Equivalences of the Probabilistic Relational Algebra

Thomas Rölleke

Abstract

This paper serves to proof equivalences of the PRA expressions known from the RA. We restrict to definitions and proofs. For an introduction to the algebra refer to [Fuhr & Rölleke 95].

1 Definition of the PRA

First we review the classical definition of a relational database:

> Let $NAMES$ denote the universal set of attribute names and relation scheme names. A relation scheme S is a 2-tuple $< N, A >$, where $N \in NAMES$ is the relation scheme name and $A \subset NAMES$ is a set of attribute names. We define $nam(S) := N$ and $attr(S) := A$. Let A_i be an attribute name. D_j (e. g. $STRING$) denotes the domain of the values of an attribute A_i. We define $dom(A_i) := D_j$. Then the domain of a relation scheme S is the (associative) cartesian product of the domains of its attributes ($dom(S) := \times_{A_i \in attr(S)} dom(A_i)$). An element of $dom(S)$ is called tuple. A relation R is a 2-tuple $< S, V >$, where S is the relation scheme and $V \subset dom(S)$ is the value of the relation. We define $sch(R) := S$ and $val(R) := V$. As abbreviations we allow $attr(R) := attr(sch(R))$ and $dom(R) := dom(sch(R))$. Furthermore we do not distinguish names and instances of a relation where the context is clear. A database is defined as a set of relations.

Our intention is to extend the relational data model with the notion of probability for the tuples of a relation. Since our model provides intensional semantics ([Fuhr & Rölleke 95], [Pearl 88]), we first introduce a representation of events. The probability of a tuple is computed as a function of these event expressions.

Definition 1 (Event, $\mathcal{E}$, event expression, $\mathcal{EE}$) *An event is of the form $R(\delta)$, where R is a relation name and $\delta \in dom(R)$ is a tuple. $R(\delta)$ represents the event that a tuple δ is an element of a relation R. $E_\emptyset$ denotes the impossible event with probability 0. $\hat{E}$ denotes the certain event with probability 1. $\mathcal{E}$ denotes the set of events.*

Event expressions are defined recursively:

- *Every event $E \in \mathcal{E}$ is an event expression.*

Abbreviations

PRA	Probabilistic Relational Algebra
RA	Relational Algebra

- *Let E, E_1, and E_2 be event expressions. NOT E, (E), E_1 AND E_2 and E_1 OR E_2 are event expressions with the common binding priority.*

$\mathcal{EE}$ denotes the set of event expressions.

A tuple δ is identified by its key value, a relation R by its name. Thus we achieve a unique identification of events within the whole database.

Definition 2 (Elementary event, Ω) *The database contains n tuples. Let $E_1, ..., E_n$ be the corresponding events. Every conjunction E_1' AND ... AND E_n' is an elementary event where E_i' is an event ($E_i' = E_i$) or a negated event ($E_i' =$ NOT E_i). Ω denotes the set of elementary events. We say that "E is an event of K", iff E occurs unnegated in the elementary event K.*

The objective of the PRA is to model the uncertainty of the membership of a tuple to a relation. This uncertainty is expressed by a probability measure. To define what we mean by probability, we start with a set of possible worlds $\mathcal{W}$ ([Nilsson 86]). If our database (possible world) contains one tuple, we can imagine two sets of possible worlds: one set of possible worlds $\mathcal{W}_1$ containing the tuple and a second set $\mathcal{W}_2$, whose possible worlds don't contain the tuple. Thus we get for n tuples 2^n sets of possible worlds. Every set of possible worlds corresponds to an elementary event.

Given this semantics of elementary events, we can derive the definition of the probability of an event as the sum of the probabilities of those elementary events which contain the event.

Definition 3 (Probability P)

$$P(E) := \sum_K P(K), K \in \Omega \wedge E \text{ is an event of } K$$

And how do we get the probabilites of the elementary events? We can assume a set of possible worlds with a given probability distribution. The sum of the probabilities of those possible worlds which are elements of the set corresponding to an elementary event yields the probability of the elementary event.

So far we have explained the probability of an event. What we want to achieve is that P is a probability measure and thus is applicable to event expressions. Showing in the following that the powerset $\mathcal{P}(\Omega)$ is isomorph to $\mathcal{EE}$ yields two results:

1. $\mathcal{EE}$ is a Boolean algebra.
2. P is a probability measure on Ω.

