AN EXTENDED RELATIONAL ALGEBRA FOR NESTED RELATIONS

A THESIS SUBMITTED TO
THE DEPARTMENT OF COMPUTER ENGINEERING AND
INFORMATION SCIENCE
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

by
Eser Sükan

January, 1993

AN EXTENDED RELATIONAL ALGEBRA FOR NESTED RELATIONS

A THESIS SUBMITTED TO
THE DEPARTMENT OF COMPUTER ENGINEERING AND
INFORMATION SCIENCE
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

by
Eser Sükan

January 1993 Eser SÜKAN

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Erol Arkun (Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Varol Akman

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. İlyas Çiçekli

Approved by the Institute of Engineering:

Prof. Mehmet Baray,
Director of the Institute of Engineering

ABSTRACT

AN EXTENDED RELATIONAL ALGEBRA FOR NESTED RELATIONS

Eser Sükan
M.S. in Computer Engineering and Information Science
Supervisor: Prof. Erol Arkun
January 1993

In this study the database models of Roth-Korth-Silberschatz (RKS) [cf. *ACM TODS 13(4): 389–417, 1988*] and Abiteboul-Bidoit (AB) [cf. *Journal of Computer and System Sciences 33(4): 361–393, 1986*] to formalize non-first-normal-form relations are presented along with their extended relational algebra. We show that the extended set operators *union* and *difference* of RKS and AB are not *information equivalent.* Using the model of RKS and restricting ourselves to *union* and *difference*, we define our extended set operators and show that these two operators and the *extended intersection* of RKS are information equivalent.

Keywords: Data models, normal forms, extended algebra, nested relations, non-first-normal-form relations, partitioned normal form

ÖZET

İÇİÇE İLİŞKİLER İÇİN GENİŞLETİLMİŞ BİR İLİŞKİSEL CEBİR

Eser Sükan
Bilgisayar ve Enformatik Mühendisliği Bölümü, Yüksek Lisans
Tez Yöneticisi: Prof. Dr. Erol Arkun
Ocak 1993

Bu çalışmada birinci normal biçimde olmayan ilişkileri formalize etmek için Roth-Korth-Silberschatz (RKS) [cf. *ACM TODS 13(4): 389–417, 1988*] ve Abiteboul-Bidoit (AB) [cf. *Journal of Computer System Sciences 33(4): 361–393, 1986*] tarafından geliştirilmiş veritabanı modelleri ve bu modeller için tanımlanmış bir ilişkisel cebir sunulmaktadır. Gerek RKS gerekse AB cebirleri içinde yer alan genişletilmiş küme operatörlerinden ***birleşim*** ve ***farkın***, ***bilgi eşdeğer*** olmadığı gösterilmektedir. RKS'nin modeli kullanılarak, genişletilmiş küme operatörlerinden ***birleşim*** ve ***fark*** yeniden tanımlanmaktadır. Ayrıca yeni tanımlanan ***birleşim, fark*** ve RKS'nin *genişletilmiş kesişim* operatörlerinin bilgi eşdeğer olduğu gösterilmektedir.

Anahtar Sözcükler: Veri modelleri, normal biçimler, genişletilmiş cebir, içiçe ilişkiler, birinci normal biçimde olmayan ilişkiler, bölümlemeli normal biçim

ACKNOWLEDGEMENTS

I wish to express my deep gratitude to my supervisors Prof. Meral Özsoyoğlu and Prof. Erol Arkun for their invaluable support during the development of this thesis. I also want to thank to Dr. Varol Akman for the time he spent on reading and correcting this thesis.

Contents

1 Introduction **1**

2 The Model **3**

2.1 The Model of RKS 3

2.2 The Verso Model of AB 4

3 Extended Relational Algebra **7**

3.1 *Nest* and *Unnest* Operators 7

3.2 The Partitioned Normal Form 9

3.3 Extended Set Operators 11

3.3.1 Extended Union 11

3.3.2 Extended Difference 24

3.3.3 Extended Intersection 35

4 Conclusions **39**

List of Figures

2.1 Tree representation of STUDENT(COURSE(BOOK GRADE)*)* 6

3.1 A sample flat relation 8

3.2 An example for *nest* operator 8

3.3 An example for *unnest* operator 9

3.4 Examples for ¬PNF and PNF relations 10

3.5 Purely hierarchical relations 12

3.6 Extended union of r_1 and r_2 12

3.7 The *desired-result* 13

3.8 Examples for ¬*purely hierarchical* relations 14

3.9 Extended union of r_1 and r_2 14

3.10 The *desired-result* 14

3.11 $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$ 15

3.12 Flat forms of $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$ 15

3.13 Extended difference of r_1 and r_2 26

3.14 The *desired-result* 26

3.15 Extended difference of r_1 and r_2 26

3.16 The *desired-result* 26

3.17 $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$ 29

3.18 Flat forms of $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$ 29

3.19 Extended intersection of r_1 and r_2 36

3.20 The *desired-result* 36

Chapter 1

Introduction

The first-normal-form (1NF) assumption of traditional relational model (in which all values are atomic) [8] has been relaxed by the introduction of new applications of database systems in areas such as text and image processing, computer-aided design, etc. which require relations within relations. A new class of relations, that of ¬1NF (non-first-normal-form or nested) relations, has been introduced for such applications. The nested relational model represents real world data better by allowing relation-valued attributes.

Nested relations have been an extensive research area since the late seventies. The nested relational model was first introduced by Makinouchi [5]; this was followed by works by others [7, 6, 2, 3, 4, 1]. Among these, Schek and Scholl [7] introduced relations with relation-valued attributes and proposed a recursive relational algebra for these relations in which the standard set operators $\cup$, $-$, and $\cap$ are applied to ¬1NF relations without any change. Abiteboul and Bidoit (AB) [2] presented the Verso model, which is a data model for ¬1NF relations. The nested structure of the Verso model is obtained by the recursive definition of the Verso instances, i.e., the attributes in a Verso instance may have Verso instances as well as atomic values. Relational algebra operators on Verso instances are also defined. (This will be discussed in the sequel.)

Roth, Korth, and Silberschatz (RKS) [6] introduced an extended relational algebra for a proper subset of nested relations which are considered to be in *partitioned normal form* (PNF). They defined *extended set operators* which are rather different than the ones in other works. The idea behind extended set operators is that tuples that agree on their atomic attributes are combined to form a new tuple. Our thesis is based on this work and a detailed discussion of these set operators is presented in the third chapter.

Garnett and Tansel [4] proposed an extended relational algebra and showed that this algebra is equivalent in expressive power to relational calculus for nested relations. They used the standard set operators $\cup$, $-$, and $\cap$ for nested relations without any change.

