AN ABSTRACT RELATIONAL MODEL AND NATURAL JOIN FUNCTORS

Kato, Akihiko
International Institute for Advanced Study of Social Information Science, Fujitsu Limited

https://doi.org/10.5109/13349

出版情報：Bulletin of informatics and cybernetics. 20 (3/4), pp.95-106, 1983-03. Research Association of Statistical Sciences

Bulletin of Informatics and Cybernetics Vol. 20, No. 3~4, 1983

AN ABSTRACT RELATIONAL MODEL AND NATURAL JOIN FUNCTORS

By

Akihiko KATO*

Abstract

A meta-model for database models called an abstract relational model which is obtained by a categorical abstraction of a relational model is proposed. This meta-model represents various database models, e.g. relational, network, hierarchical models as special cases. It is proved that a natural join is the right adjoint of a decomposition in the relational model. On the other hand, in our abstract relational model a natural join is defined as the right adjoint of a decomposition. A sufficient condition is shown for a database model to have natural joins.

1. Introduction

Several database models such as relational, network and hierarchical models have been investigated [2, 3, 4]. And also there are some investigations for unifications and conversions among these database models, e.g. [1]. However, these studies still adhere to actual database models and have not unified perspectives of various database models.

Category theory can be expected to give a comprehensive perspective for various theories. In fact, it has been applied to automata theory, system theory and computation theory etc. [8].

The purpose of this paper is to propose a meta-model, an abstract relational model, which is obtined by generalizing the relational database model in terms of category theory[1]. This abstract represents various database models, e. g. relational, network and hierarchical models as special cases. Furthermore, it gives a framework which allows to discuss database models without going into details of concrete database models. Particularly, a natural join is defined as the right adjoint of a decomposition in our abstract relational model while a natural join is defined by going into details of tuples of relations in Codd's relational model. Furthermore, our definition of natural joins shows clearly a relationship between natural joins and decompositions, which are not always the inverses of natural joins.

As an application of the abstract relational models, a sufficient condition is shown for a database model to have natural joins.

---

* International Institute for Advanced Study of Social Information Science, Fujitsu Limited, Numazu, 410-03 Japan.

1) There are other studies which attempt to apply category theory to relational databases [7, 9, 11].

For the above purpose, we take an approach that we reformalize the relational model by the category of sets and then we define an abstract relational model by generalizing such a categorical formalization.

In Section 2, we define categories of relations by introducing morphisms between relations using the category of sets to reformalize a relational database model. Projections, decompositions and naturl joins are defined to be functors and it is proved that a natural join is the right adjoint of a decomposition. In Section 3, an abstract relational model is defined by generalizing the relational model of Section 2. A natural join is defined as the right adjoint of a decomposition functor, As an important application, it is proved that an abstract relational model has natural joins if each functor included as a constituent of this model has the right adjoint and each category of occurrences has pullbacks. Finally, we exemplify formulations of relational, network and hierarchical madels as abstract relational models.

2. Relation Data Model on the Category of Sets

In this section, we formalize the category of relation on each attribute set and show that a natural join is regarded as the right adjoint of a decomposition by projections.

2.1 Category of relations and projection functors

We define the category of relations by introducing morphisms between relations.

DEFINITION 2.1. (Relation R) Let A be a set. We call A a *set of attributes* or an *attribute set*. A *relation on* A is a triple $R=(A, D, B)$, where $D=\{D_i\}_{i\in A}$ is a family of sets called *domains*, and B is a subset (called a *body* in this paper) of the Cartesian Product $\prod_{i\in A} D_i$ which is the set $\{t: A\to\bigcup_{i\in A} D_i \mid (\forall i\in A)t(i)\in D_i\}$. An element t of $\prod_{i\in A} D_i$ is called a *tuple* and is described by $(t(i))_{i\in A}$.

We introduce operators D^A and $\bar{\ }$ to take out the family of domains D and the body B of a given relation R:

$$R=(A, D^A R, \bar{R}).$$

Furthermore, for an attribute i $(i\in A)$, the domain D_i is denoted by $D_i^A R$ with the operator D_i^A.

The operator D^A will be extended to a functor later in this section.

DEFINITION 2.2. (morphism f) Let Q and R are relations on the same attribute set A. A *morphism f from Q to R* is a family $\{f_i: D_i^A Q\to D_i^A R\}_{i\in A}$ of functions indexed by A such that

$$(\prod_{i\in A} f_i)(\bar{Q})\subset(\bar{R}).$$

$\prod_{i\in A} f_i: \prod_{i\in A} D_i^A Q\to\prod_{i\in A} D_i^A R$ is the Cartesian product which assigns $t\in\prod_{i\in A} D_i^A Q$ to $(f_i(t(i)))_{i\in A}$.

We write $f: Q\to R$ or $Q\xrightarrow{f}R$ to indicate that the morphism f is from Q to R.

DEFINITION 2.3. (category $\boldsymbol{R}^A$) Let A be a set of attributes. $\boldsymbol{R}^A$ denotes the category of all the relations on A and all the morphisms among these relations. The composition $g\circ f: P\to R$ of morphisms $g: Q\to R$ and $f: P\to Q$ in $\boldsymbol{R}^A$ is defined as $\{g_i\circ f_i\}_{i\in A}$.