Theorem 1 *The powerset $\mathcal{P}(\Omega)$ is isomorph to $\mathcal{EE}$.*

Proof: Let E_i be event expressions. Then the A_i are the corresponding sets of elementary events.

$$\begin{aligned}
\mathcal{P}(\Omega) &\hat{=} \mathcal{EE} \\
A_1 \cup A_2 &\hat{=} E_1 \text{ OR } E_2 \\
A_1 \cap A_2 &\hat{=} E_1 \text{ AND } E_2 \\
\Omega \setminus A_i &\hat{=} \text{ NOT } E_i \\
\emptyset &\hat{=} E_\emptyset \\
\Omega &\hat{=} \hat{E}
\end{aligned}$$

□

Theorem 2 *($\mathcal{EE}$, AND , OR , NOT , $E_\emptyset, \hat{E}$) is a Boolean Algebra.*

Proof: Because $\mathcal{EE}$ is isomorph to $\mathcal{P}(\Omega)$ and $\mathcal{P}(\Omega)$ is a Boolean Algebra ([Bronstein 87]), $\mathcal{EE}$ is a Boolean algebra. □

So the following equivalences of a Boolean algebra hold for event expressions. Let E_i be event expressions:

$$\begin{array}{rcl}
E_1 \text{ AND } E_2 & = & E_2 \text{ AND } E_1 \\
E_1 \text{ OR } E_2 & = & E_2 \text{ OR } E_1 \\
E_1 \text{ AND } (E_2 \text{ AND } E_3) & = & (E_1 \text{ AND } E_2) \text{ AND } E_3 \\
E_1 \text{ OR } (E_2 \text{ OR } E_3) & = & (E_1 \text{ OR } E_2) \text{ OR } E_3 \\
E_1 \text{ AND } (E_2 \text{ OR } E_3) & = & E_1 \text{ AND } E_2 \text{ OR } E_1 \text{ AND } E_3 \\
E_1 \text{ OR } (E_2 \text{ AND } E_3) & = & (E_1 \text{ OR } E_2) \text{ AND } (E_1 \text{ OR } E_3) \\
\text{NOT } (E_1 \text{ AND } E_2) & = & \text{NOT } E_1 \text{ OR NOT } E_2 \\
\text{NOT } (E_1 \text{ OR } E_2) & = & \text{NOT } E_1 \text{ AND NOT } E_2 \\
E_1 \text{ AND } (E_1 \text{ OR } E_2) & = & E_1 \\
E_1 \text{ OR } (E_1 \text{ AND } E_2) & = & E_1 \\
E_i \text{ AND } E_\emptyset & = & E_\emptyset \\
E_i \text{ OR } E_\emptyset & = & E_i \\
E_i \text{ AND } \hat{E} & = & E_i \\
E_i \text{ OR } \hat{E} & = & \hat{E} \\
E_i \text{ AND NOT } E_i & = & E_\emptyset \\
E_i \text{ OR NOT } E_i & = & \hat{E}
\end{array}$$

Theorem 3 *P is a probability measure.*

Proof: We proof the three laws of Kolmogorow ([Bronstein 87]): If we assume $\sum_{W \in \mathcal{W}} P(W) = 1$, then $\sum_{E \in \Omega} P(E) = P(\Omega) = 1$ holds. Thus we get $P(\hat{E}) = 1$. $P(E) \geq 0$ for all E and $P(E_1 \text{ OR } E_2) = P(E_1) + P(E_2)$ for disjunctive E_1 and E_2 directly follows regarding the events as sets of elementary events and elementary events as sets of possible worlds (s. a.). □

Now we introduce the formal definition of a probabilistic relation.

Definition 4 (Probabilistic relation R)

$$\begin{array}{rcl}
R & := & < S, \eta > \\
sch(R) & := & S, nam(S) := N, attr(S) := A \\
\eta(R) & := & \eta_R := \eta \\
dom(R) & := & \begin{cases} D_i, \textit{if R is an attribute} \\ \times_{A \in attr(R)} dom(A), \textit{if } R \textit{ is a probabilistic relation} \end{cases} \\
\eta & : & dom(R) \rightarrow \mathcal{EE}
\end{array}$$

The function η assigns event expressions to the tuples of the relation domain. The value of a relation given in the classical definition is derived by defining $val(R) := \{\delta | \delta \in dom(R) \wedge \eta_R(\delta) \neq E_\emptyset\}$. Because of this derivation it is sufficient to specify in the following formal definition of the algebra operations merely the manipulation of the function η.

