Chapter 26 Mutually-inversistic relational databases

26.1 First-level single quasi-relational databases

Example 26.1: The known: turn_around(*planet*, *fixed_star*) denoted by Table 26.1, turn_around(*satellite*, *planet*) denoted by Table 26.2, and the single empirical or mathematical theorem turn_around(*planet*, *fixed_star*)∩turn_around(*satellite*, *planet*)$\subseteq^{-1}$ turn_around(*satellite*, *fixed_star*). And we want to query: turn_around(Moon, Sun).

Table 26.1 Turn_around (*planet, fixed_star*)

planet	*fixed_star*
Earth	Sun
Jupiter	Sun

Table 26.2 Turn_around (*satellite, planet*)

satellite	*planet*
Moon	Earth
Jupiter_satel_one	Jupiter

Solution: We use indirect method of proof. We want to query turn_around(Moon, Sun), then we suppose that ~turn_around(Moon, Sun) holds. We make the negative expression of hypothetical inference from it and the empirical or mathematical theorem turn_around (*planet*, *fixed_star*)∩turn_around(*satellite*, *planet*)$\subseteq^{-1}$turn_around(*satellite*, *fixed_star*), inferring

~{turn_around(*planet*, Sun)∩turn_around(Moon, *planet*)} (26.1)

Formula (26.1) reminds us to make the fact second intersection of turn_around(*planet*, Sun) (Table 26.1)and turn_around(Moon, *planet*) (Table 26.2), shown in Table 26.3.

Table 26.3 turn_around(*satellite, fixed_star*)

satellite	*fixed_star*
Moon	Sun
Jupiter_satel_one	Sun

Formula (26.1) says that the fact second intersection of turn_around(*planet*, Sun) and turn_around(Moon, *planet*) does not exist, while Table 26.3 says that it exists, thus, contradiction occurs. This shows that the supposition ～turn_around(Moon, Sun) is false, and we prove (query) turn_around(Moon, Sun). The SQL statements of the query is as follows:

SELECT *satellite*, *fixed_star*
FROM turn_around(*planet*, *fixed_star*), turn_around(*satellite*, *planet*)
WHERE *satellite*=Moon, *fixed_star*=Sun.

26.2 Second-level single quasi-relational databases

The design of the second-level single quasi-relational database should be that no table is redundant, and that query is flexible and efficient, capable of querying more information. In order for each table being not redundant, we do the following.

$\{P\cup|^{-1}Q\}=^{-1}\{\sim P\subseteq^{-1}Q\}$
$\{P\cup^{-1}Q\}=^{-1}\{\sim P\subset^{-1}Q\}$
$\{P\oplus^{-1}Q\}=^{-1}\{\sim P=^{-1}Q\}$.

Because of the above formulas, we can transform $\cup|^{-1}$, $\cup^{-1}$, and $\oplus^{-1}$ into $\subseteq^{-1}$, $\subset^{-1}$, and $=^{-1}$ respectively. Therefore , in the construction of the second-level single quasi-relational database, we need only to consider $|\cap^{-1}$, $\subseteq^{-1}$, $=^{-1}$, $\subset^{-1}$, and $\times^{-1}$.

Some empirical or mathematical connection operators are strong, others are weak. For example,

$\{P\subset^{-1}Q\}\subseteq^{-1}\{P\subseteq^{-1}Q\}$

tells us that from $P\subset^{-1}Q$, $P\subseteq^{-1}Q$ can be inferred. Therefore, $P\subset^{-1}Q$ is stronger than $P\subseteq^{-1}Q$. If two fact propositions satisfy a stronger empirical or mathematical connection operator, then there is no need to preserve a weaker one, because the weaker one can be inferred from the stronger one. For example, if int(x)$\subset^{-1}$rat(x) holds, then there is no need to preserve int(x)$\subseteq^{-1}$rat(x), because the latter can be inferred from the former. Other single set theorems reflecting the strongness of empirical or mathematical connection operators are

$\{P=^{-1}Q\}\subseteq^{-1}\{P\subseteq^{-1}Q\}$
$\{P\subseteq^{-1}Q\}\subseteq^{-1}\{P|\cap^{-1}Q\}$
$\{P\times^{-1}Q\}\subseteq^{-1}\{P|\cap^{-1}Q\}$.

