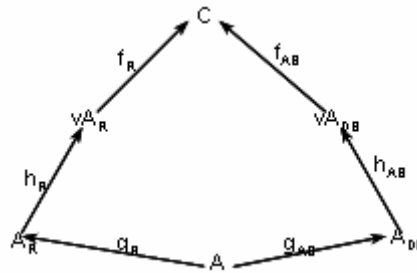


Fig.3 Distributed System

	{'Furniture', 'Electronic Appliance', 'Display', 'Hide', 'Wood', 'Others'}	...	{'Furniture', 'Display', 'Wood'}	...	{'Display', 'Electronic Appliance', 'Others'}	...
T1	1		1			
T2	1					
T3	1					
T4	1				1	
T5	1		1			
T6	1				1	
T7	1					
T8	1					
T9	1					
T10	1		1			

Table 3. The table fragment of  $VA_R$  classification

	{Furniture, Electronic Appliance, Display, Hide, Wood, Others}	...	{Furniture Display, Wood}	...	{Display, Electronic Appliance, Others}	...
desk	1		1			
dining table	1					
painting	1					
lamp	1				1	
chair	1		1			
LCD TV	1				1	
PC mainframe	1					
PC accessories	1					
HiFi	1					
bed	1		1			

Table 4. The table fragment of  $VA_{DB}$  classification

$VA_R$  is defined as disjoint union of classification  $A_R$ , i.e., the types are disjoint union of  $A_R$  while tokens are the same as  $A_R$  with corresponding classification relations.

There are natural infomorphisms  $h_R$  and  $h_{AB}$  which connect  $A_R$  and  $A_{DB}$  with  $VA_R$  and  $VA_{DB}$  respectively. The classification  $C$  associated with the IF channel is constructed such that The type set of the IF Channel is the disjoint union of types of  $VA_R$  and  $VA_{DB}$ ; the token set is the connections (pair of tokens) that connect a token from  $VA_R$  with a token from  $VA_{DB}$  only when the two tokens are sent by the infomorphisms  $g_R$  and  $g_{DB}$  to the tokens of the classification  $A$  that are classified as of the same type. For example, the core  $C$  will have the token  $\langle \text{bed}, T10 \rangle$  connecting  $VA_R$ 's token bed with  $VA_{DB}$ 's token T10 because  $g_R * (\text{desk}) = n42$ ;  $g_{DB} * (T1) = n42$ ;

The following is a fragment of the IF channel classification on the core.

	{a}	{c}	{f}	{a,e}	{c,f}	{f,b}	{a,c,e}	{c,f,b}	{f,b,d}
$\langle \text{dining table}, T2 \rangle$	1			1					
$\langle \text{desk}, T1 \rangle$	1	1		1			1		
$\langle \text{desk}, T5 \rangle$	1	1		1			1		
$\langle \text{desk}, T10 \rangle$	1	1		1			1		
$\langle \text{chair}, T1 \rangle$	1	1		1			1		
$\langle \text{chair}, T5 \rangle$	1	1		1			1		
$\langle \text{chair}, T10 \rangle$	1	1		1			1		
$\langle \text{bed}, T1 \rangle$	1	1		1			1		
$\langle \text{bed}, T5 \rangle$	1	1		1			1		
$\langle \text{bed}, T10 \rangle$	1	1		1			1		
$\langle \text{painting}, T3 \rangle$		1	1		1				
$\langle \text{lamp}, T4 \rangle$		1	1		1	1		1	
$\langle \text{lamp}, T6 \rangle$		1	1		1	1		1	
$\langle \text{LCD TV}, T4 \rangle$		1	1		1	1		1	
$\langle \text{LCD TV}, T6 \rangle$		1	1		1	1		1	
$\langle \text{HiFi}, T9 \rangle$					1	1			
$\langle \text{PC mainframe}, T7 \rangle$			1			1			1
$\langle \text{PC mainframe}, T8 \rangle$			1			1			1
$\langle \text{PC accessories}, T7 \rangle$			1			1			1
$\langle \text{PC accessories}, T8 \rangle$			1			1			1

Table 5. The table fragment of the IF channel classification on the core

As aforementioned, the modeller models the characteristics of the real world objects as the conceptual attributes. Determined by modeller's knowledge of modeling, this mapping is one-to-one, i.e., each identified feature used to classify the real world's objects is reflected by a corresponding DB attribute. Therefore, in the above table, we do not show the disjoint union type sets as two separate sets of types because they mirror each other. We can produce the concept lattice for the IF channel classification as follows.