For the common assumptions of the algebra operations (scheme equivalence, disjunctive attribute sets) refer to [Rölleke 94].

Definition 5 (Basic operations of the PRA)

$$\begin{aligned}
\eta_{R\cup S}(t) &:= \eta_R(t)\ \mathit{OR}\ \eta_S(t), \forall t \in dom(R \cup S) && (1)\\
\eta_{R\setminus S}(t) &:= \eta_R(t)\ \mathit{AND\ \ NOT}\ \eta_S(t), \forall t \in dom(R \setminus S) && (2)\\
\eta_{R\times S}(t) &:= \eta_R(< t[A_1], ..., t[A_n] >)\ \mathit{AND}\ \eta_S(< t[B_1], ..., t[B_m] >), && (3)\\
&\quad \forall t \in dom(R \times S)\\
\eta_{\sigma[\varphi](R)}(t) &:= \eta_R(t)\ \mathit{AND}\ \varphi'(t), \forall t \in dom(\sigma[\varphi](R)) && (4)\\
\eta_{\pi[L](R)}(t) &:= \ \mathit{OR}_{\ r\in dom(R)}\ \eta_R(< t[L], r[attr(R) - L] >), && (5)\\
&\quad \forall t \in dom(\pi[L](R))
\end{aligned}$$

2 Equivalences of the basic operations

Equivalences of an algebra are used to optimize the costs of evaluating an expression. Known heuristic strategies are trying to do selections and projections before unions, joins, etc. ([Maier 83], [Date 90], [Ullman 88]). We try to give a complete overview of equivalences by considering every possible connection of two operators.

Commutativity and associativity of the binary operations

$$\begin{aligned}
R \cup S &= S \cup R\\
R \times S &= S \times R\\
R \cup (S \cup T) &= (R \cup S) \cup T\\
R \times (S \times T) &= (R \times S) \times T
\end{aligned}$$

Proof: Because AND and OR are commutative and associative, the above equivalences hold. □

Union with difference

$$R \cup (S \setminus T) = (R \cup S) \setminus (T \setminus R)$$

Proof:

$$\begin{aligned}
\eta(t) &= \eta_{R\cup(S\setminus T)}(t) \Leftrightarrow\\
\eta(t) &= \eta_R(t)\ \text{OR}\ \eta_{S\setminus T}(t) \Leftrightarrow\\
\eta(t) &= \eta_R(t)\ \text{OR}\ (\eta_S(t)\ \text{AND\ \ NOT}\ \eta_T(t)) \Leftrightarrow\\
\eta(t) &= (\eta_R(t)\ \text{OR}\ \eta_S(t))\ \text{AND}\ (\eta_R(t)\ \text{OR\ \ NOT}\ \eta_T(t)) \Leftrightarrow\\
\eta(t) &= (\eta_R(t)\ \text{OR}\ \eta_S(t))\ \text{AND\ \ NOT}\ (\eta_T(t)\ \text{AND\ \ NOT}\ \eta_R(t)) \Leftrightarrow\\
\eta(t) &= \eta_{R\cup S}(t)\ \text{AND\ \ NOT}\ \eta_{T\setminus R}(t) \Leftrightarrow\\
\eta(t) &= \eta_{(R\cup S)\setminus(T\setminus R)}(t)
\end{aligned}$$

□

Difference with union

$$\begin{aligned}
R \setminus (S \cup T) &= (R \setminus S) \setminus T\\
(S \cup T) \setminus R &= (S \setminus R) \cup (T \setminus R)
\end{aligned}$$

Proof:

$$\begin{aligned}
\eta(t) &= \eta_{R\setminus(S\cup T)}(t) \Leftrightarrow \\
\eta(t) &= \eta_R(t) \text{ AND NOT } \eta_{S\cup T}(t) \Leftrightarrow \\
\eta(t) &= \eta_R(t) \text{ AND NOT } (\eta_S(t) \text{ OR } \eta_T(t)) \Leftrightarrow \\
\eta(t) &= \eta_R(t) \text{ AND NOT } \eta_S(t) \text{ AND NOT } \eta_T(t) \Leftrightarrow \\
\eta(t) &= \eta_{(R\setminus S)\setminus T}(t)
\end{aligned}$$