According to these theorems, we can make a figure reflecting the strongness of the empirical or mathematical connection operators shown in Fig. 26.1.

In Fig. 26.1, the three unicellular operators at the top are the strongest, $\subseteq^{-1}$ in the middle is intermediate, $|\cap^{-1}$ at the bottom is the weakest. We only consider the construction of $\subset^{-1}$ table, $=^{-1}$ table, and $\times^{-1}$ table.

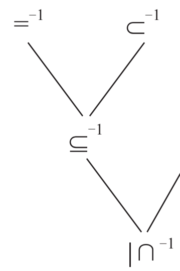

Fig. 26.1 Strongness of empirical or mathematical connection operators

There exist $\subset^{-1}$ connections among int(x), rat(x), and real(x), shown in Fig. 26.2, where the arrows show the directions of function determinations; i.e., int(x) functionally determines rat(x), and rat(x) functionally determines real(x).

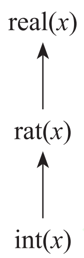

Fig. 26.2 $\subset^{-1}$ connections

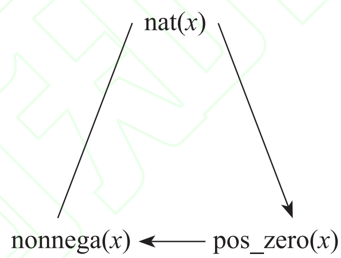

Fig. 26.3 $=^{-1}$ connections

According to Fig. 26.2, we can make the $P\subset^{-1}Q$ table shown in Table 26.4, where P is the primary key.

Table 26.4 $P\subset^{-1}Q$

P	Q
int(x)	rat(x)
rat(x)	real(x)

Table 26.5 $P=^{-1}Q$

P	Q
nat(x)	pos_zero(x)
pos_zero(x)	nonnega(x)

Table 26.4 only contains the two $\subset^{-1}$ connections indicated by the arrows of Fig. 26.2, it does not contain int(x)$\subset^{-1}$real(x). This is because $\subset^{-1}$ satisfies transitive law, int(x)$\subset^{-1}$ real(x) can be inferred from the two $\subset^{-1}$ connections indicated by the arrows of Fig. 26.2. Thus, Table 26.4 eliminates the transitive dependency, satisfying BC normal form.

There exist $=^{-1}$ connections among nat(x), pos_zero(x) (positive integers and zero), and nonnega(x) (nonnegative integers), shown in Fig. 26.3, where the arrows show the directions of function determinations. According to Fig. 26.3, we can make the $P=^{-1}Q$ table shown in Table 26.5, where P is the primary key. Table 26.5 only contains the two $=^{-1}$

connections indicated by the arrows of Fig. 26.3, it does not contain nonnega(x)$=^{-1}$nat(x). This is because $=^{-1}$ satisfies transitive law, nonnega(x)$=^{-1}$nat(x) can be inferred from the two $=^{-1}$ connections indicated by the arrows of Fig. 26.3. Thus, Table 26.5 eliminates the transitive dependency, satisfying BC normal form.

If a partition of a concept contains only two parts, then the partition is called a binary partition. Two different binary partitions of a concept satisfy $\times^{-1}$ connection. For example, a number is divided into a positive number (pos) and a nonpositive number (nonpos), it is also divided into a negative number (nega) and a nonnegative number (nonnega). Thus, pos(x)$\times^{-1}$ nega(x) holds. $\times^{-1}$ connections are shown in Fig. 26.4, where the arrows show the directions of function determinations.

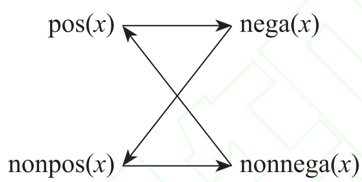

Fig. 26.4 $\times^{-1}$ connections

According to Fig. 26.4, we make $P\times^{-1}Q$ table shown in Table 26.6, where P is the primary key. Table 26.6 satisfies BC normal form.

Table 26.6 $P\times^{-1}Q$

P	Q
pos(x)	nega(x)
nega(x)	nonpos(x)
nonpos(x)	nonnega(x)
nonnega(x)	pos(x)

Example 26.2: Suppose we have $P\times^{-1}Q$ table shown in Table 26.7 and $Q\subset^{-1}R$ table shown in Table 26.8. We want to query pos(x)$|\cap^{-1}$rat(x).