In this work we restrict ourselves to the set operators union, difference, and intersection for nested relations in *partitioned normal form.* Our aim is twofold: to show that the extended set operators, *union* and *difference*, defined in [6] and [2], are not information equivalent, and to define information equivalent set operators for nested relations. A set operator is *information equivalent* if it generates a result which becomes equal to the desired-result when it is flattened. Here the *desired-result* is the result obtained by first flattening the two relations and then applying the standard set operator to the flat relations.

This thesis is structured as follows. We present the models for nested relations introduced by RKS and AB in the second chapter. The third chapter contains the relational algebra of RKS and AB. We show that their extended set operators *union* and *difference* are not information equivalent and introduce information equivalent set operators ($\cup^e, -^e$). Proofs showing that our extended set operators and the *extended intersection* of RKS are information equivalent are also included in this chapter. Chapter four concludes the thesis.

Chapter 2

The Model

We assume that the reader is familiar with the relational model and do not go through well-known concepts such as attribute, domain, etc. We first present the model introduced by RKS. This is the model our work is based on. We then present the Verso model introduced by AB.

2.1 The Model of RKS

A $\neg$1NF *database scheme* S is defined as a collection of rules of the form $R_j = (R_{j_1}, \ldots, R_{j_n})$, where R_j, and R_{j_i}, $1 \leq i \leq n$, are *names.* (The model uses names and attributes interchangeably.) Each of these rules represents a higher-order or a zero-order name. This means that the rules in a $\neg$1NF database scheme may consist of any number of zero-order or higher-order names as long as the scheme is not recursive. A rule R_j is a *higher-order name* if it appears on the left-hand side of a rule, and is a *zero-order name* otherwise. The names on the right-hand side of a rule R_j form the set E_{R_j}, viz. the elements of R_j.

A zero-order name is an atomic attribute which has an associated domain. A higher-order name is a nested relation scheme whose domain is composed of the related domains of each zero-order name in this scheme.

Example: Consider a database scheme which contains the following rules:

STUDENT = (STUDENT_ID, STUDENT_NAME, COURSES)
COURSES = (COURSE_NAME, BOOK, GRADE)

The STUDENT database contains student identification (STUDENT_ID), student name (STUDENT_NAME), and the courses taken by the student (COURSES), for each student. STUDENT and COURSES are higher-order names and the others are zero-order names. □

A *relation scheme* R is called a *subscheme* if no zero-order name appears on the right-hand side of two different rules in the scheme. To define the subscheme of a database S, let R_j appear only on the left-hand side of some rule in S (i.e., R_j is an *external* name). The rules in S that are accessible from R_j form a subscheme of S defined as follows:

1. $R_j = (R_{j_1}, \ldots, R_{j_n})$ is in the subscheme, and
2. Whenever a higher-order name R_k is on the right-hand side of some rule in the subscheme, the rule $R_k = (R_{k_1}, \ldots, R_{k_n})$ is also in the subscheme.

An *instance* r of a name R is defined as an ordered tuple $< R, V_R >$ where V_R is a value for R. For zero-order names, V_R is an atomic value from the associated domain of R, while for higher-order names, it is a value composed of the values from the related domains of the names on the right-hand side of R.

A *database structure* $\mathcal{S} = < S, s >$ is composed of the database scheme S and an instance s of that scheme. A *relation structure* $\mathcal{R} = < R, r >$ is composed of the relation scheme R and an instance r of that scheme. Two structures $\mathcal{S}_1$ and $\mathcal{S}_2$ are equal if their schemes and instances are equal, respectively. (Two relation schemes R_1 and R_2 are equal if they consist of the same rules.)

NB. In this model (of RKS), null values in $\neg$1NF relations are not considered.

2.2 The Verso Model of AB

Before we define the model, we present the notation of AB. The set of tuples over a relational scheme V is denoted *tup(V)*, and the set of relations is denoted *rel(V)*. The set of ordered tuples over some *string* X (i.e., a set of attributes,

$X = A_1 \ldots A_n$) is denoted *Otup(X)* and the corresponding set of attributes in a string X is denoted *set(X)* ($= \{A | A \in X\}$).

The data structure of the Verso model is defined by using the concept of *format.* A format is defined as follows:

1. If X is a finite string of attributes with no repeated attribute, then X is a flat format over *set(X)*, and
2. If X is a nonempty finite string of attributes with no repeated attribute and $f_1, \ldots, f_n$ formats over $Y_1, \ldots, Y_n$, respectively, then the string $X(f_1)^* \ldots (f_n)^*$ is a format over the set *set(X)*$Y_1 \ldots Y_n$, where *set(X)*, $Y_1, \ldots, Y_n$ are pairwise disjoint.

Null values can be represented in the Verso model. The empty string is a format which is denoted Λ. If $f = X(f_1)^* \ldots (f_n)^*$ is a format, and $f_i = \Lambda$ for some i, $1 \leq i \leq n$, then $f = X(f_1)^* \ldots (f_{i-1})^*(f_{i+1})^* \ldots (f_n)^*$.

Example : If we let f_1 = STUDENT COURSE GRADE, then f_1 is a format over {STUDENT, COURSE, GRADE}. Now if we let f_2 = STUDENT(COURSE-(BOOK GRADE)*)*, then f_2 is a format over {STUDENT, COURSE, BOOK, GRADE}.

Directed trees are used in [2] to represent formats. Figure 2.1 shows the tree representation of f_2. The root of the tree is STUDENT (the flat format of f_2), and the only branch of the tree is (COURSE(BOOK GRADE)*)*. □

The set of all *instances, inst(f)*, over a format f is defined as follows:

1. If $f \equiv X$ and *set(X)* $\neq \emptyset$, then I is in *inst(f)* iff I is a finite subset of *Otup(X)*, and
2. If $f \equiv X(f_1)^* \ldots (f_n)^*$, where $f_1, \ldots, f_n$ are nonempty, then I is in *inst(f)* iff
   (a) I is a finite subset of *Otup(X)*$\times$*inst*$(f_1) \times \ldots \times$ *inst*(f_n), and
   (b) if $< u, I_1, \ldots, I_n >$ and $< u, J_1, \ldots, J_n >$ are in I for some $u, I_1, \ldots, I_n, J_1, \ldots, J_n$, then $I_i = J_i$, for all i, $1 \leq i \leq n$.

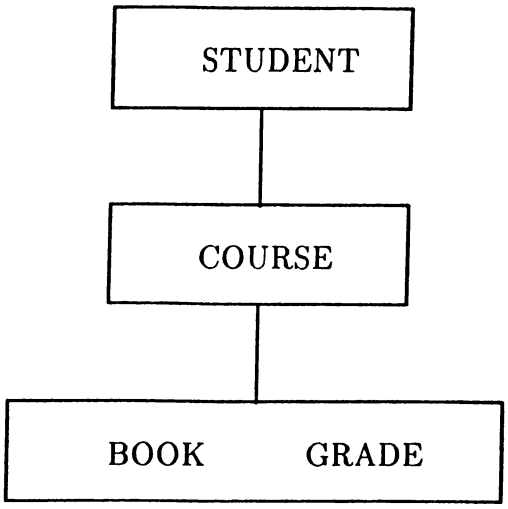

Figure 2.1: Tree representation of STUDENT(COURSE(BOOK GRADE)*)*

Thus, in the light of condition (2), the atomic attributes of a format constitute a key.