□

$$\begin{aligned}
\eta(t) &= \eta_{(S\cup T)\setminus R}(t) \Leftrightarrow \\
\eta(t) &= \eta_{S\cup T}(t) \text{ AND NOT } \eta_R(t) \Leftrightarrow \\
\eta(t) &= (\eta_S(t) \text{ OR } \eta_T(t)) \text{ AND NOT } \eta_R(t) \Leftrightarrow \\
\eta(t) &= \eta_{(S\setminus R)\cup(T\setminus R)}(t)
\end{aligned}$$

□

Cartesian product with union

$$T = R \times (S \cup T) \quad \Rightarrow \quad T = (R \times S) \cup (R \times T)$$

Proof:

$$\begin{aligned}
\eta(t) &= \eta_{R\times(S\cup T)}(t) \Leftrightarrow \\
\eta(t) &= \eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_{S\cup T}(< t[B_1], ..., t[B_m] >) \Leftrightarrow \\
\eta(t) &= \eta_R(< t[A_1], ..., t[A_n] >) \text{ AND} \\
&\quad (\eta_S(< t[B_1], ..., t[B_m] >) \text{ OR } \eta_T(< t[B_1], ..., t[B_m] >)) \Leftrightarrow \\
\eta(t) &= \eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_S(< t[B_1], ..., t[B_m] >) \text{ OR} \\
&\quad \eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_T(< t[B_1], ..., t[B_m] >) \Rightarrow \\
\eta(t) &= \eta_{R\times S}(t) \text{ OR } \eta_{R\times T}(t) \Leftrightarrow \\
\eta(t) &= \eta_{(R\times S)\cup(R\times T)}(t)
\end{aligned}$$

□

Cartesian product with difference

$$T = R \times (S \setminus T) \quad \Rightarrow \quad T = (R \times S) \setminus (R \times T)$$

Proof:

$$\begin{aligned}
\eta(t) &= \eta_{R\times(S\setminus T)}(t) \Leftrightarrow \\
\eta(t) &= \eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_{S\setminus T}(< t[B_1], ..., t[B_m] >) \Leftrightarrow \\
\eta(t) &= \eta_R(< t[A_1], ..., t[A_n] >) \text{ AND} \\
&\quad \eta_S(< t[B_1], ..., t[B_m] >) \text{ AND NOT } \eta_T(< t[B_1], ..., t[B_m] >) \Leftrightarrow \\
\eta(t) &= (\eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_S(< t[B_1], ..., t[B_m] >)) \text{ AND} \\
&\quad (\text{ NOT } \eta_R(< t[A_1], ..., t[A_n] >) \text{ OR NOT } \eta_T(< t[B_1], ..., t[B_m] >)) \Leftrightarrow \\
\eta(t) &= (\eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_S(< t[B_1], ..., t[B_m] >)) \text{ AND NOT} \\
&\quad (\eta_R(< t[A_1], ..., t[A_n] >) \text{ AND } \eta_T(< t[B_1], ..., t[B_m] >)) \Rightarrow \\
\eta(t) &= \eta_{R\times S}(t) \text{ AND NOT } \eta_{R\times T}(t) \Leftrightarrow \\
\eta(t) &= \eta_{(R\times S)\setminus(R\times T)}(t)
\end{aligned}$$

□

Selection with union

$$\sigma[\varphi](R \cup S) \quad = \quad \sigma[\varphi](R) \cup \sigma[\varphi](S)$$

Selection is distributive over union.
Proof:

$$\begin{aligned}
\eta(t) &= \eta_{\sigma[\varphi](R \cup S)}(t) \Leftrightarrow \\
\eta(t) &= \eta_{R \cup S}(t) \text{ AND } \varphi'(t) \Leftrightarrow \\
\eta(t) &= \eta_R(t) \text{ AND } \varphi'(t) \text{ OR } \eta_S(t) \text{ AND } \varphi'(t) \Leftrightarrow \\
\eta(t) &= \eta_{\sigma[\varphi](R)}(t) \text{ OR } \eta_{\sigma[\varphi](S)}(t) \Leftrightarrow \\
\eta(t) &= \eta_{\sigma[\varphi](R) \cup \sigma[\varphi](S)}(t)
\end{aligned}$$

□

Selection with difference

$$\sigma[\varphi](R \setminus S) \quad = \quad \sigma[\varphi](R) \setminus \sigma[\varphi](S)$$