It is easy to see that $\boldsymbol{R}^A$ is a well-defined category.
The operator D^A becomes a funcor

$$D^A : \boldsymbol{R}^A \longrightarrow \boldsymbol{Set}^A$$

$$\begin{array}{ccc} Q & & D^A Q \\ \downarrow f & \longmapsto & \downarrow f \\ R & & D^A R \end{array}$$

where $\boldsymbol{Set}$ is the category of sets and functions and $\boldsymbol{Set}^A$ is the functor category from A (regarded as a discrete category) to $\boldsymbol{Set}$. $\boldsymbol{Set}^A$ can be regarded as the category of A-indexed families of sets and A-indexed families of functions.

DEFINITION 2.4. Let A and B be attribute sets and $a : A \to B$ be a function. Two functor $a^* : \boldsymbol{Set}^B \to \boldsymbol{Set}^A$ (the *projection functor of domains on* a) and $a^{(*)} : \boldsymbol{R}^B \to \boldsymbol{R}^A$ (the *projection functor of relations* on a) is defined as[2)]

(i) for $D = \{D_j\}_{j \in B} \in Ob\ \boldsymbol{Set}^B$, a^*D is defined as the family $\{D_{a(i)}\}_{i \in A}$.

(ii) for $g = \{g_j\}_{j \in B} : D \to D' \in Mor\ \boldsymbol{Set}^B$, a^*g is defined as the family $\{g_{a(i)}\}_{i \in A}$,

(iii) for $R \in Ob\ \boldsymbol{R}^B$, $a^{(*)}R = (A,\ a^*D^AR,\ \{(t(a(i)))_{i \in A} \mid t \in \bar{R}\})$,

(iv) for $h : Q \to R \in Mor\ \boldsymbol{R}^B$, $a^{(*)}h$ is equal to a^*h as a family of functions.

It is easy to see the followings:
for an attribute set A, $id_A^* = Id_{\boldsymbol{Set}_A}$ (indentity functor) and $id_A^{(*)} = Id_{\boldsymbol{R}^A}$, for a function $a : A \to B$ between attribute sets, $a^* \circ D^B = D^A \circ a^{(*)}$, and for attribute sets A, B, C and functions $a : A \to B$, $b : B \to C$, $(b \circ a)^* = a^* \circ b^*$ and $(b \circ a)^{(*)} = a^{(*)} \circ b^{(*)}$.

Fig. 2.1, shows the diagrams for understanding these statements.

$$\begin{tikzcd} \boldsymbol{R}^A \arrow[loop above, "id_A^{(*)}"] \arrow[r, "D^A"] & \boldsymbol{Set}^A \arrow[loop above, "id_A^*"] & A \arrow[d, "a"] \\ \boldsymbol{R}^B \arrow[u, "a^{(*)}"] \arrow[r, "D^B"] & \boldsymbol{Set}^B \arrow[u, "a^*"'] & B \arrow[d, "b"] \\ \boldsymbol{R}^C \arrow[u, "b^{(*)}"] & \boldsymbol{Set}^C \arrow[u, "b^*"'] & C \end{tikzcd}$$

Fig. 2.1.

2.2 Decomposition and natural join

A natural join composes a relation of two "compatible" (or "joinable") relations. On the other hand, two projections decompose a relation into two relations which are always "compatible."

We introduce categories of compatible relations and define decomposition functors

2) In this paper, for category $\boldsymbol{C}$, $Ob\ \boldsymbol{C}$ denotes the class of all the objects and $Mor\ \boldsymbol{C}$ denotes the class of all the morphisms of $\boldsymbol{C}$. Moreover, for objects X and Y of $\boldsymbol{C}$, $\boldsymbol{C}(X, Y)$ denotes the set of all the morphisms from X to Y.

and natural join functors. Finally, we show the adjointness theorem between natural joins and decompositions.

Throughout the following three definitions and a theorem. Let us suppose the diagram Fig. 2.2. for convenience, where A, B and C are attribute sets and b, c are functions.

$$\begin{tikzcd} B \arrow[dr, "b"'] & & C \arrow[dl, "c"] \\ & A & \end{tikzcd} \quad (\text{in } \boldsymbol{Set})$$

Fig. 2.2.

DEFINITION 2.5. ((b, c)-compatible) For the diagram Fig. 2.2, the *category* $\boldsymbol{R}^{b;c}$ *of* (b, c)*-compatible relations* is defined as follows.

An object is a triple (D, P, Q) where $D \in Ob\ \boldsymbol{Set}^A$, $P \in Ob\ \boldsymbol{R}^B$, $Q \in Ob\ \boldsymbol{R}^C$, $D^B P = b^* D$ and $D^C Q = c^* D$, i.e., the family of domains of P is the projection of D on b and the family of domains of Q is the projection of D on c.

A morphism is a triple

$$(g, h, k): (D, P, Q) \longrightarrow (D', P', Q')$$

where $g: D \to D'$ in $\boldsymbol{Set}^A$, $h: P \to P'$ in $\boldsymbol{R}^B$, $k: Q \to Q'$ in $\boldsymbol{R}^C$, $D^B h = b^* D$ and $D^C k = c^* D$.