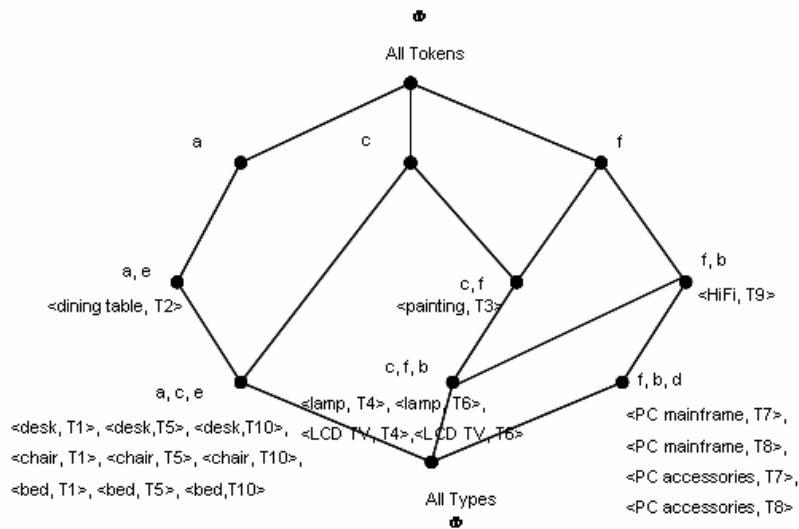


Fig.4 The IF Channel Concept Lattice

It can be seen that the concept hierarchy of this lattice is the same as the one developed in the modeller's mind on syntactic level. Although the tokens become pairwise due to the construction of the IF channel, the concepts and the relations among them are not changed. This reflects the procedure of modeling according to the modeller's prior knowledge. The most important thing is that the structure of the concept lattice is preserved throughout the two levels. It is this lattice that represents what should happen in the modeller's mind. It shows the interrelationship between the two levels of modeling.

Another thing that is worth noting is that according to the modeller's perception at this particular time (Different modeller may hold different perceptions even they share the same knowledge and the same modeller may hold different perceptions at different times. Therefore, here, we emphasise the 'particular time'), some tokens are retained in the same concepts. In other words, they belong to the same set of types. Hence, if we turn to consider some particular instances of relations, for example, the relation 'an object being placed on another object', we would not be able to get this instance using the available concept hierarchy as the instances of a concept are not distinguishable [10] from others.

#### 4. Information Flow that shows the Correctness of the Modeling

We have constructed an IF channel as a mathematical model of a conceptual data modeling process. This channel captures the regularities that govern how the real world context is informationally related to the model context. The regularities are reflected in the information flow within the IF channel. That is to say, if there is a right information flow from the real world context to the model context, the modeling process has been conducted correctly. The information flow is captured by the constraints over the distributed system that is mathematical models for the real world context and the model contexts respectively.

In the example, when we set up the infomorphisms between the room and DB context, we stressed that given the correspondences between the types, for the information flow to be ever possible, the tokens have to correspond in a certain way. We have shown, and we re-iterate here in order to make it easier to explain our points, that on the type level, each identified feature used to classify the real world objects is modelled with a corresponding DB attribute, which constitute the type level function of the infomorphisms. The correspondences that constitute this function are as follows:

$$\begin{aligned}
 g_R^*(a) &= \text{Furniture}; & g_{DB}^*(a) &= \text{'Furniture'}; \\
 g_R^*(b) &= \text{Electronic Appliance}; \\
 g_{DB}^*(b) &= \text{'Electronic Appliance'}; \\
 g_R^*(c) &= \text{Display}; & g_{DB}^*(c) &= \text{'Display'};
 \end{aligned}$$

$$\begin{aligned} g_R^*(d) &= \text{Hide}; g_{DB}^*(d) = \text{'Hide'}; \\ g_R^*(e) &= \text{Wood}; g_{DB}^*(e) = \text{'Wood'}; \\ g_R^*(f) &= \text{Others}; g_{DB}^*(f) = \text{'Others'}. \end{aligned}$$