Selection is distributive over difference.
Proof:

$$\begin{aligned}
\eta(t) &= \eta_{\sigma[\varphi](R \setminus S)}(t) \Leftrightarrow \\
\eta(t) &= \eta_{R \setminus S}(t) \text{ AND } \varphi'(t) \Leftrightarrow \\
\eta(t) &= \eta_R(t) \text{ AND } \text{ NOT } \eta_S(t) \text{ AND } \varphi'(t) \Leftrightarrow \\
\eta(t) &= \eta_R(t) \text{ AND } \varphi'(t) \text{ AND } \text{ NOT } (\eta_S(t) \text{ AND } \varphi'(t)) \Leftrightarrow \\
\eta(t) &= \eta_{\sigma[\varphi](R) \setminus \sigma[\varphi](S)}(t)
\end{aligned}$$

□

Selection with cartesian product

$$T = \sigma[\varphi](R) \times S \quad \Rightarrow \quad T = \sigma[\varphi](R \times S)$$

Selection is distributive over cartesian product, if all attributes occurring in φ are attributes of one argument relation of the cartesian product.
Proof:

$$\begin{aligned}
\eta(t) &= \eta_{\sigma[\varphi](R) \times S}(t) \Leftrightarrow \\
\eta(t) &= \eta_{\sigma[\varphi](R)}(t) \text{ AND } \eta_S(t) \Rightarrow \\
\eta(t) &= \eta_R(t) \text{ AND } \varphi'(t) \text{ AND } \eta_S(t) \Leftrightarrow \\
\eta(t) &= \eta_{R \times S}(t) \text{ AND } \varphi'(t) \Leftrightarrow \\
\eta(t) &= \eta_{\sigma[\varphi](R \times S)}(t)
\end{aligned}$$

□

Selection with selection

$$\begin{aligned} \sigma[\varphi_1](\sigma[\varphi_2](R)) &= \sigma[\varphi_2](\sigma[\varphi_1](R)) \\ &= \sigma[\varphi_1 \wedge \varphi_2](R) \end{aligned}$$

The order of a selection sequence is alterable. The conditions of a selection sequence may be combined by conjunction.

Proof:

$$\begin{aligned} \eta(t) &= \eta_{\sigma[\varphi_1](\sigma[\varphi_2](R))}(t) \Leftrightarrow \\ \eta(t) &= \eta_{\sigma[\varphi_2](R)}(t) \text{ AND } \varphi_1'(t) \Leftrightarrow \\ \eta(t) &= \eta_R(t) \text{ AND } \text{ AND } \varphi_2'(t) \text{ AND } \varphi_1'(t) \Leftrightarrow \\ \eta(t) &= \eta\sigma[\varphi_2](\sigma[\varphi_1](R))(t) \end{aligned}$$

□

Selection with projection

$$T = \sigma[\varphi](\pi[L](R)) \quad \Rightarrow \quad T = \pi[L](\sigma[\varphi](R))$$

The reversed direction is only allowed, if all the attributes occurring in φ are also elements of L.

Proof:

$$\begin{aligned} \eta(t) &= \eta_{\sigma[\varphi](\pi[L](R))}(t) \Leftrightarrow \\ \eta(t) &= \eta_{\pi[L](R)}(t) \text{ AND } \varphi'(t) \Leftrightarrow \\ \eta(t) &= (\text{ OR }_{r \in dom(R)} \eta_R(< t[L], r[attr(R) - L >)) \text{ AND } \varphi'(t) \Rightarrow \\ \eta(t) &= \text{ OR }_{r \in dom(\sigma[\varphi](R))} \eta_{\sigma[\varphi](R)}(< t[L], r[attr(\sigma[\varphi](R)) - L] >) \Leftrightarrow \\ \eta(t) &= \eta_{\pi[L](\sigma[\varphi](R))}(t) \end{aligned}$$

□

Projection with union

$$T = \pi[L](R \cup S) \quad \Rightarrow \quad T = \pi[L](R) \cup \pi[L](S)$$

The reversed direction is only allowed, if the relations R and S have the same scheme, i. e. $attr(R) = attr(S)$.