Chapter 3

Extended Relational Algebra

In this chapter we present the extended relational algebra of RKS, and AB by restricting ourselves to $\cup$, $-$, and $\cap$. We also show that the extended operators *union* and *difference* are not information equivalent and introduce our own extended set operators which are shown to be information equivalent.

3.1 *Nest* and *Unnest* Operators

Two new operators *nest*(ν) and *unnest*(μ) are introduced in the extended relational algebra of RKS. We use these operators in order to show that our extended set operators are information equivalent. These operators modify the relation structures that they act upon.

Nest combines the data values which agree on some of their attributes and is defined as follows in [6]:

Let R be a relation scheme, in database scheme S, which contains a rule $R = (A_1, \ldots, A_n)$ for external name R. Let $\{B_1, \ldots, B_m\} \subset E_R$ and $\{C_1, \ldots, C_k\} = E_R - \{B_1, \ldots, B_m\}$. Assume that either the rule $B = (B_1, \ldots, B_m)$ is in S or that B does not appear on the left-hand side of any rule in S and $(B_1, \ldots, B_m)$ does not appear on the right-hand side of any rule in S. Then $\nu_{B=(B_1,\ldots,B_m)}(\mathcal{R}) = < R', r' > = \mathcal{R}'$ where:

1. $R' = (C_1, \ldots, C_k, (B_1, \ldots, B_m)) = (C_1, \ldots, C_k, B)$ and $B = (B_1, \ldots, B_m)$ is appended to the set of rules in S if it is not already in S, and

r

A	C	D	F	G
a_1	c_1	d_1	f_1	g_1
a_1	c_1	d_1	f_2	g_2
a_1	c_1	d_1	f_3	g_3
a_1	c_2	d_2	f_1	g_1
a_1	c_2	d_2	f_2	g_2
a_2	c_3	d_3	f_1	g_1
a_2	c_3	d_3	f_4	g_4
a_2	c_4	d_4	f_1	g_1
a_2	c_4	d_4	f_4	g_4

Figure 3.1: A sample flat relation

$\nu_{E=(F,G)}(\nu_{B=(C,D)}(r))$

A	B		E	
	C	D	F	G
a_1	c_1	d_1	f_1	g_1
	c_2	d_2	f_2	g_2
a_1	c_1	d_1	f_3	g_3
a_2	c_3	d_3	f_1	g_1
	c_4	d_4	f_4	g_4

$\nu_{B=(C,D)}(\nu_{E=(F,G)}(r))$

A	B		E	
	C	D	F	G
a_1	c_1	d_1	f_1	g_1
			f_2	g_2
			f_3	g_3
a_1	c_2	d_2	f_1	g_1
			f_2	g_2
a_2	c_3	d_3	f_1	g_1
	c_4	d_4	f_4	g_4

Figure 3.2: An example for *nest* operator

2. $r' = \{t \mid$ there exists a tuple $u \in r$ such that $t[C_1 \ldots C_k] = u[C_1 \ldots C_k]$ $\wedge t[B] = \{v[B_1 \ldots B_m] \mid v \in r \wedge v[C_1 \ldots C_k] = t[C_1 \ldots C_k]\}\}$

Example: Let r be a relation on the relation scheme $R = (A, C, D, F, G)$ (Figure 3.1). Two relations $\nu_{B=(C,D)}(\nu_{E=(F,G)}(r))$ and $\nu_{E=(F,G)}(\nu_{B=(C,D)}(r))$ (Figure 3.2) with the scheme $R' = (A, B, E)$, $B = (C, D)$, $E = (F, G)$ are obtained from r by applying the nest operators in different orders (i.e., in the first table of Figure 3.2 r is nested with respect to E, B and in the second table it is nested with respect to B, E.) □

Unnest, on the other hand, flattens a relation on some attributes, and is defined as follows in [6]:

Let R be a relation scheme, in database scheme S, which contains a rule R

$\mu_B(r_1)$

A	C	D	E	
			F	G
a_1	c_1	d_1	f_1	g_1
			f_2	g_2
a_1	c_2	d_2	f_1	g_1
			f_2	g_2
a_1	c_1	d_1	f_3	g_3
a_2	c_3	d_3	f_1	g_1
			f_4	g_4
a_2	c_4	d_4	f_1	g_1
			f_4	g_4

$\mu_B(r_2)$

A	C	D	E	
			F	G
a_1	c_1	d_1	f_1	g_1
			f_2	g_2
			f_3	g_3
a_1	c_2	d_2	f_1	g_1
			f_2	g_2
a_2	c_3	d_3	f_1	g_1
			f_4	g_4
a_2	c_4	d_4	f_1	g_1
			f_4	g_4

Figure 3.3: An example for *unnest* operator

$= (A_1, \ldots, A_n)$ for external name R. Assume that B is some higher-order name in E_R with an associated rule $B = (B_1, \ldots, B_m)$. Let $\{C_1, \ldots, C_k\} = E_R - B$. Then $\mu_{B=(B_1,\ldots,B_m)}(\mathcal{R}) = < R', r' > = \mathcal{R}'$ where:

1. $R' = (C_1, \ldots, C_k, B_1, \ldots, B_m)$ and $B = (B_1, \ldots, B_m)$ is removed from the set of rules in S if it does not appear in any other relation scheme, and

2. $r' = \{t \mid$ there exists a tuple $u \in r$ such that $t[C_1 \ldots C_k] = u[C_1 \ldots C_k]$ $\wedge\ t[B_1 \ldots B_m] \in u[B]\}$.