Now in order to make the information flow ever possible, infomorphisms have to exist between the components of the IF channel, i.e., between the real world context and the core of the IF channel, and the data model context and the core of the IF channel respectively. To this end, the fundamental property of infomorphisms has to be satisfied. Therefore, the correspondences between the tokens have to be defined as follows:

$$\begin{aligned} g_R^*(\text{desk}) &= n42; g_{DB}^*(T1) = n42; \\ g_R^*(\text{dining table}) &= n34; g_{DB}^*(T2) = n34; \\ g_R^*(\text{painting}) &= n9; g_{DB}^*(T3) = n9; \\ g_R^*(\text{lamp}) &= n25; g_{DB}^*(T4) = n25; \\ g_R^*(\text{chair}) &= n42; g_{DB}^*(T5) = n42; \\ g_R^*(\text{LCD TV}) &= n25; g_{DB}^*(T6) = n25; \\ g_R^*(\text{PC mainframe}) &= n21; g_{DB}^*(T7) = n21; \\ g_R^*(\text{PC accessories}) &= n21; g_{DB}^*(T8) = n21; \\ g_R^*(\text{HiFi}) &= n17; g_{DB}^*(T9) = n17; \\ g_R^*(\text{bed}) &= n42; g_{DB}^*(T10) = n42. \end{aligned}$$

As the correctness of the modeling lies with the required information flow, none of the above can be violated. This is because any violation of the above specified correspondences would result in the non-existence of the required information flow.

In addition to the functional correspondences shown that are required by the fundamental property of infomorphism, there is another part on the channel, which is significant for the modeling and the correctness of it. This is the constraints of ‘regular theory’ (Terminology 6) [3] on the core of the IF channel. For our example, we identify the following constraints that constitute the regular theory on the core of the IF channel:

$$e \vdash a; b \vdash f; d \vdash b; d \vdash f; \vdash a, c, f; a, c, f \vdash.$$

These constraints make the entailment relationships between each identified types (in terms of concepts) clearly to be seen. They clarify the inheritance hierarchy of concepts on the IF Channel which in deed reflects the modeller’s understandings during the modeling process. Like the process of metaphor, these constraints represent the abstraction of common characteristics of both real world and DB context. Constraints capture what information flows. To show the information flow from the real world context to the model context, we translate, by using the Elimination rule (Terminology 8) [3] of the relational infomorphisms, these constraints are moved to distributed systems (we name them as distributed constraints), that are models of the real world context and the model context.

On real world context:

$$\begin{aligned} &\text{Wood} \vdash \text{Furniture}; \\ &\text{Electronic Appliance} \vdash \text{Others}; \\ &\text{Hide} \vdash \text{Electronic Appliance}; \\ &\text{Hide} \vdash \text{Others}; \\ &\vdash \text{Furniture, Display, Others}; \\ &\text{Furniture, Display, Others} \vdash. \end{aligned}$$

On DB context:

$$\begin{aligned} &\text{'Wood'} \vdash \text{'Furniture'}; \\ &\text{'Electronic Appliance'} \vdash \text{'Others'}; \\ &\text{'Hide'} \vdash \text{'Electronic Appliance'}; \\ &\text{'Hide'} \vdash \text{'Others'}; \\ &\vdash \text{'Furniture', 'Display', 'Others'}; \\ &\text{'Furniture', 'Display', 'Others'} \vdash. \end{aligned}$$

Following the third principle of information flow put forward by Barwise and Seligman [3] it is the particulars (i.e., tokens) that carry information. In the example, these particulars are indeed the token pairs

(shown in figure 3) which are determined by the infomorphisms  $h_R \circ f_R$ , and  $h_{DB} \circ f_{DB}$ . This also means that conform to the mode of information flow, we would get the induced view of what should be inside of the real world and DB context. In terms of concept lattice, they are shown as follows respectively.