Table 26.7 $P\times^{-1}Q$

P	Q
pos(x)	int(x)
int(x)	

Table 26.8 $Q\subset^{-1}R$

Q	R
int(x)	rat(x)
rat(x)	real(x)
real(x)	

Solution: First, let us carry out integrity analysis to Tables 26.7 and 26.8. Table 26.7 satisfies entity integrity, its primary key P does not adopt null value. Likewise, Table 26.8 satisfies entity integrity. Table 26.7 satisfies referential integrity, its foreign key Q adopts either the value of the primary key Q of the $Q\subset^{-1}R$ table (say, int(x)), or null value. Likewise, Tables 26.7 and 26.8 satisfy referential integrity themselves.

We use the indirect proof method to query. In order to query pos(x)$|\cap^{-1}$rat(x), we suppose $\sim$\{pos(x)$|\cap^{-1}$rat(x)\} to be true. $\sim$\{pos(x)$|\cap^{-1}$rat(x)\} and $\{P|\cap^{-1}Q\}\cap\{Q\subseteq^{-1}R\}\subseteq^{-1}$ $\{P|\cap^{-1}R\}$ make the second-level negative expression of hypothetical inference, inferring

$$\sim\{\{\text{pos}(x)|\cap^{-1}Q\}\cap\{Q\subseteq^{-1}\text{rat}(x)\}\} \quad (26.2)$$

Formula (26.2) is equivalent to

$$\sim\{\text{pos}(x)|\cap^{-1}Q\}\cup\sim\{Q\subseteq^{-1}\text{rat}(x)\} \quad (26.3)$$

Formula (26.3) can be changed to

$$\sim\{\text{pos}(x)|\cap^{-1}Q\}\cup|^{-1}\sim\{Q\subseteq^{-1}\text{rat}(x)\} \quad (26.4)$$

Formula (26.4) reminds us to query $Q\subseteq^{-1}$rat(x) table and pos(x)$|\cap^{-1}Q$ table respectively.

First, we query $Q\subseteq^{-1}$rat(x) table. There is no $\subseteq^{-1}$ table. But, $\sim$\{$Q\subseteq^{-1}$rat(x)\} and $\{P\subset^{-1}Q\}$ $\subseteq^{-1}\{P\subseteq^{-1}Q\}$ make the second-level negative expression of hypothetical inference, inferring $\sim$\{$Q\subset^{-1}$rat(x)\}. $\sim$\{$Q\subset^{-1}$rat(x)\} reminds us to query the $Q\subset^{-1}$rat(x) table. As expected, in Table 26.8, we query int(x)$\subset^{-1}$rat(x). Int(x)$\subset^{-1}$rat(x) and $\{Q\subset^{-1}R\}\subseteq^{-1}\{Q\subseteq^{-1}R\}$ make the second-level affirmative expression of hypothetical inference, inferring int(x)$\subseteq^{-1}$rat(x). Thus, we can make $Q\subseteq^{-1}R$ table shown in Table 26.9.

Table 26.9 $\boldsymbol{Q\subseteq^{-1}R}$

Q	R
int(x)	rat(x)

Table 26.10 $\boldsymbol{P|\cap^{-1}Q}$

P	Q
pos(x)	int(x)

Then, we query pos(x)$|\cap^{-1}Q$ table. There is no $|\cap^{-1}$ table. But, $\sim$\{pos(x)$|\cap^{-1}Q$\} and $\{P\times^{-1}Q\}\subseteq^{-1}\{P|\cap^{-1}Q\}$ make the second-level negative expression of hypothetical inference, inferring $\sim$\{pos(x)$\times^{-1}Q$\}. $\sim$\{pos(x)$\times^{-1}Q$\} reminds us to query pos(x)$\times^{-1}Q$ table. As expected, in Table 26.7, we query pos(x)$\times^{-1}$int(x). pos(x)$\times^{-1}$int(x) and $\{P\times^{-1}Q\}\subseteq^{-1}\{P|\cap^{-1}Q\}$ make the second-level affirmative expression of hypothetical inference, inferring pos(x)$|\cap^{-1}$ int(x). Thus, we can make $P|\cap^{-1}Q$ table shown in Table 26.10.