Proof:

$$\begin{aligned} \eta(t) &= \eta_{\pi[L](R \cup S)}(t) \Leftrightarrow \\ \eta(t) &= \text{ OR }_{r \in dom(R \cup S)} \eta_{R \cup S}(< t[L], r[attr(R) - L] >) \Rightarrow \\ \eta(t) &= \text{ OR }_{r \in dom(R \cup S)} (\eta_R(< t[L], r[attr(R) - L] >) \text{ OR } \eta_S(< t[L], r[attr(S) - L] >)) \Leftrightarrow \\ \eta(t) &= \text{ OR }_{r \in dom(R)} \eta_R(< t[L], r[attr(R) - L] >) \text{ OR } \\ & \quad \text{ OR }_{r \in dom(S)} \eta_S(< t[L], r[attr(S) - L] >) \Leftrightarrow \\ \eta(t) &= \eta_{\pi[L](R)}(t) \text{ OR } \eta_{\pi[L](S)}(t) \Leftrightarrow \\ \eta(t) &= \eta_{\pi[L](R) \cup \pi[L](S)}(t) \end{aligned}$$

□

Projection with difference There is no meaningful general equivalence with projection and difference.

Projection with cartesian product

$$\begin{aligned}
\pi[L_R, L_S](R \times S) &= \pi[L_R](R) \times \pi[L_S](S), \\
&\quad\ L_R \subseteq attr(R) \wedge L_S \subseteq attr(S) \\
\pi[L_R, L_S](R \times S) &= \pi[L_R](R), L_S = \emptyset \wedge \exists t \in dom(S)(\eta_S(t) = \hat{E})
\end{aligned}$$

Proof:

$$\begin{aligned}
\eta(t) &= \eta_{\pi[L_R, L_S](R \times S)}(t) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(R \times S)}\ \eta_{R \times S}(< t[L_R, L_S], t[attr(R \times S) - L_R - L_S >) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(R \times S)}\ \eta_R(< t[attr(R) >) \text{ AND } \eta_S(< t[attr(S) >) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(R)}\ \eta_R(< t[L_R], r[attr(R) - L_R] >) \text{ AND} \\
&\quad\ \text{OR}_{r \in dom(S)}\ \eta_S(< t[L_S], r[attr(S) - L_S] >) \Leftrightarrow \\
\eta(t) &= \eta_{\pi[L_R](R)}(< t[L_R] >) \text{ AND } \eta_{\pi[L_S](S)}(< t[L_S] >) \Leftrightarrow \\
\eta(t) &= \eta_{\pi[L_R](R) \times \pi[L_S](S)}(t)
\end{aligned}$$

□

Projection with selection See selection with projection ...

Projection with projection

$$T = \pi[L_1](\pi[L_2](R)) \quad \Rightarrow \quad T = \pi[L_1](R)$$

Proof:

$$\begin{aligned}
\eta(t) &= \eta_{\pi[L_1](\pi[L_2](R))}(t) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(\pi[L_2](R))}\ \eta_{\pi[L_2](R)}(< t[L_1], r[attr(\pi[L_2](R)) - L_1] >) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(\pi[L_2](R))}\ \text{OR}_{r' \in dom(R)}\ \eta_R(< t[L_1], r[L_2 - L_1], r'[attr(R) - L_2] >) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(R)}\ \eta_R(< t[L_1], r[L_2 - L_1], r[attr(R) - L_2] >) \Leftrightarrow \\
\eta(t) &= \text{OR}_{r \in dom(R)}\ \eta_R(< t[L_1], r[attr(R) - L_1] >) \Leftrightarrow \\
\eta(t) &= \eta_{\pi[L_1](R)}(t)
\end{aligned}$$

□

3 Definition of derived operations

Definition 6 (Derived operations)

$$\begin{aligned}
R \cap S &:= R \setminus (R \setminus S) \\
R \underset{A \Theta B}{\bowtie} S &:= \sigma[A \Theta B](R \times S) \\
R \bowtie S &:= \pi[L](\sigma[\varphi_{\bowtie}](R \times S)), L = attr(R) \cup attr(S), \\
&\qquad \varphi_{\bowtie} = (A_i^R = A_i^S \wedge \ldots), A_i \in attr(R) \cap attr(S)
\end{aligned}$$

4 Equivalences of the derived operations

Commutativity, associativity, distributivity of derived operations

$$\begin{aligned}
R \cup (S \cap T) &= (R \cup S) \cap (R \cup T) \\
R \setminus (S \cap T) &= (R \setminus S) \cup (R \setminus T) \\
(S \cap T) \setminus R &= (S \setminus R) \cap (T \setminus R) \\
R \times (S \cap T) &= (R \times S) \cap (R \times T)
\end{aligned}$$

Intersection

$$\begin{aligned}
R \cap S &= S \cap R \\
R \cap (S \cap T) &= (R \cap S) \cap T \\
R \cap (S \cup T) &= (R \cap S) \cup (R \cap T)
\end{aligned}$$

There is no equivalence for $R \cap (S \setminus T)$.