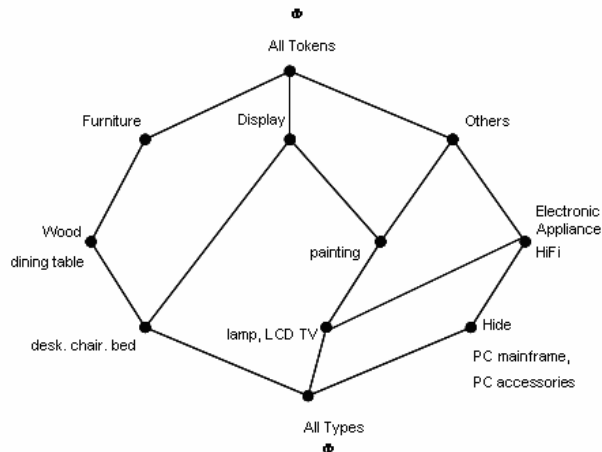


Fig.5 Concept Lattice of Induced Real World Context

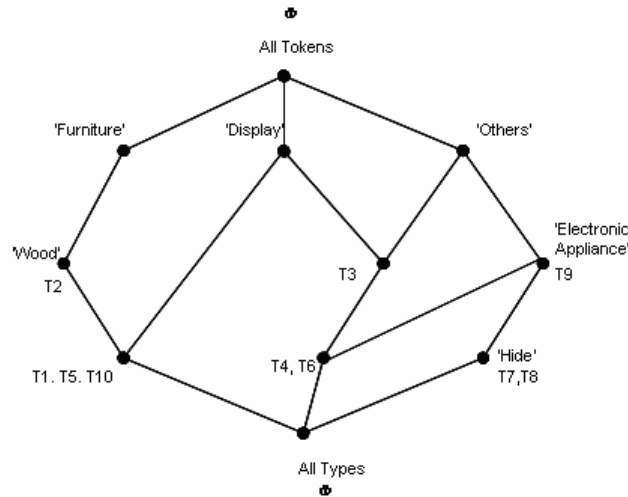


Fig.6 Concept Lattice of Induced DB Context

Therefore, we see that the induced real world and DB context are holding the distributed constraints produced earlier. Also, the tokens are corresponding ones in each pairwise token. Now, if we turn to the original real world and DB contexts without considering the IF Channel, we could possibly get the different concept lattices respectively. If differences are recognized, it means that the IF Channel does not reflect the correct or complete informational relations between the two contexts. As a result, the modeling process is not succeeded and completed.

For the conditions of incomplete modeling, constraints on the IF Channel are dynamically changeable along with the development of the understanding of the real world domain being modelled. This is what normally happens in the data modeling for a database. Accordingly the IF channel evolves and at the same time the requirements for the existence of the channel continue providing guidance to the modeling process and justifying the correctness of it.

In summary, with our approach, the syntactic level of the IF Channel based on relational infomorphisms are fulfilled by the semantic level of the IF Channel based on detailed infomorphisms i.e.,  $h$ ,  $f$ , and  $g$ . The semantic level of constraints, if we make them clear, they are

$$\begin{aligned} \text{Furniture} \vdash \text{'Furniture'}; \text{Electronic Appliance} \vdash \text{'Electronic Appliance'}; \text{Display} \vdash \text{'Display'}; \\ \text{Hide} \vdash \text{'Hide'}; \text{Wood} \vdash \text{'Wood'}; \text{Others} \vdash \text{'Others'}. \end{aligned}$$

They are fostering the distribution of constraints on the syntactic level. The information flow that is captured by the regular theory and token connections by the infomorphisms on the core of the IF channel verifies the correctness of the modeling.

## 5. Some General Implications of this Work

We believe that qualitative (also called semantic) information theory represented by the work of Dretske [8], Barwise and Seligman [3], Devlin [6] and Barwise and Perry [2] and information philosophy by Floridi [11] represent the most advanced knowledge thus far regarding information and information flow. This should provide sound theoretical underpinnings and insights for tackling some tough and elusive problems, such as information creation and information flow in the context of information systems and the correctness and formulation of conceptual data modeling. To this end, there are two questions of a general nature that have been ‘bothering’ us, namely:

- How a general theory of information flow might help understand how information creates and information flow takes place within a particular domain, for example, information systems, which in turn help understand the domain.
- On the other hand, to construct an IF channel, a layer that might be called ‘domain-dependent foundational layer’ which is corresponding to the syntactic level in this paper under the IF channel, which enables information flow to take place, would seem needed.

Is this a ‘chicken and egg’ problem? The work reported within this paper represents a step toward finding a convincing answer to this question. Through this work, we find:

- That domain knowledge is still needed in establishing an original set of correspondences between objects in the real world and those in a conceptual data model. This constitutes the basis for any further modeling work by using IF and FCA. That is to say, the aforementioned ‘domain-dependent foundational layer’ under the IF channel is necessary, which would determine how a system works, which also reflect syntactically as information flow between its components. Information flow in the sense that we could know something about B by looking at A, does exist, but it is a result of regularities that govern the working of the system.
- That IF and FCA do help conceptual data modeling and database design in that all those that are required by the existence of an IF channel guide and eventually determine the process of conceptual data modeling. This is because the construction of database or an information system in general is essentially a job of using a set of ‘representations’, which are external objects that we use to present information about some other objects on the basis of some fixed semantic rules [16] to represent objects in the modelled real world domain. This, in terms of IF, is a matter of making sure that there exists an IF channel in which information flows from the real world context to the data model and database context. That is to say, conceptual data modeling, database design and information systems construction is, at the heart of the matter, a problem of IF channel construction in certain information flow.