Formula (26.2) says that the empirical or mathematical second intersection of pos(x)$|\cap^{-1}Q$ and $Q\subseteq^{-1}$rat(x) does not exist, which reminds us to make the empirical or mathematical second intersection of Tables 26.10 and 26.9, obtaining Table 26.11.

Table 26.11 $\{\mathbf{pos}(x)|\cap^{-1}Q\}\cap\{Q\subseteq^{-1}\mathbf{rat}(x)\}$

P	*R*
pos(x)	rat(x)

Formula (26.2) says that the empirical or mathematical second intersection of pos(x)$|\cap^{-1}Q$ and $Q\subseteq^{-1}$rat(x) does not exist, while Table 26.11 shows that it exists. Thus, contradiction occurs. This shows the supposition $\sim$\{pos(x)$|\cap^{-1}$rat(x)\} is false, and we prove (query) pos(x)$|\cap^{-1}$rat(x). The SQL statements of the query is as follows:

SELECT *P*, *R*

FROM $P|\cap^{-1}Q$, $Q\subseteq^{-1}R$

WHERE *P*=pos(x), *R*=rat(x).

In the query, we can use the second-order main. For example, we can query int(x) $\subset^{-1}r(x)$; i.e., we ask what can be inferred from int(x). The answer is r=rat and r=real. In the query, we can also use the third-order main. For example, we can query int(x)φreal(x); i.e., we can ask what connection is satisfied by int(x) and real(x). The answer is $\varphi=\subset^{-1}$.

26.3 Combined single quasi-relational databases

Combined single quasi-relational databases combine first and second-level single quasi-relational databases.

Example 26.3: Suppose we have a personal_information table shown in Table 26.12, and a is_subset_of(*P*, *Q*) table shown in Table 26.13.

Table 26.12 Personal_information

name	occupation	work_at	work_year	father
Little Wang	journalist	Beijing Daily	25	Big Wang

Table 26.13 Is_subset_of (*P*, *Q*)

P	*Q*
journalist	capable_of_writing
capable_of_writing	literate
literate	

Based on Table 26.12, we can establish binary views:

CREATE VIEW occupation_view(name, occupation)

AS

```
SELECT name, occupation
FROM personal_information
```

and

```
CREATE VIEW father_view(name, father)
AS
SELECT name, father
FROM personal_information
```

We have the following first-level Datalog statements:

ancestor(x, y): -father_view(x, y).

ancestor(x, z): -father_view(x, y), ancestor(y, z).

We have the following combined Datalog statements:

is_member_of(x, Q): -occupation_view(x, P) AND
is_subset_of(P, Q).

is_member_of(x, Q): -is_member_of(x, P) AND
is_subset_of(P, Q).

These two statements correspond to $\{x\in P\}\wedge\{P\subseteq^{-1}Q\}\subseteq^{-1}\{x\in Q\}$, where $x\in P$ and $x\in Q$ are facts, $P\subseteq^{-1}Q$ is an empirical or mathematical connection. Hence they are combined Datalog statements. We have the following second-level Datalog statement:

is_subset_of(P, R): -is_subset_of(P, Q) AND
is_subset_of(Q, R).

The statement corresponds to $\{P\subseteq^{-1}Q\}\cap\{Q\subseteq^{-1}R\}\subseteq^{-1}\{P\subseteq^{-1}R\}$.

The system can make the following first-level inference:

ancestor(x, y): -fataher_view(x, y) AND
x=Little Wang AND
y=Big Wang,

inferring that Big Wang is Little Wang's ancestor.

The system can make the following combined inference:

is_member_of(x, Q): -occupaation_view(x, P) AND
is_subset_of(P, Q) AND
x=Little Wang AND
P=journalist AND
Q=capable-of_writing,

inferring that Little Wang is capable of writing.

The system can make the following second-level inference:

is_subset_of(P, R): -is_subset_of(P, Q) AND
is_subset_of(Q, R) AND
P=journalist AND

Q=capable_of_writing AND

R=literate,

inferring that journalists are all literate.

Combined single quasi_relational databases can be used to implement semantic networks: let the names of the binary views or binary tables be the tags of the directed arcs, let the first attributes be the starting vertices, let the second attributes be the arriving vertices. Such relations of semantic networks as is_subset_of, is_member_of, and is_part_of can all be implemented by combined single quasi-relational databases.