DEFINITION 2.6. ((b, c)-decomposition functor) The (b, c)*-decomposition functor* $Dec^{b;c}$ which decomposes a relation on A into (b, c)-compatible relations is a functor defined by

$$\begin{array}{lcl} Dec^{b;c}: \boldsymbol{R}^A & \longrightarrow & \boldsymbol{R}^{b;c} \\ R & & (D^A R,\ b^{(*)} R,\ c^{(*)} R) \\ \Big\downarrow f & \longmapsto & \Big\downarrow (D^A f,\ b^{(*)} f,\ c^{(*)} f) \\ R' & & (D^A R',\ b^{(*)} R',\ c^{(*)} R'). \end{array}$$

On the other hand, natural join functors are defined in the following.

DEFINITION 2.7. ((b, c)-natural join functor) The (b, c)*-natural join functor* $Join^{b;c}$ is defined by

$$\begin{array}{lcl} Join^{b;c}: \boldsymbol{R}^{b;c} & \longrightarrow & \boldsymbol{R}^A \\ (D, P, Q) & & P * Q \\ \Big\downarrow (q, h, k) & \longmapsto & \Big\downarrow h * k \\ (D', P', Q') & & P' * Q' \end{array}$$

where $D^A(P*Q) = D$, $\overline{P*Q} = \{t \in \prod_{a \in A} D_a \mid (h_i t b(i))_{i \in B} \in \bar{P}$ and $(k_j t c(j))_{j \in C} \in \bar{Q}$ ($P'*Q'$ is defined similar to $P*Q$) and $h*k = g$.

Fig. 2.3 shows an example of (b, c)-natural join.

$$\begin{tikzcd}
B=\{i_1, i_2, i_3\} \arrow[dr, "b"'] & & C=\{j_1, j_2\} \arrow[dl, "c"] \\
& A=\{l_1, l_2, l_3\} &
\end{tikzcd}$$

P	i_1	i_2	i_3
	x_1	y_1	z_1
	x_1	x_1	z_2
	x_2	y_2	z_3

Q	j_1	j_2
	z_1	u_1
	z_2	u_2
	z_2	u_3

$P*Q$	l_1	l_2	l_3
	x_1	z_2	u_2
	x_1	z_2	u_3

Fig. 2.3.

THEOREM 1. (*Adjointness theorem between decompositions and natural joins*) *The natural join functor* $Join^{b;c}$ *is the right adjoint of the decomposition functor* $Dec^{b;c}$.

$$\begin{tikzcd}
\boldsymbol{R}^{b;c} \arrow[r, shift right, "Join^{b;c}"'] & \boldsymbol{R}^{A} \arrow[l, shift right, "Dec^{b;c}"']
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[dr, "b"] & \\
& A \\
C \arrow[ur, "c"'] &
\end{tikzcd}$$

PROOF. Let $R \in Ob\,\boldsymbol{R}^A$, $(D, P, Q) \in Ob\,\boldsymbol{R}^{b;c}$. We will show there is a bijective correspondence between $\boldsymbol{R}^{b;c}((D^AR, b^{(*)}R, c^{(*)}R), (D, P, Q))$ and $\boldsymbol{R}^A(R, P*Q)$.

Let $(g, h, k): (D^AR, b^{(*)}R, c^{(*)}R) \to (D, P, Q)$ be in $\boldsymbol{R}^{b;c}$ and $t \in \bar{R}$. We have $(h_i tb(i))_{i\in B} = (\prod_{i\in B} h_i)(tb(i))_{i\in B} \in \bar{P}$ since $\overline{b^{(*)}R} \ni (tb(i))_{i\in B}$. Similarly, $(k_j tc(j))_{j\in C} \in \bar{Q}$. Therefore, $t \in \overline{P*Q}$, i.e., g is a morphism $R \to P*Q$ in R^A.

We can conversely show that $(g, b^*g, c*g)$ is a morphism $(D^AR, b^{(*)}R, c^{(*)}R) \to (D, P, Q)$ for any $g: R \to P*Q$. (*q.e.d.*)

3. Abstract relational model

The purpose of this section is to propose an abstract relational model which is a meta-model of various database models such as relational, network and hierarchical model.

Theorem 1 holds when we change each $\boldsymbol{Set}^A$ to a functor category $\boldsymbol{D}^A$ where $\boldsymbol{D}$ is a category with direct products.

Furthermore, we can generalize the relational model of Section 2 to generate an abstract relational model.

3.1 Definition of abstract relational model

DEFINITION 3.1. An *abstract relational model* is a quadruple $M=(\boldsymbol{S}, \boldsymbol{T}, \boldsymbol{O}, T)$ where

(i) $\boldsymbol{S}$ is a category called the *category of schemata.*

(ii) for each $A \in Ob\,\boldsymbol{S}$, there are a *category* $\boldsymbol{T}^A$ *of types* (or *of domains*), a *caregory*

$\boldsymbol{O}^A$ *of occurrences on* A and a functor $T^A : \boldsymbol{O}^A \to \boldsymbol{T}^A$,

(iii) for each $a : A \to B \in Mor\, \boldsymbol{S}$, there are the *projection functor of types* $a^* : \boldsymbol{T}^B \to \boldsymbol{T}^A$ and the *projection functor of occurrences* $a^{(*)} : \boldsymbol{O}^B \to \boldsymbol{O}^A$, subject to the following conditions