### References:

- [1]. C. Ashworth and M. Goodland, *SSADM - A Practical Approach*, McGraw-Hill, London, 1990.
- [2]. J. Barwise and J. Perry, *Situations and Attitudes*, Cambridge, Mass.: Bradford-MIT, 1983.
- [3]. J. Barwise and J. Seligman, *Information Flow: the Logic of Distributed Systems*, Cambridge University Press, Cambridge, 1997.
- [4]. J.A. Bubenko and A. Olive, *Dynamic or temporal modeling- an illustrative comparison*, Syslab working paper 117, Syslab, Univ. of Stockholm, Stockholm, 1986.
- [5]. C.J. Date, *An Introduction to Database Systems*, Sixth edition. Addison-Wesley Publishing Company, Reading, 1995.
- [6]. K. Devlin, *Logic and information*. Cambridge University Press 1991.
- [7]. P.P. Chen, *The entity relationship model -- Toward a unified view of data*, TODS, 1:1, March, 1976.
- [8]. F.I. Dretske, *Knowledge and the flow of information*. Basil Blackwell, Oxford, (1981) 1999.
- [9]. R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*. The Benjamin/Cummings Company, Inc., Redwood City, California, 1994.
- [10]. J. Feng, Conditions for Information Bearing Capability, *Computing and Information Systems Technical Reports No 28*, University of Paisley, ISSN 1461-6122, 2005.
- [11]. L. Floridi, Information. In Floridi L (eds) *Philosophy of computing and information*. Blackwell Publishing Ltd, 2004.
- [12]. D. Gentner and K. Forbus, MAC/FAC: A Model of Similarity-based Retrieval. *Proceedings of the Cognitive Science Conference*, pp. 504-509, 1991.

- [13].T. Halpin, UML Data Models from an ORM Perspective: Part 1 The Journal of Conceptual Modeling, 1999.
- [14].P. Loucopoulos and R. Zicari ed, Conceptual Modeling, Databases, and CASE, John Wiley & Sons, New York, 1992.
- [15].L.J. Old and U. Priss, Metaphor and Information Flow. In: Proceedings of the 12th Midwest Artificial Intelligence and Cognitive Science Conference, 2001, pp. 99-104, 2001.
- [16].A. Shimojima, On the Efficacy of Representation, Ph.D. Thesis. The Department of Philosophy, Indiana University, 1996.
- [17].R. Stamper, Organizational Semiotics. In Information Systems: An Emerging Discipline?, (J. Mingers and F.A. Stowell), McGraw-Hill, London, 1997.
- [18].UML Partners UML Semantics, version 1.1, OMG document ad/97-08-04, ad/97-08-05, ad/97-08-08, 1997,
- [19].P.L. Weaver, Practical SSADM Version 4, Pitman, London 1993.
- [20].R. Wille, Introduction to Formal Concept Analysis. In G. Negrini (Ed.), Modelli e modellizzazione. Models and modeling. Consiglio Nazionale delle Ricerche, Istituto di Studi sulli Ricerca e Documentazione Scientifica, Roma, pp.39-51, 1997.