Example: Let us unnest the relations $r_1 = \nu_{E=(F,G)}(\nu_{B=(C,D)}(r))$ and $r_2 = \nu_{B=(C,D)}(\nu_{E=(F,G)}(r))$ (Figure 3.2) with B. The results $\mu_B(r_1)$ and $\mu_B(r_2)$ are shown in Figure 3.3. If these results are unnested with E, the flat relation r (Figure 3.1) is generated. □

3.2 The Partitioned Normal Form

Since it is possible to obtain different relations by nesting the same relation with respect to the same nest operators in different orders, the class of ¬1NF relations are restricted and only the relations in *partitioned normal form* (PNF) are considered in [6]. The partitioned normal form restriction guarantees that *nest* is an inverse of *unnest* and provides a less redundant representation of ¬1NF relations.

r_1

A	B	
	C	D
a_1	c_1	d_1
	c_2	d_2
a_1	c_3	d_3
a_2	c_4	d_4
	c_1	d_2

r_1'

A	B	
	C	D
a_1	c_1	d_1
	c_2	d_2
	c_3	d_3
a_2	c_4	d_4
	c_1	d_2

Figure 3.4: Examples for ¬PNF and PNF relations

Example: The relation r_1 (Figure 3.4) is a ¬1NF relation that is not in PNF, while r_1' in the same figure is a ¬1NF relation in PNF that represents the same information as r_1. □

Now let us introduce the definitions for PNF as presented in [6] :

Definition 5.1 Let X, $Y \subseteq E_R$ for some relation structure $\mathcal{R} = < R, r >$. The *functional dependency* (FD), $X \rightarrow Y$, holds in r iff for all tuples t_1, t_2 in r, if $t_1[X] = t_2[X]$ then $t_1[Y] = t_2[Y]$. (If X or Y is a higher-order name then we mean set equality.)

Definition 5.2 Let $\mathcal{R} = < R, r >$ be a relation structure with attribute set E_R containing zero-order names $A_1, \ldots, A_k$ and higher-order names $X_1, \ldots, X_l$. $\mathcal{R}$ is in *partitioned normal form* (PNF) iff

1. $A_1, A_2, \ldots, A_k \rightarrow E_R$, and
2. For all $t \in r$ and for all X_i, $1 \leq i \leq l$, $\mathcal{R}_{t_i} = < X_i, t[X_i] >$ is in PNF.

In the light of these definitions, a nested relation without any zero-order attributes ($k = 0$) is in PNF iff it contains a single tuple (cf. [6], p. 397).

The work of RKS aims to prove that given a relation in PNF, whenever an operator (*nest* or *unnest*) is applied, the result is also in PNF . This is true for *unnest* in any case, and true for *nest* in some special cases. These are stated as **Theorems 5.1** and **5.2** and proved in [6]. For convenience, we state these theorems now.

Theorem 5.1 *The class of PNF relations is closed under unnesting.*

Theorem 5.2 *The nesting of a PNF relation is in PNF iff in the PNF relation* $\mathcal{R} = < R, r >$, $A_1, \ldots, A_k \rightarrow X_1, \ldots, X_l$ *where* $A_1, \ldots, A_k$ *are the zero-order names in* E_R *not being nested and* $X_1, \ldots, X_l$ *are the higher-order names in* E_R *not being nested.*

3.3 Extended Set Operators

A common point of *extended set operators* defined in [6], [2], and our work is that they are all recursive formulations. In another approach, two relations are flattened, any standart set operator is applied to these flat relations, and the resultant flat relation is restructured into its original structure. In this approach the property that *nest* is an inverse operator for *unnest* is required. (This is not always possible.)

3.3.1 Extended Union

Extended Union of RKS

To be able to take the union of two structures, the schemes R_1 and R_2 of these structures must be equal. We do not need restructuring, i.e., the scheme of the resultant structure is also equal to R_1 and R_2. The *extended union* is defined by RKS as follows:

Let X range over the zero-order names in E_{R_1} and Y range over the higher-order names in E_{R_1}. Then,

$$
\begin{aligned}
r_1 \cup^e r_2 = \{t \mid & (\exists t_1 \in r_1, \exists t_2 \in r_2: \\
& (\forall X, Y \in E_{R_1}: t[X]=t_1[X]=t_2[X] \wedge t[Y] = (t_1[Y] \cup^e t_2[Y]))) \\
& \vee (t \in r_1 \wedge (\forall t' \in r_2 : (\forall X \in E_{R_1}: t[X] \neq t'[X]))) \\
& \vee (t \in r_2 \wedge (\forall t' \in r_1 : (\forall X \in E_{R_1}: t[X] \neq t'[X])))\}
\end{aligned}
$$

This definition of [6] should be corrected as follows:

r_1

A	B		
	C	D	
		E	F
a_1	c_1	e_1	f_1
		e_2	f_2
	c_2	e_3	f_3
a_2	c_3	e_4	f_4

r_2

A	B		
	C	D	
		E	F
a_1	c_1	e_1	f_1
		e_7	f_7
	c_4	e_4	f_4
a_3	c_5	e_5	f_5

Figure 3.5: Purely hierarchical relations

$r_1 \cup^e r_2$

A	B		
	C	D	
		E	F
a_1	c_1	e_1	f_1
		e_2	f_2
		e_7	f_7
	c_2	e_3	f_3
	c_4	e_4	f_4
a_2	c_3	e_4	f_4
a_3	c_5	e_5	f_5

$\mu_B(\mu_D(r_1 \cup^e r_2))$

A	C	E	F
a_1	c_1	e_1	f_1
a_1	c_1	e_2	f_2
a_1	c_1	e_7	f_7
a_1	c_2	e_3	f_3
a_1	c_4	e_4	f_4
a_2	c_3	e_4	f_4
a_3	c_5	e_5	f_5

Figure 3.6: Extended union of r_1 and r_2

$$\begin{aligned} r_1 \cup^e r_2 = \{t \mid & (\exists t_1 \in r_1, \exists t_2 \in r_2: \\ & (\forall X, Y \in E_{R_1}: t[X]=t_1[X]=t_2[X] \wedge t[Y] = (t_1[Y] \cup^e t_2[Y]))) \\ & \vee (t \in r_1 \wedge (\forall t' \in r_2 : (\exists X \in E_{R_1}: t[X] \neq t'[X]))) \\ & \vee (t \in r_2 \wedge (\forall t' \in r_1 : (\exists X \in E_{R_1}: t[X] \neq t'[X])))\} \end{aligned}$$

The examples of *extended union* in [6] are interpreted with respect to this corrected definition. If they were interpreted with respect to the original RKS definition, it would not be possible to obtain the results in [6]. In the following examples the corrected *extended union* definition is applied to the relations r_1 and r_2 in Figure 3.5 . The result $r_1 \cup^e r_2$ and the flat form of this result $\mu_B(\mu_D(r_1 \cup^e r_2))$ are shown in Figure 3.6 . If we compare the flattened result with the *desired-result* that is found in Figure 3.7, we see that they are equal.