(1) $id_A^* \cong Id_{\boldsymbol{T}^A},\ id_A^{(*)} \cong Id_{\boldsymbol{O}^A}$ $(A \in Ob\, \boldsymbol{S})$,

(2) $(b \circ a)^* \cong a^* \circ b^*,\ (b \circ a)^{(*)} \cong a^{(*)} \circ b^{(*)}$ (if $b \circ a$ is defined in $\boldsymbol{S}$),

(3) $a^* \circ T^B \cong T^A \circ a^{(*)}$ $(a : A \to B \in Mor\, \boldsymbol{S})$,

(4) a^* has the right adjoint a_* $(a \in Mor\, \boldsymbol{S})$.

We can say $\boldsymbol{T}$ and $\boldsymbol{O}$ are $\boldsymbol{S}$-indexed categories and $T : \boldsymbol{O} \to \boldsymbol{T}$ is a $\boldsymbol{S}$-indexed functor except that $\boldsymbol{S}$ may not have finite limits [5].

3.2 Decomposition and natural join in abstract relational models

Let $M=(\boldsymbol{S}, \boldsymbol{T}, \boldsymbol{O}, T)$ be an abstract relational model. We suppose the diagram Fig. 3.1. of $\boldsymbol{S}$ similarly to Section 2.2.

$$\begin{tikzcd} B & & C \\ & A \arrow[ul, "b"] \arrow[ur, "c"'] & \end{tikzcd} \quad (\text{in } \boldsymbol{S})$$

Fig. 3.1.

DEFINITION 3.2. ((b, c)-compatible occurrence) The *category* $\boldsymbol{O}^{b;c}$ of (b, c)-*compatible occurrences* is defined as follows.

(i) an object is a triple (D, P, Q) where $D \in Ob\, \boldsymbol{T}^A$, $P \in Ob\, \boldsymbol{O}^B$ and $Q \in Ob\, \boldsymbol{O}^C$ such that $T^B P \cong b^* D$ and $T^C Q \cong c^* D$,

(ii) a morphism $f : (D, P, Q) \to (D', P', Q')$ is a triple $f=(g, h, k)$ where $g : D \to D'$ $\in Mor\, \boldsymbol{T}^A$, $h : P \to P' \in Mor\, \boldsymbol{O}^B$ and $k : Q \to Q' \in Mor\, \boldsymbol{O}^C$ such that the following diagrams are commutative

$$\begin{tikzcd} D^B P \arrow[r, "D^B h"] \arrow[d, "\cong"] & D^B P' \arrow[d, "\cong"] \\ b^* D \arrow[r, "b^* g"] & b^* D' \end{tikzcd} \qquad \begin{tikzcd} D^C Q \arrow[r, "D^C k"] \arrow[d, "\cong"] & D^C Q' \arrow[d, "\cong"] \\ c^* D \arrow[r, "c^* g"] & c^* D' \end{tikzcd}$$

Decomposition functors are defined similarly to Definition 2.6.

DEFINITION 3.3. ((b, c)-decomposition functor) The (b, c)-*decomposition functor* $Dec^{b;c}$ is a functor defined by the following diagram,

$$\begin{array}{lccl} Dec^{b;c} : & \boldsymbol{O}^A & \longrightarrow & \boldsymbol{O}^{b;c} \\ & O & & (D^A O,\ b^{(*)} O,\ c^{(*)} O) \\ & f \downarrow & \longmapsto & \quad \downarrow (D^A f,\ b^{(*)} f,\ c^{(*)} f) \\ & O' & & (D^A O',\ b^{(*)} O',\ c^{(*)} O'). \end{array}$$

Natural join functors is defined using Theorem 1.

DEFINITION 3.4. ((*b*, *c*)-natural join functor) If the (*b*, *c*)-decomposition functor $Dec^{b;c}: \boldsymbol{O}^A \to \boldsymbol{O}^{b;c}$ has the right adjoint, then we call this right adjoint functor (*b*, *c*)-*national join functor* and write as

$$Join^{b;c}: \boldsymbol{O}^{b;c} \to \boldsymbol{O}^A .$$

$Join^{b;c}(D, P, Q)((D, P, Q) \in Ob\, O^{b;c})$ and $Join^{b;c}(g, h, k)((g, h, k) \in Mor\, \boldsymbol{O}^{b;c})$ are written as $P*Q$ and $h*k$ respectively for short.

We prove a lemma before Theorem 2 which shows a sufficient condition for an abstract relational model to have natural joins.