### Terminology

- 1) A *formal context* is a triple  $(G, M, I)$ . If  $G$  and  $M$  are sets and  $I \subseteq G \times M$  is a binary relation between  $G$  and  $M$ . The elements of  $G$  are usually called objects and the elements of  $M$  attributes.
  - 2) A *classification* is a structure  $A = \langle U, \Sigma_A, \vdash_A \rangle$  where  $U$  is the tokens of  $A$ ,  $\Sigma_A$  the types of  $A$  used to classify the tokens, and  $\vdash_A$  is their binary relations.
- Information channel* consists of an indexed family  $C = \{f_i: A_i \leftrightarrow C\}_{i \in I}$  of infomorphisms with a common co-domain  $C$ , the core of the channel. Let  $A$  and  $C$  to be two *classifications*. An *infomorphism* between them is a pair  $f = \langle f^*, f_* \rangle$  of functions. For all tokens  $c$  of  $C$  and all types  $\alpha$  of  $A$ , it is true that  $f^*(c) \vdash_A \alpha$  iff  $c \vdash_{Cf_*} \alpha$ . This rule is also called as '*fundamental property of infomorphisms*'.
- 3) A *formal concept* of  $\mathbf{K} = (G, M, I)$  is defined as a pair  $(A, B)$  where  $A \subseteq G$ ,  $B \subseteq M$  and  $A^\uparrow = B$  and  $B^\downarrow = A$  where  $A^\uparrow$  is the set of common attributes of  $A$ , formally described as  $A^\uparrow := \{m \in M \mid \forall g \in A \ g I m\}$  and  $B^\downarrow$  is the set of common objects of  $B$ ,  $B^\downarrow := \{g \in G \mid \forall m \in B \ g I m\}$ .  $A$  is called the *extent* and  $B$  the *intent* of  $(A, B)$ .
  - 4) *Concept Lattice*: The set of all formal concepts of  $\mathbf{K}$  is denoted by  $\mathbf{B}(\mathbf{K})$ . The conceptual hierarchy among concepts is defined by set inclusion: For  $(A_1, B_1), (A_2, B_2) \in \mathbf{B}(\mathbf{K})$  let  $(A_1, B_1) \leq (A_2, B_2) : \Leftrightarrow A_1 \subseteq A_2$  (which is equivalent to  $B_2 \subseteq B_1$ ).
  - 5) *Theory*  $T = \langle \text{typ}(T), \vdash \rangle$  consist of a set  $\text{typ}(T)$  of types, and a binary relation  $\vdash$  between subsets of  $\text{typ}(T)$ . Pairs  $\langle \Gamma, \Delta \rangle$  of subsets of  $\text{typ}(T)$  are called sequents. If  $\Gamma \vdash \Delta$ , for  $\Gamma, \Delta \subseteq \text{typ}(T)$ , then the sequent  $\Gamma \vdash \Delta$  is a constraint.  $T$  is regular if for all  $\alpha \in \text{typ}(T)$  and all sets  $\Gamma, \Gamma', \Delta, \Delta', \Sigma', \Sigma_0, \Sigma_1$  of type: Identity:  $\alpha \vdash \alpha$ , Weakening: if  $\Gamma \vdash \Delta$ , then  $\Gamma, \Gamma' \vdash \Delta, \Delta'$ , Global Cut: if  $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$  for each partition  $\langle \Sigma_0, \Sigma_1 \rangle$  ( $\Sigma_0 \cup \Sigma_1 = \Sigma'$  and  $\Sigma_0 \cap \Sigma_1 = \emptyset$ ), then  $\Gamma \vdash \Delta$ .
  - 6) The core of the IF channel is a channel  $C$  is an index family  $C = \{f_i: A_i \leftrightarrow C\}_{i \in I}$  of infomorphisms with a common codomain  $C$  called the core of the  $C$ . the tokens of  $C$  are called *connections*; a *connection*  $c$  is said to *connect* the tokens  $f_i(c)$  for  $i \in I$ . A channel with index set  $\{0, \dots, n-1\}$  is called an *n-ary* channel.
  - 7) Introduction and Elimination Rules:

$$\text{f-Intro: } \frac{\Gamma \uparrow \vdash_A \Delta \uparrow}{\Gamma \vdash_B \Delta}$$

$$\text{f-Elim: } \frac{\Gamma \uparrow \vdash_B \Delta \uparrow}{\Gamma \vdash_A \Delta}$$

- 8) Let  $A$  be a classification and  $\Sigma \subseteq \text{Typ}(A)$ . A partition  $\langle \Gamma, \Delta \rangle$  of  $\Sigma$  is realized in  $A$  if there is a token  $a \in \text{tok}(A)$  such that  $\Gamma = \{\alpha \in \Sigma \mid a \vdash_A \alpha\}$ . Otherwise  $\langle \Gamma, \Delta \rangle$  is said to be spurious in  $A$ .
- 9) A representation system  $R = \langle C, \zeta \rangle$  consists of a binary channel  $C = \{f: A \leftrightarrow C, g: B \leftrightarrow C\}$ , with one of the classifications designated as source (say  $A$ ) and the other as target, together with a local logic  $\zeta$  on the core  $C$  of this channel. A local logic  $L = \langle A, \vdash_L, \text{NL} \rangle$  consists of a classification  $A$ , a set of sequent  $\vdash_L$  involving the types of  $A$ , the constraints of  $L$ , and a subset  $\text{NL}$  as the normal tokens of  $L$ , which satisfy  $\vdash_L$ .