$\mu_B(\mu_D(r_1))$

A	C	E	F
a_1	c_1	e_1	f_1
a_1	c_1	e_2	f_2
a_1	c_2	e_3	f_3
a_2	c_3	e_4	f_4

$\mu_B(\mu_D(r_2))$

A	C	E	F
a_1	c_1	e_1	f_1
a_1	c_1	e_7	f_7
a_1	c_4	e_4	f_4
a_3	c_5	e_5	f_5

$\mu_B(\mu_D(r_1)) \cup \mu_B(\mu_D(r_2))$

A	C	E	F
a_1	c_1	e_1	f_1
a_1	c_1	e_2	f_2
a_1	c_1	e_7	f_7
a_1	c_2	e_3	f_3
a_1	c_4	e_4	f_4
a_2	c_3	e_4	f_4
a_3	c_5	e_5	f_5

Figure 3.7: The *desired-result*

Although it is not mentioned in [6], the *extended union* operator produces correct results for only nested relations that are *purely hierarchical.* A purely hierarchical relation is a nested relation with n nesting levels, $n \in \mathrm{N}^+$, for all nesting depths $i, 1 \leq i \leq n$, $|HA_i| = 1$, where HA_i is the set of higher-order attributes in the relation structure of the i^{th} nesting-level. If a nested relation is not purely hierarchical (i.e., if it contains more than one higher-order attributes in at least one of the nesting levels), the *extended union* operator introduces some irrelevant tuples.

Example: Let us show the validity of our last remark with an example. r_1, r_2, $r_1 \cup^e r_2$, $\mu_X(\mu_Y(r_1 \cup^e r_2))$, $\mu_X(\mu_Y(r_1))$, $\mu_X(\mu_Y(r_2))$, and $\mu_X(\mu_Y(r_1)) \cup \mu_X(\mu_Y(r_2))$ are shown in Figures 3.8, 3.9, and 3.10. $\mu_X(\mu_Y(r_1 \cup^e r_2))$ includes some irrelevant tuples, e.g., $< a_2b_7k_7c_3d_3 >$ and $< a_2b_8k_8c_2d_2 >$, which are neither in $\mu_X(\mu_Y(r_1))$ nor in $\mu_X(\mu_Y(r_2))$. As a result, the *extended union* operator of [6] is not information equivalent. □

The class of PNF relations is closed under *extended union* of [6] which is stated as a theorem (**Theorem 6.1**) in [6]. This theorem states that the structure $\mathcal{R}_3 = < R, r_3 >$ is in PNF, given that the structures $\mathcal{R}_1 = < R, r_1 >$ and $\mathcal{R}_2 = < R, r_2 >$ are in PNF. We think that the PNF restriction on the resultant structure makes the *extended union* definition non information equivalent. Dropping this restriction on the resultant relation structures provides us with a new definition for *extended union.* The class of PNF relations is not closed under the new *extended union.*

r_1

A	X		Y	
	B	K	C	D
a_1	b_1	k_1	c_1	d_1
	b_2	k_2		
a_2	b_1	k_1	c_1	d_1
	b_7	k_7	c_2	d_2

r_2

A	X		Y	
	B	K	C	D
a_2	b_1	k_1	c_1	d_1
	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

Figure 3.8: Examples for $\neg$*purely hierarchical* relations

$r_1 \cup^e r_2$

A	X		Y	
	B	K	C	D
a_1	b_1	k_1	c_1	d_1
	b_2	k_2		
a_2	b_1	k_1	c_1	d_1
	b_7	k_7	c_2	d_2
	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

$\mu_X(\mu_Y(r_1 \cup^e r_2))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_1	k_1	c_3	d_3
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2
a_2	b_7	k_7	c_3	d_3
a_2	b_8	k_8	c_1	d_1
a_2	b_8	k_8	c_2	d_2
a_2	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

Figure 3.9: Extended union of r_1 and r_2

$\mu_X(\mu_Y(r_1))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2

$\mu_X(\mu_Y(r_2))$

A	B	K	C	D
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_3	d_3
a_2	b_8	k_8	c_1	d_1
a_2	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

$\mu_X(\mu_Y(r_1)) \cup \mu_X(\mu_Y(r_2))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_1	k_1	c_3	d_3
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2
a_2	b_8	k_8	c_1	d_1
a_2	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

Figure 3.10: The *desired-result*

$r_1 \cup^e r_{2(1)}$

A	X		Y	
	B	K	C	D
a_1	b_1 b_2	k_1 k_2	c_1	d_1
a_2	b_1	k_1	c_1 c_2 c_3	d_1 d_2 d_3
a_2	b_7	k_7	c_1 c_2	d_1 d_2
a_2	b_8	k_8	c_1 c_3	d_1 d_3
a_4	b_4	k_4	c_4	d_4

$r_1 \cup^e r_{2(2)}$

A	X		Y	
	B	K	C	D
a_1	b_1 b_2	k_1 k_2	c_1	d_1
a_2	b_1 b_7	k_1 k_7	c_2	d_2
a_2	b_1 b_8	k_1 k_8	c_3	d_3
a_2 a_2 a_2	b_1 b_7 b_8	k_1 k_7 k_8	c_1	d_1
a_4	b_4	k_4	c_4	d_4

Figure 3.11: $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$

$\mu_X(\mu_Y(r_1 \cup^e r_2))_{(1)}$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_1	k_1	c_3	d_3
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2
a_2	b_8	k_8	c_1	d_1
a_2	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

$\mu_X(\mu_Y(r_1 \cup^e r_2))_{(2)}$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_2	d_2
a_2	b_1	k_1	c_3	d_3
a_2	b_8	k_8	c_3	d_3
a_2	b_1	k_1	c_1	d_1
a_2	b_7	k_7	c_1	d_1
a_2	b_8	k_8	c_1	d_1
a_4	b_4	k_4	c_4	d_4

Figure 3.12: Flat forms of $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$

Extended Union of AB

Before defining the new *extended union*, let us go through the *extended union* of [2].

Let f be a format and I, J two instances over f. Then the *union* of I and J is the instance over f, denoted $I \oplus J$, defined by:

1. If $f \equiv X$, where X is nonempty, then $I \oplus J = I \cup J$, and
2. If $f \equiv X(f_1)^* \ldots (f_n)^*$, where $f_1, \ldots, f_n$ are nonempty, then:

$$
\begin{aligned}
I \oplus J = & \left\{ < u(I_1 \oplus J_1) \ldots (I_n \oplus J_n) > \;\middle|\; \begin{array}{c} < uI_1 \ldots I_n > \in I \text{ and} \\ < uJ_1 \ldots J_n > \in J \end{array} \right\} \\
\cup & \left\{ < uI_1 \ldots I_n > \;\middle|\; \begin{array}{c} < uI_1 \ldots I_n > \in I, \text{ and} \\ \forall J_1 \ldots J_n, < uJ_1 \ldots J_n > \notin J \end{array} \right\} \\
\cup & \left\{ < uJ_1 \ldots J_n > \;\middle|\; \begin{array}{c} < uJ_1 \ldots J_n > \in J \text{ and} \\ \forall I_1 \ldots I_n, < uI_1 \ldots I_n > \notin I \end{array} \right\}
\end{aligned}
$$

The *extended union* of [2] is similar to that of [6] and produces the same results with the previous examples; the tuples that agree on their atomic attributes are combined to form a new tuple. It produces correct results only for purely hierarchical relations (and therefore it is not information equivalent).