LEMMA 3.1. *Let* b^*, c^*, $b^{(*)}$, $c^{(*)}$, T^A, T^B *and* T^C *have the right adjoints* b_*, c_*, $b_{(*)}$, $c_{(*)}$, $Fill^A$, $Fill^B$ *and* $Fill^C$ *respectively, i.e.,*

$$\begin{tikzcd}
\boldsymbol{T}^B \arrow[r, shift right, "b_*"'] \arrow[d, shift left, "Fill^B"] & \boldsymbol{T}^A \arrow[l, shift right, "b^*"'] \arrow[r, shift left, "c^*"] \arrow[d, shift left, "Fill^A"] & \boldsymbol{T}^C \arrow[l, shift left, "c_*"] \arrow[d, shift left, "Fill^C"] \\
\boldsymbol{O}^B \arrow[u, shift left, "T^B"] \arrow[r, shift right, "b_{(*)}"'] & \boldsymbol{O}^A \arrow[u, shift left, "T^A"] \arrow[l, shift right, "b^{(*)}"'] \arrow[r, shift left, "c^{(*)}"] & \boldsymbol{O}^C \arrow[u, shift left, "T^C"] \arrow[l, shift left, "c_{(*)}"]
\end{tikzcd}$$

Then, for any $(D, P, Q) \in Ob\, \boldsymbol{O}^{b;c}$, *we have*

(1) $b_{(*)}Fill^B T^B P \cong Fill^A b_* b^* D$

(2) $c_{(*)}Fill^C T^C Q \cong Fill^C c_* c^* D$.

PROOF. We prove only (1). (2) is proved similarly.

Let natural transformations $\eta^B: Id_{O^B} \to Fill^B T^B$ and $e^b: Id_{T^A} \to b_* b^*$ be units of adjoint pairs $T^B \dashv Fill^B$ and $b^* \dashv b_*$ respectively. From the properties of adjoint functors, b^*T^A is the left adjoint of $Fill^A b_*$ and $T^B b^{(*)}$ is the left adjoint of $b_{(*)}Fill^B$. However, $b^*T^A \cong T^B b^{(*)}$ is assumed in Definition 3.1. Therefor, we have $Fill^A b_* \cong b_{(*)} Fill^B$ from the uniqueness of a rigt adjoint. (1) is satisfied since $T^B P \cong b^* D$. (*q.e.d.*)

THEOREM 2. *An abstract relational model M has natural joins if*

(1) *for each* $S \in Ob\, \boldsymbol{S}$, $T^S: \boldsymbol{O}^S \to \boldsymbol{T}^S$ *has the right adjoint* $Fill^S: \boldsymbol{T}^S \to \boldsymbol{O}^S$ *and* $\boldsymbol{O}^S$ *has pullbacks, and*

(2) *for each* $s: S \to S' \in Mor\, \boldsymbol{S}$, $s^{(*)}: \boldsymbol{O}^{S'} \to \boldsymbol{O}^S$ *has the right adjoint* $s_{(*)}: \boldsymbol{O}^S \to \boldsymbol{O}^{S'}$.

To prove this theorem, we use a property about adjoint functors that if $F, F': \boldsymbol{D} \to \boldsymbol{C}$ are functors with the right adjoint functors $G, G': \boldsymbol{C} \to \boldsymbol{D}$ respectively and $\rho: F' \to F$ is a natural transformation, then there is a natural transformation $\tau: G \to G'$ such that

$$\begin{tikzcd}
FY \arrow[dr, "f"] & \\
& X \\
F'Y \arrow[u, "\rho_Y"] \arrow[ur, "g"'] &
\end{tikzcd}$$

commutes if and only if

$$
\begin{tikzcd}
 & GX \arrow[d, "\tau_X"] \\
Y \arrow[ur, "\hat{f}"] \arrow[r, "\hat{g}"'] & G'X
\end{tikzcd}
$$

commutes where f and g are corresponding morphisms of f and g by adjoint pairs $F \dashv G$ and $F' \dashv G'$. Specially, if ρ is isomorphic, then τ is also isomorphic (e.g. [10, See 16.4.3]).

PROOF of Theorem 2. Let (D, P, Q) be an object of $\boldsymbol{O}^{b;c}$. We define $P*Q$ by Fig. 3.2 using isomorphisms in Lemma 3.1.

$$
\begin{tikzcd}
P*Q \arrow[r, dashed] \arrow[d, dashed] \arrow[dr, phantom, "\text{pullback}"] & Y \arrow[rr, dashed] \arrow[d, dashed] \arrow[drr, phantom, "\text{pullback}"] & & c_{(*)}Q \arrow[d, "c_{(*)}\eta^C_Q"] \\
X \arrow[r, dashed] \arrow[ddd, dashed] \arrow[dr, phantom, "\text{pullback}"] & Fill^A D \arrow[rr, "Fill^A e^b_D"'] \arrow[d, "Fill^A e^c_D"] & & Fill^A c_* c^* D \cong c_{(*)} Fill^C T^C Q \\
 & Fill^A b_* b^* D \arrow[d, phantom, "\cong" rotate=-90] & & \\
 & & & \\
b_{(*)}P \arrow[r, "b_{(*)}\eta^B_P"'] & b_{(*)} Fill^B T^B P & &
\end{tikzcd}
$$

Fig. 3.2.

in Fig. 3.2, η^B, η^C, e^b and e^c are units of adjoint pairs $T^B \dashv Fill^B$, $T^C \dashv Fill^C$, $b^* \dashv b_*$ and $c^* \dashv c_*$ respectively.

Let W be the subdiagram of the diagram in Fig. 3.2. which is represented by solid arrows.