The following equivalences of theta-join and join marked with (*) hold, if the corresponding attributes of the join selection are attributes of the relation S.

Theta-Join

$$\begin{aligned}
R \underset{A\Theta B}{\bowtie} S &= S \underset{A\Theta B}{\bowtie} R \\
R \underset{A\Theta B_i}{\bowtie} (S \underset{B_j\Theta C}{\bowtie} T) &= (R \underset{A\Theta B_i}{\bowtie} S) \underset{B_j\Theta C}{\bowtie} T \\
R \underset{A\Theta B}{\bowtie} (S \cup T) &= (R \underset{A\Theta B}{\bowtie} S) \cup (R \underset{A\Theta B}{\bowtie} T) \\
R \underset{A\Theta B}{\bowtie} (S \setminus T) &= (R \underset{A\Theta B}{\bowtie} S) \setminus (R \underset{A\Theta B}{\bowtie} T) \\
R \underset{A\Theta B}{\bowtie} (S \times T) &= (R \underset{A\Theta B}{\bowtie} S) \times T(*) \\
R \underset{A\Theta B}{\bowtie} (S \cap T) &= (R \underset{A\Theta B}{\bowtie} S) \cap (R \underset{A\Theta B}{\bowtie} T) \\
R \underset{A\Theta B}{\bowtie} (S \bowtie T) &= (R \underset{A\Theta B}{\bowtie} S) \bowtie T(*)
\end{aligned}$$

Join

$$\begin{aligned}
R \bowtie S &= S \bowtie R \\
R \bowtie (S \bowtie T) &= (R \bowtie S) \bowtie T \\
R \bowtie (S \cup T) &= (R \bowtie S) \cup (R \bowtie T) \\
R \bowtie (S \setminus T) &= (R \bowtie S) \setminus (R \bowtie T) \\
R \bowtie (S \times T) &= (R \bowtie S) \times T(*) \\
R \bowtie (S \cap T) &= (R \bowtie S) \cap (R \bowtie T) \\
R \bowtie (S \underset{A\Theta B}{\bowtie} T) &= (R \bowtie S) \underset{A\Theta B}{\bowtie} T(*)
\end{aligned}$$

Selection with Intersection

$$\sigma[\varphi](R \cap S) = \sigma[\varphi](R) \cap \sigma[\varphi](S)$$

Proof:

$$\begin{aligned}
T &= \sigma[\varphi](R \cap S) \Leftrightarrow \\
T &= \sigma[\varphi](R \setminus (R \setminus S)) \Leftrightarrow \\
T &= \sigma[\varphi](R) \setminus \sigma[\varphi](R \setminus S) \Leftrightarrow \\
T &= \sigma[\varphi](R) \setminus (\sigma[\varphi](R) \setminus \sigma[\varphi](S)) \Leftrightarrow \\
T &= \sigma[\varphi](R) \cap \sigma[\varphi](S)
\end{aligned}$$

□

Selection with Theta-Join

$$T = \sigma[\varphi](R) \underset{A\Theta B}{\bowtie} S \quad \Rightarrow \quad T = \sigma[\varphi](R \underset{A\Theta B}{\bowtie} S)$$

Proof:

$$\begin{aligned}
T &= \sigma[\varphi](R) \underset{A\Theta B}{\bowtie} S \Leftrightarrow \\
T &= \sigma[A\Theta B](\sigma[\varphi](R) \times S) \Rightarrow \\
T &= \sigma[A\Theta B](\sigma[\varphi](R \times S)) \Leftrightarrow \\
T &= \sigma[\varphi](\sigma[A\Theta B](R \times S)) \Leftrightarrow \\
T &= \sigma[\varphi](R \underset{A\Theta B}{\bowtie} S)
\end{aligned}$$

□

Selection with Join

$$T = \sigma[\varphi](R) \bowtie S \quad \Rightarrow \quad T = \sigma[\varphi](R \bowtie S)$$