The New Extended Union

In the following *extended union* definition, HA is the set of all higher-order names in E_R, and HA_{Y_i} is the set of all higher-order names in E_{Y_i}. X ranges over the zero-order names, while Y ranges over the higher-order names in E_R. Given two relation structures $\mathcal{R}_1 = < R, r_1 >$ and $\mathcal{R}_2 = < R, r_2 >$ in PNF, the *extended union* with the structure $\mathcal{R}_3 = < R, r_1 \cup^e r_2 >$ is defined as follows at the instance level:

$$
\begin{aligned}
r_1 \cup^e r_2 \;\; = \;\; & \{ t \mid (\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
& \quad (\forall X, Y \in E_{R_1}, |HA| \leq 1 : t[X] = t_1[X] = t_2[X] \\
& \quad\quad \wedge t[Y] = (t_1[Y] \cup^e t_2[Y]))) \\
& \bigvee (\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
& \quad (\forall X, \exists Y_i \in E_{R_1}, 1 \leq i \leq |HA|, |HA| > 1 : (\exists Y_j \in (HA - \{Y_i\}) :
\end{aligned}
$$

$$
\begin{aligned}
&t_1[Y_j] \neq t_2[Y_j]) \wedge t[X] = t_1[X] = t_2[X] \\
&\wedge t[Y_i] = \{t_y | (\exists t'_{y_i} \in t_1[Y_i] : t_y = t'_{y_i} \wedge (\forall t''_{y_i} \in t_2[Y_i] : \\
&\qquad (\exists X \in E_{Y_i} : t'_{y_i}[X] \neq t''_{y_i}[X]))\} \\
&\wedge t[HA - \{Y_i\}] = t_1[HA - \{Y_i\}])) \\
\bigvee &(\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
&(\forall X, \exists Y_i \in E_{R_1}, 1 \leq i \leq |HA|, |HA| > 1 : (\exists Y_j \in (HA - \{Y_i\}) : \\
&t_1[Y_j] \neq t_2[Y_j]) \wedge t[X] = t_1[X] = t_2[X] \\
&\wedge t[Y_i] = \{t_y | (\exists t'_{y_i} \in t_2[Y_i] : t_y = t'_{y_i} \wedge (\forall t''_{y_i} \in t_1[Y_i] : \\
&\qquad (\exists X \in E_{Y_i} : t'_{y_i}[X] \neq t''_{y_i}[X]))\} \\
&\wedge t[HA - \{Y_i\}] = t_2[HA - \{Y_i\}])) \\
\bigvee &(\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
&(\forall X, \exists Y_i \in E_{R_1}, 1 \leq i \leq |HA|, |HA| > 1 : (\exists Y_j \in (HA - \{Y_i\}) : \\
&t_1[Y_j] \neq t_2[Y_j]) \wedge t[X] = t_1[X] = t_2[X] \wedge X_{Y_i} =_{def} \{X | X \in E_{Y_i}\} \\
&\wedge t[X_{Y_i}] = \{t_{y_i} | (\exists t'_{y_i} \in t_1[Y_i], \exists t''_{y_i} \in t_2[Y_i] : \\
&\qquad (\forall X \in E_{Y_i} : t_{y_i}[X] = t'_{y_i}[X] = t''_{y_i}[X]))\} \\
&\wedge HA =_{def} (HA - \{Y_i\}) \cup HA_{Y_i} \\
&\wedge[(|HA| > 1 : t[HA] \in (t_1[HA] \cup^e t_2[HA])) \\
&\quad \vee(|HA| \leq 1 : t[HA] = (t_1[HA] \cup^e t_2[HA]))])) \\
\bigvee &(\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
&(\forall X \in E_{R_1}, 1 \leq i \leq |HA|, |HA| > 1 : t[X] = t_1[X] = t_2[X] \\
&\wedge(\forall Y_j \in (HA - \{Y_i\}) : t_1[Y_j] = t_2[Y_j] \wedge t[Y_j] = t_1[Y_j]) \\
&\wedge t[Y_i] = (t_1[Y_i] \cup^e t_2[Y_i]))) \\
\bigvee &(t \in r_1 \wedge (\forall t' \in r_2 : (\exists X \in E_{R_1} : t[X] \neq t'[X]))) \\
\bigvee &(t \in r_2 \wedge (\forall t' \in r_1 : (\exists X \in E_{R_1} : t[X] \neq t'[X]))))\}
\end{aligned}
$$

Example: When the new *extended union* operator is applied to the relations r_1 and r_2 (Figure 3.8), it is possible to obtain the results $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$ (Figure 3.11). If we compare the flattened forms $\mu_X(\mu_Y(r_1 \cup^e r_2))_{(1)}$ and $\mu_X(\mu_Y(r_1 \cup^e r_2))_{(2)}$ (Figure 3.11) of $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$ with the desired-result (Figure 3.10), we notice that these three are equal. The difference between $r_1 \cup^e r_{2(1)}$ and $r_1 \cup^e r_{2(2)}$ is because of different *permutations* of Y_i's in the above *extended union* definition. Y_i's can be selected randomly among the higher-order names in E_R. We have n permutations of Y_i's with n higher-order names (that is, $r_1 \cup^e r_2$ can be represented in n different formats). This is an expected result once we remember that $r_1 \cup^e r_2$ is not in PNF and *nest* is not an inverse operator for *unnest* in this case. □

Theorem 3.1 *The extended union operator is information equivalent.*
Proof The proof has several cases:

1. $|HA| = 0$ (flat relations).
2. nesting-depth $= n$ ($\in N^+$), for all nesting-depth, i, $1 \leq i \leq n$: $|HA| = 1$ (purely hierarchical relations).
3. $|HA| > 1$, and each higher-order attribute Y in E_R is a flat relation.
4. $|HA| = n$ ($\in N^+$) and $\exists Y \in E_R : |HA_Y| = m$ ($\in N^+$).

(1) In this case r_1 and r_2 are flat relations, so we show that $r_1 \cup^e r_2 = r_1 \cup r_2$.

$\subseteq$ *part:* If $t \in r_1 \cup^e r_2$, then t satisfies one of the following three disjuncts of the $\cup^e$ definition:

(a) $(t \in r_1 \wedge (\forall t' \in r_2 : (\exists X \in E_{R1} : t[X] \neq t'[X])))$
(b) $(t \in r_2 \wedge (\forall t' \in r_1 : (\exists X \in E_{R1} : t[X] \neq t'[X])))$
(c) $(\exists t_1 \in r_1, \exists t_2 \in r_2 : (\forall X, Y \in E_{R1}, |HA| \leq 1 : t[X] = t_1[X] = t_2[X]$
$\wedge t[Y] = (t_1[Y] \cup^e t_2[Y])))$
(since $|HA| = 0$, there is no higher-order attribute and there is no $t[Y]$)

If t satisfies the first disjunct, then $t \in r_1$ only, the second, then $t \in r_2$ only, and the third, then $t \in r_1$, or r_2, or in both. It is obvious that $t \in r_1 \cup r_2$ in any of these three cases, therefore $r_1 \cup^e r_2 \subseteq r_1 \cup r_2$.