Pullbacks in Fig. 3.2 make $P*Q$ the limit of W. Notice that there is a bijective correspondence between a morphism $f: R \to P*Q$ and a triple $(l: R \to Fill^A D,\ m: R \to b_{(*)}P,\ n: R \to c_{(*)}R)$ such that the diagram of Fig. 3.3 commutes since $P*Q = \varprojlim W$. We define Z as

$$
\begin{tikzcd}
R \arrow[rr, "n"] \arrow[dr, "l"] \arrow[ddd, "m"'] & & c_{(*)}Q \arrow[d, "c_{(*)}\eta^C_Q"] \\
 & Fill^A D \arrow[r, "Fill^A e^c_D"] \arrow[d, "Fill^A e^b_D"] & Fill^A c_* c^* D \cong c_{(*)} Fill^C T^C Q \\
 & Fill^A b_* b^* D \arrow[d, phantom, "\cong" rotate=-90] & \\
b_{(*)}P \arrow[r, "b_{(*)}\eta^B_P"'] & b_{(*)} Fill^B T^B P &
\end{tikzcd}
$$

Fig. 3.3.

the set of all the triples (l, m, n) such that Fig. 3.3 commutes.

It is sufficient for proving this theorem to show $Z \cong \boldsymbol{R}^{b;c}(Dec^{b;c}R, (D, P, Q))$.

Let $(g, h, k) : (T^A R, b^{(*)}R, c^{(*)}R) \to (D, P, Q)$ be in $\boldsymbol{R}^{b;c}$. We define $l : R \to Fill^A D$, $m : R \to b_{(*)}P$ and $n : R \to c_{(*)}Q$ by

$$\begin{tikzcd} R \arrow[r, "\eta^A_R"] \arrow[dr, "l"'] & Fill^A T^A R \arrow[d, "Fill^A g"] \\ & Fill^A D \end{tikzcd} \qquad \begin{tikzcd} R \arrow[r, "e^b_R"] \arrow[dr, "m"'] & b_{(*)}b^{(*)}R \arrow[d, "b_{(*)}h"] \\ & b^{(*)}P \end{tikzcd}$$

$$\begin{tikzcd} R \arrow[r, "e'^c_R"] \arrow[dr, "n"'] & c_{(*)}c^{(*)}R \arrow[d, "c_{(*)}k"] \\ & c_{(*)}Q \end{tikzcd}$$

where $e'^b : Id_{O^A} \to b_{(*)}b^{(*)}$ and $e'^c : Id_{O^A} \to c_{(*)}c^{(*)}$ are units of adjoint pairs $b^{(*)} \dashv b_{(*)}$ and $c^{(*)} \dashv c_{(*)}$ respectively.

First, we prove that the following diagram commutes:

$$\begin{tikzcd} R \arrow[rr, "l"] \arrow[ddd, "m"'] & & Fill^A D \arrow[d, "Fill^A e^b D"] \\ & & Fill^A b_* b^* D \arrow[d, phantom, "\cong" rotate=90] \\ & & \\ b_{(*)}P \arrow[rr, "b_{(*)}"'] & & b_{(*)}Fill^B T^B P. \end{tikzcd}$$

Let $s : T^B b^{(*)}R \to b^* D$ be a morphism in $\boldsymbol{T}^B$ such that

$$\begin{tikzcd} T^B b^{(*)}R \arrow[r, "\cong"] \arrow[d, "T^B h"'] \arrow[dr, "s"] & b^* T^A R \arrow[d, "b^* g"] \\ T^B P \arrow[r, "\cong"] & b^* D \end{tikzcd}$$

commutes.

From properties of adjoint functors, the diagrams of Fig. 3.4 are all commutative. In Fig. 3.4, operators "$\wedge$" and "$\sim$" are the adjunction isomorphisms of $Fill^A \dashv T^A$ (or $Fill^B \dashv T^B$) and $b^* \dashv b_*$ respectively.

$$\begin{tikzcd} T^B b^{(*)}R \arrow[r, "\cong"] \arrow[d, "T^B h"'] \arrow[dr, "s"] & b^* T^A R \arrow[d, "b^* g"] \\ T^B P \arrow[r, "\cong"] & b^* D \end{tikzcd}$$

$$\begin{tikzcd} R \arrow[d, "(T^B h)^{\wedge\sim}"'] \arrow[dr, "\tilde{\hat{s}}"] \arrow[drr, "(b^*g)^{\sim\wedge}"] & & \\ b_{(*)}Fill^B T^B P \arrow[r, "\cong"] & b_{(*)}Fill^B b^* D & Fill^A b_* b^* D \arrow[l, "\cong"'] \end{tikzcd}$$

$$\begin{tikzcd} T^B b^{(*)}R \arrow[d, "T^B h"'] \arrow[dr, "T^B h"] & \\ T^B P \arrow[r, "id_{T^B P}"] & T^B P \end{tikzcd} \qquad \begin{tikzcd} b^* T^A R \arrow[r, "b^* g"] \arrow[dr, "b^* g"'] & b^* D \arrow[d, "id_{b^* D}"] \\ & b^* D \end{tikzcd}$$

$$\begin{tikzcd} b^{(*)}R \arrow[d, "h"'] \arrow[dr, "(T^B h)^{\wedge}"] & \\ P \arrow[r, "\eta^B_P"'] & Fill^B T^B P \end{tikzcd} \qquad \begin{tikzcd} T^A R \arrow[r, "g"] \arrow[dr, "(b^* g)^{\sim}"'] & D \arrow[d, "e^b_D"] \\ & b^* b_* D \end{tikzcd}$$

$$\begin{tikzcd} R \arrow[d, "m"'] \arrow[dr, "(T^B h)^{\wedge\sim}"] & \\ b_{(*)}P \arrow[r, "b_{(*)}\eta^B_P"'] & b_{(*)}Fill^B T^B P \end{tikzcd} \qquad \begin{tikzcd} R \arrow[r, "l"] \arrow[dr, "(b^* g)^{\sim\wedge}"'] & Fill^A D \arrow[d, "Fill^A e^b_D"] \\ & Fill^A b_* b^* D \end{tikzcd}$$

Fig. 3.4.