Proof:

$$\begin{aligned}
T &= \sigma[\varphi](R) \bowtie S \Leftrightarrow \\
T &= \pi[L](\sigma[\varphi_{\bowtie}](\sigma[\varphi](R) \times S)) \Rightarrow \\
T &= \pi[L](\sigma[\varphi_{\bowtie}](\sigma[\varphi](R \times S))) \Leftrightarrow \\
T &= \sigma[\varphi](\pi[L](\sigma[\varphi_{\bowtie}](R \times S))) \Leftrightarrow \\
T &= \sigma[\varphi](R \bowtie S)
\end{aligned}$$

□

Projection with Intersection There is no meaningful general equivalence with projection and intersection.

Projection with Theta-Join

$$\pi[L_R, L_S](R \underset{A\Theta B}{\bowtie} S) \quad = \quad \pi[L_R](R) \underset{A\Theta B}{\bowtie} \pi[L_S](S), A \in L_R \wedge B \in L_S$$

Proof:

$$\begin{aligned}
T &= \pi[L_R, L_S](R \underset{A\Theta B}{\bowtie} S) \Leftrightarrow \\
T &= \pi[L_R, L_S](\sigma[A\Theta B](R \times S)) \Leftrightarrow \\
T &= \sigma[A\Theta B](\pi[L_R, L_S](R \times S)) \Leftrightarrow \\
T &= \sigma[A\Theta B](\pi[L_R](R) \times \pi[L_S](S)) \Leftrightarrow \\
T &= \pi[L_R](R) \underset{A\Theta B}{\bowtie} \pi[L_S](S)
\end{aligned}$$

□

Projection with Join

$$\pi[L_R, L_S](R \bowtie S) \quad = \quad \pi[L_R](R) \bowtie \pi[L_S](S)$$

Proof:

$$\begin{aligned}
T &= \pi[L_R, L_S](R \bowtie S) \Leftrightarrow \\
T &= \pi[L_R, L_S](\pi[L_\bowtie](\sigma[\varphi_\bowtie](R \times S))) \Leftrightarrow \\
T &= \pi[L_R, L_S](\sigma[\varphi_\bowtie](R \times S)) \Leftrightarrow \\
T &= \pi[L_R, L_S](\sigma[\varphi_\bowtie](\pi[L_R, L_S](R \times S))) \Leftrightarrow \\
T &= \pi[L_\bowtie](\sigma[\varphi_\bowtie](\pi[L_R](R) \times \pi[L_S](S))) \Leftrightarrow \\
T &= \pi[L_R](R) \bowtie \pi[L_S](S)
\end{aligned}$$

□

5 Reduction rules

In the following we give reduction rules of algebra expressions. We omit the straight forward proofs.

5.1 Redundant Expressions

$$\begin{aligned}
R \cup R &= R \\
R \setminus R &= \emptyset \\
R \cap R &= R \\
R \bowtie R &= R
\end{aligned}$$

5.2 Operations with the empty relation

$$\begin{aligned}
R \cup \emptyset &= R \\
R \setminus \emptyset &= R \\
\emptyset \setminus R &= \emptyset \\
R \cap \emptyset &= \emptyset \\
R \times \emptyset &= \emptyset \\
R \bowtie \emptyset &= \emptyset \\
\sigma[\varphi](\emptyset) &= \emptyset \\
\pi[L](\emptyset) &= \emptyset
\end{aligned}$$

References

Bronstein, I. N. (1987). *Taschenbuch der Mathematik.* Harri Deutsch, Thun, Frankfurt am Main.

Date, C. (1990). *An Introduction to Database Systems*, volume I. Addison-Wesley, Reading (Mass.), 5th edition.

Fuhr, N.; Rölleke, T. (1995). *A Probabilistic Relational Algebra for the Integration of Information Retrieval and Database Systems*. (Submitted for publication).

Maier, D. (1983). *Relational Database Theory*. Computer Science Press, Rockville, Md.

Nilsson, N. J. (1986). Probabilistic Logic. *Artificial Intelligence 28*, pages 71–87.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufman, San Mateo, Cal.

Rölleke, T. (1994). *Development of a Probabilistic Relational Data Model*. Diploma thesis, University of Dortmund, Department of Computer Science. (in German).

Ullman, J. (1988). *Principles of Database and Knowledge-Base Systems*, volume I. Computer Science Press, Rockville (Md.).