$\supseteq$ *part:* Let $t \in r_1 \cup r_2$, then t is either in:
(a) r_1 only, or
(b) r_2 only, or
(c) r_1 and r_2.

Since three disjuncts mentioned in the $\subseteq$ part of the proof include all those tuples either only in r_1, or only in r_2, or in both, a tuple t in $r_1 \cup r_2$ will be in $r_1 \cup^e r_2$. Therefore $r_1 \cup^e r_2 \supseteq r_1 \cup r_2$.

(2) In this case we show that

$$\begin{aligned}&\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))\\ &\qquad = \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)),\end{aligned}$$

where Y_i is the higher-order attribute of the i^{th} nesting level. The proof is by induction on the nesting-depth n.

Basis: We show that $\mu_Y(r_1 \cup^e r_2) = \mu_Y(r_1) \cup \mu_Y(r_2)$, where $n = 1$ and $Y = X_1 \ldots X_m$.

$\supseteq$ *part:* We show that if $t \in \mu_Y(r_1) \cup \mu_Y(r_2)$, then $t \in \mu_Y(r_1 \cup^e r_2)$. $\mu_Y(r_1)$ and $\mu_Y(r_2)$ are flat relations, so t is either only in $\mu_Y(r_1)$, or only in $\mu_Y(r_2)$, or in both, and it's either unnested from some u_1 in r_1, or some u_2 in r_2, or some u_3 in both. We can say that $t[X_1 \ldots X_m] \in u_1[Y] \vee t[X_1 \ldots X_m] \in u_2[Y]$. In the *extended union* of r_1 and r_2, u_1 and u_2 will be included either as two distinct tuples, or as a tuple u, where $u[Y] = u_1[Y] \cup^e u_2[Y]$. Obviously t will be included in $\mu_Y(r_1 \cup^e r_2)$ in any case.

$\subseteq$ *part:* We show that if $t \in \mu_Y(r_1 \cup^e r_2)$, then $t \in \mu_Y(r_1) \cup \mu_Y(r_2)$. If we partition $\mu_Y(r_1 \cup^e r_2)$ on $E_R - X_1 \ldots X_m$ and obtain the partitions $u_1, \ldots, u_k$, then we must show that all tuples $t_1, \ldots t_n$ in any partition of $\mu_Y(r_1 \cup^e r_2)$ are in $\mu_Y(r_1) \cup \mu_Y(r_2)$. The tuples $t_1, \ldots, t_n$ are obtained by unnesting the set of tuples $u_1, \ldots, u_k$, each of which is a partition on $E_R - Y$ in $r_1 \cup^e r_2$. This means that for all i, $1 \leq i \leq n$, $\exists j$, $1 \leq j \leq k$, such that $t_i[X_1 \ldots X_m] \in u_j[Y]$, and $\bigcup_{j=1}^{k} u_j[Y] = \{t_i[X_1 \ldots X_m] \mid 1 \leq i \leq n\}$. Each u_j is created by the extended union of two tuples, $u_j{}^1 \in r_1$ and $u_j{}^2 \in r_2$. Since Y is a flat relation, $\bigcup_{j=1}^{k} u_j[Y] \in (\bigcup_{j=1}^{k} u_j{}^1[Y] \vee \bigcup_{j=1}^{k} u_j{}^2[Y])$. When the tuples $u_j{}^1$ and $u_j{}^2$ are unnested into tuples $v_l{}^1$, $(1 \leq i \leq p_1)$ and $v_l{}^2$, $(1 \leq l \leq p_2)$, we have $\bigcup_{j=1}^{k} u_j{}^1[Y] = \{v_l{}^1[X_1 \ldots X_m] \mid 1 \leq l \leq p_1\}$ and $\bigcup_{j=1}^{k} u_j{}^2[Y] = \{v_l{}^2[X_1 \ldots X_m] \mid 1 \leq l \leq p_2\}$, and we can say $\{t_i[X_1 \ldots X_m] \mid 1 \leq i \leq n\} \subseteq \{v_l{}^1[X_1 \ldots X_m] \mid 1 \leq l \leq p_1\} \cup \{v_l{}^2[X_1 \ldots X_m] \mid 1 \leq l \leq p_2\}$. Therefore $\mu_Y(r_1) \cup \mu_Y(r_2)$ contains all the tuples

$t_1, \ldots, t_n$ in $\mu_Y(r_1 \cup^e r_2)$.

Induction Step: By the induction hypothesis, we know that

$$\mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup^e \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_2))\ldots))$$

for the first $(n-1)$ nesting levels, where Y_i is the higher-order attribute at the i^{th} nesting level, $1 \le i \le n-1$. We now show that this is also true for n nesting levels. If we unnest both sides of the previous equation with Y_n, we obtain

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{Y_n}[\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots) \cup^e \mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)]$$

Let $r_1' = \mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)$ and $r_2' = \mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)$,

now we have

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots)) = \mu_{Y_n}(r_1' \cup^e r_2').$$

Since r_1' and r_2' are relations whose nesting-depths are 1, $\mu_{Y_n}(r_1' \cup^e r_2') = \mu_{Y_n}(r_1') \cup \mu_{Y_n}(r_2')$, which is proved to be true in the basis step. If we substitute r_1' and r_2' by their equivalents, we will have

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots))$$

(3) In this case we show that

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots))$$

The proof is by induction on the number of the higher-order attributes at the first and only nesting level.

Basis: We show that $\mu_{Y_1}(\mu_{Y_2}(r_1 \cup^e r_2)) = \mu_{Y_1}(\mu_{Y_2}(r_1)) \cup \mu_{Y_1}(\mu_{Y_2}(r_2))$, where $|HA| = 2$ and $Y_1 = X_1 \ldots X_m$, $Y_2 = X_l \ldots X_k$.

$\supseteq$ *part:* We show that if $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) \cup \mu_{Y_1}(\mu_{Y_2}(r_2))$, then $t \in \mu_{Y_1}(\mu_{Y_2}(r_1 \cup^e r_2))$. Since $\mu_{Y_1}(\mu_{Y_2}(r_1))$ and $\mu_{Y_1}(\mu_{Y_2}(r_2))$ are flat relations, t is only in $\mu_{Y_1}(\mu_{Y_2}(r_1))$, or only in $\mu_{Y_1}(\mu_{Y_2}(r_2))$, or in both. So t is unnested from some $u_1 \in r_1$, or $u_2 \in r_2$, or u_3 in r_1 and r_2. Then we can say that $(t[X_1 \ldots X_m] \in u_1[Y_1] \wedge t[X_l \ldots X_k] \in u_1[Y_2]) \vee (t[X_1 \ldots X_m] \in u_2[Y_1] \wedge t[X_l \ldots X_k] \in u_2[Y_2])$. In the *extended union* of r_1 and r_2, u_1 and u_2 will be included either as two distinct tuples, or as a new tuple (formed by u_1 and u_2). In any case, t is in the unnested form of the tuple, therefore $t \in \mu_{Y_1}(\mu_{Y_2}(r_1 \cup^e r_2))$.