Therefore, we have

$$\begin{tikzcd} R \arrow[r, "l"] \arrow[dd, "m"'] \arrow[dr, "(b^* g)^{\sim\wedge}"] \arrow[ddr, "(T^B h)^{\wedge\sim}"'] & Fill^A D \arrow[d, "Fill^A e^b_D"] \\ & Fill^A b_* b^* D \\ b_{(*)}P \arrow[r, "b_{(*)}\eta^B_P"'] & b_{(*)}Fill^B T^B P \arrow[u, "\cong"'] \end{tikzcd}$$

Similarly, the following diagram commutes:

$$\begin{tikzcd} R \arrow[rr, "n"] \arrow[d, "l"'] & & c_{(*)}Q \arrow[d, "c_{(*)}\eta^C_Q"] \\ Fill^A D \arrow[r, "Fill^A e^c_D"'] & Fill^A c_* c^* D & c_{(*)}Fill^C T^C Q, \arrow[l, "\cong"'] \end{tikzcd}$$

that is, $(l, m, n) \in Z$.

Obviously, this correspondence $(g, h, k) \mapsto (l, m, n)$ is naturally bijective. $(q.e.d.)$

3.3 Examples of abstract reational models

(1) Relational Data Model

We have the relational model when $\boldsymbol{S}$ is $\boldsymbol{Set}$ and for each (attribute) set A, Set, $\boldsymbol{T}^A$, $\boldsymbol{O}^A$ and T^A are $\boldsymbol{Set}^A$, $\boldsymbol{R}^A$ and D^A of Section 2 respectively. For a function $a: A \to B$ between attribute sets and $D=\{D_i\}_{i\in A} \in Ob\, \boldsymbol{Set}^A$, $a_* D$ is the family $\{\prod_{a(i)=j} D_i\}_{j\in B}$.

(2) Network Model

We define a simple network model $\boldsymbol{M}=(\boldsymbol{S}, \boldsymbol{T}, \boldsymbol{O}, T)$ as an abstract relational model.

(i) $\boldsymbol{S}$ is the category of multigraphs, i.e., Bachman's diagrams $G=(S, R, \Delta: S \to R\times R)$ where S is a set of set-types, R is a set of record-types and Δ assigns a set-type s to a pair $(own(s), mem(s))$ of owner and member record-types. A morphism $f:(S, R, \Delta) \to (S', R', \Delta')$ is a pair $f=(f^S: S \to S', f^R: R \to R')$ such that

$$\begin{tikzcd} S \arrow[r, "f^S"] \arrow[d, "\Delta"] & S \arrow[d, "\Delta'"] \\ R\times R \arrow[r, "f^R \times f^R"'] & R'\times R' \end{tikzcd}$$

commutes.

(ii) For $G=(S, R, \Delta)\in Ob\, \boldsymbol{S}$, $\boldsymbol{T}^G$ is the category $\boldsymbol{Set}^R$ whose object $\{V_r\}_{r\in R}$ is the family of value sets of record-types.

(iii) The occurrence category $\boldsymbol{O}^G$ $(G=(S, R, \Delta)\in Ob\, \boldsymbol{S})$ is defined as follows:

(a) an object is $N=(\{V_r\}_{r\in R}, \{X_r\}_{r\in R}, \{v_r: X_r \to V_r\}_{r\in R}, \{o_s: X_{mem(s)} \to X_{own(s)}\}_{s\in S})$ where $\{V_r\}_{r\in R} \in Ob\, \boldsymbol{T}^G$, $\{X_r\}_{r\in R}$ is a family of record sets, $v_r: X_r \to V_r$ assigns a record x to its value and $o_s: X_{mem(s)} \to X_{own(s)}$ assigns a member record to its owner record,

(b) a morphism $g: N \to N'$ $(N=(\{V_r\}_{r\in R}, \{X_r\}_{r\in R}, \{v_r\}_{r\in R}, \{o_s\}_{s\in S}),\ N'=(\{V'_r\}_{r\in R}, \{X'_r\}_{r\in R}, \{v'_r\}_{r\in R}, \{o'_s\}_{s\in S}))$ is a pair $g=(\{g^V_r: V_r \to V'_r\}_{r\in R}, \{g^X_r: X_r \to X'_r\}_{r\in R})$ such that $g^V_r \circ v_r = v'_r \circ g^X_r$ $(r\in R)$ and $g^X_{mem(s)} \circ o_s = o'_s \circ g^X_{own(s)}$.

(iv) Let $f: G \to G'$ $(G=(S, R, \Delta), G'=(S', R', \Delta'))$ be in $\boldsymbol{S}$. $f^*: \boldsymbol{T}^{G'} \to \boldsymbol{T}^G$ is equal to $(f^R)^*: \boldsymbol{Set}^{R'} \to \boldsymbol{Set}^R$ defined as in Section 2. $f_*: \boldsymbol{T}^G \to \boldsymbol{T}^{G'}$ is equal to $(f^R)_*: \boldsymbol{Set}^R \to \boldsymbol{Set}^{R'}$ defined as in the above example (1). $f^{(*)}: \boldsymbol{O}^{G'} \to \boldsymbol{O}^G$ assigns $N=(\{V_{r'}\}_{r'\in R'}, \{X_{r'}\}_{r'\in R'}, \{v_{r'}\}_{r'\in R'}, \{o_{s'}\}_{s'\in S'}) \in Ob\, O^{G'}$ to $(\{V_{f^R(r)}\}_{r\in R}, \{X_{f^R(r)}\}_{r\in R}, \{v_{f^R(r)}\}_{r\in R}, \{o_{f^S(s)}\}_{s\in S})$ and assigns $g=(\{g^V_{r'}\}_{r'\in r'}, \{g^X_{r'}\}_{r'\in R'}) \in Mor\, \boldsymbol{T}^{G'}$ to $(\{g^V_{f^R(r)}\}_{r\in R}, \{g^X_{f^R(r)}\}_{r\in R})$.

(v) $T^G: \boldsymbol{O}^G \to \boldsymbol{T}^G$ $(G=(S, R, \Delta))$ assigns $(\{V_r\}_{r\in R}, X, v, o) \in Ob\, \boldsymbol{O}^G$ to $\{V_r\}_{r\in R}$ and assigns $(\{g^V_r\}_{r\in R}, \{g^X_r\}_{r\in R})$ to $\{g^V_r\}_{r\in R}$.

(3) Hierarchical Model

The hierarchical model can be obtained restricting $\boldsymbol{S}$ of the network model in (2) to the category of trees.

The above examples all satisfy the condition of Theorem 2, that is, they have natural joins. We omit its proof. In the examples (2) and (3), notice that each category $\boldsymbol{O}^G$ of occurrences is equivalent to a certain $\boldsymbol{Set}$-valued functor category $\boldsymbol{Set}^{\tilde{G}}$.

4. Conclusion

We have proposed an abstract relational model which is a meta-model for database models such as relational, network and hierarchical models.

A natural join functor has been defined as the right adjoint of a decompositon functor. This definition has been obtained by the result that a natural join is the right adjoint of a decomposition in a relational database model on the category of sets.

As an application, a sufficient condition has been proved for a database model to have natural joins.

We can expect that concepts of the relational database models such as dependencies, normal forms, algebras are disscussed independently of database models in our approach.

Another topic that requires further research is semantics of an astract relational model.

Acknowledgements

In preparing this paper, the author is grateful to Dr. Tosio Kjtagawa, Director of the International Institute for Advanced Study of Social Information Science for his encouragement and advices, to Dr. Kaname Kobayasi, Mr. Hajime Sawamura and Mr. Taku Takesima of the insttitute for their valuable discussions and to Assistant Professor Dr. Yasuo Kawahara of Kyushu University, Mr. Yuzuru Tanaka of Hokkaido University and Mr. Tatsunari Mizukami of Kusiro Technical College for useful discussions.

References

[1] CHEN, P.P.S.: *The Etity-Relationship Model—Toward a Unified View of Data*, ACM Trans. on Database Systems, Vol. 1, No. 1 (1976), 9-36.

[2] CODD, E.F.: *Relational Model of Data for Larde Shared Data Banks*, CACM, Vol. 13, No. 6 (1970), 377-387.

[3] CODD, E.F.: *Relational Completeness of Database Sublanguages*, Courant Computer science Symposia 6: Database Systems, Prentice Hall (1972), 65-98.

[4] DATE, C.J.: *An Introduction to Database Systems*, Addison-Wesley (1975).

[5] JOHNSTONE, P.T. and PARÉ, R., ed.: *Indexed Categories and Their Applications*, Lec. Notes in Math. 661, Springer-Verlag (1978).

[6] KATO A.: *Projection/Join Functors and Decomposability of Relations*, Int. Inst. for Advanced Study of Soc. Inf. Sci., Fujitsu Ltd., Research Report No. 22 (1981).

[7] KAWAHARA, Y.: *Ken ni okeru Chusho Database ni tsuite (On Abstract Databases in Categories)*, unpublished, in Japanese (1981).

[8] MANES, E.G., ed.: *Category Theory Applied to Computation and Control*, Lec. Notes in Comp. Sci. 25, Springer-Verlag (1975).

[9] MIZUKAMI, T.: *Data Base Semantics Based on Intuitionistic Logics*, to appear.

[10] SHUBELT, H.: *Categories*, (translated from German by E. Gray), Springer-Verlag (1972).

[11] TANAKA, Y.: *Information Space Model*, Proc. Formal Bases for Data Bases (Toulous, 1979).

Communicated by T. Kitagawa
Received November 8, 1982