$\subseteq$ *part:* We show that if $t \in \mu_{Y_1}(\mu_{Y_2}(r_1 \cup^e r_2))$, then $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) \cup \mu_{Y_1}(\mu_{Y_2}(r_2))$. In this case, t must be unnested from some u in $r_1 \cup^e r_2$, and $t \in \mu_{Y_1}(\mu_{Y_2}(u))$. Since $u \in r_1 \cup^e r_2$, u satisfies one of the disjuncts in the $\cup^e$ definition. Each of these disjuncts includes those tuples either only in r_1, or only in r_2, or in both. Then $\mu_{Y_1}(\mu_{Y_2}(u))$ is either:

(i) $\mu_{Y_1}(\mu_{Y_2}(u)) \subseteq \mu_{Y_1}(\mu_{Y_2}(r_1))$, or

(ii) $\mu_{Y_1}(\mu_{Y_2}(u)) \subseteq \mu_{Y_1}(\mu_{Y_2}(r_2))$, or

(iii) $\mu_{Y_1}(\mu_{Y_2}(u)) \subseteq \mu_{Y_1}(\mu_{Y_2}(r_2))$, and $\mu_{Y_1}(\mu_{Y_2}(u)) \subseteq \mu_{Y_1}(\mu_{Y_2}(r_2))$

From (i), (ii), and (iii), $\mu_{Y_1}(\mu_{Y_2}(u)) \subseteq \mu_{Y_1}(\mu_{Y_2}(r_1)) \cup \mu_{Y_1}(\mu_{Y_2}(r_2))$. Since we know that $t \in \mu_{Y_1}(\mu_{Y_2}(u))$, then $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) \cup \mu_{Y_1}(\mu_{Y_2}(r_2))$.

Induction Step: By the induction hypothesis, we know that

$$\begin{aligned}&\mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))\\ &\quad = \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup^e \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_2))\ldots)),\end{aligned}$$

for the first $(n-1)$ higher-order attributes of E_R, where $n \geq 3$. Now we show that this is also true for n:

$$\begin{aligned}&\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))\\ &\quad = \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup^e \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)).\end{aligned}$$

The proof is similar to the induction step of case (2). If $\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e$

$r_2))\ldots)$ is unnested with Y_n and r_1' and r_2' are substituted as in *case (ii)*, we obtain,

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots)) = \mu_{Y_n}(r_1' \cup^e r_2').$$

Since r_1' and r_2' are relations which have one higher-order attribute and one nesting level, $\mu_{Y_n}(r_1' \cup^e r_2') = \mu_{Y_n}(r_1') \cup \mu_{Y_n}(r_2')$, which is proved to be true in the basis of case(2). Therefore

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) \cup \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)).$$

(4) This is the most general case of a nested relation, viz. a nested relation with n higher-order attributes, each of which is also a nested relation with a finite number of higher-order attributes and nesting levels.

We show that the *extended union* operator is information equivalent with this kind of relation structures in several steps. Using a recursive procedure, we obtain the most general nested structure and show that the *extended union* operator is information equivalent to this structure.

Now let the relation structures of r_1 and r_2 have $n \in N^+$ higher-order attributes, where each has a relation structure which is equal to that of (1), (2), or (3) and let this new structure be (4.a). To show that *extended union* is information equivalent in this case, we show that

$$\mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots)) \cup \mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))$$

where S_{Y_i} is the unnest sequence (a set of higher-order names in E_{Y_i}) required to flatten the i^{th} higher-order attribute in E_R.

The proof is by induction on the number of higher-order attributes in E_R.

Basis: In this case, $|HA| = 1$ and there's only one higher-order attribute in E_R. The structure of this higher-order attribute is equal to that of (1), (2), or (3). Since we've shown that the *extended union* operator is information equivalent with the structures of (1), (2), and (3), we conclude that

$$\mu_Y(\mu_{S_Y}(r_1 \cup^e r_2)) = \mu_Y(\mu_{S_Y}(r_1)) \cup \mu_Y(\mu_{S_Y}(r_2))$$

Induction Step: By the induction hypothesis we know that

$$\mu_{(Y_{n-1},Y_{n-2},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))\cup^e \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))$$

for the first $(n-1)$ higher-order attributes of E_R. We now show that this is also true for all the higher-order attributes of E_R, which is stated as follows:

$$\mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))\cup^e \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))$$

If we nest both sides of the equality introduced by the induction hypothesis with Y_n and S_{Y_n}, we obtain

$$\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots)))) = \mu_{Y_n}(\mu_{S_{Y_n}}$$
$$[\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))\cup^e \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))])$$

Let $r_1{}' = \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))$ and
$r_2{}' = \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))$

If we replace $\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))$ and $\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))$ with $r_1{}'$ and $r_2{}'$ respectively, we have

$$\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots)))) = \mu_{Y_n}(\mu_{S_{Y_n}}(r_1{}' \cup^e r_2{}')$$

The structure of $r_1{}'$ and $r_2{}'$ contains one higher-order attribute which is in one of the forms (1), (2), or (3). Since it is shown in the basis step that *extended union* is information equivalent to the structures of (1), (2), and (3), we conclude that

$$\mu_{Y_n}(\mu_{S_{Y_n}}(r_1{}' \cup^e r_2{}') = \mu_{Y_n}(\mu_{S_{Y_n}}(r_1{}')) \cup \mu_{Y_n}(\mu_{S_{Y_n}}(r_2{}'))$$

Using this equation, we obtain the following equality:

$$\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots))))$$
$$= \mu_{Y_n}(\mu_{S_{Y_n}}(r_1')) \cup \mu_{Y_n}(\mu_{S_{Y_n}}(r_2')),$$

If r_1' and r_2' are substituted with their equivalents, we obtain

$$\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots))))$$
$$= \mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_1))\ldots)))) \cup$$
$$\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_2))\ldots))))$$

By **Theorem 8.1.b** of RKS, given a relation structure $\mathcal{R}$, the following property holds: $\mu_A(\mu_B(\mathcal{R})) = \mu_B(\mu_A(\mathcal{R}))$. With respect to this theorem, the order of *unnest* is not important, so we can reorganize the previous equality by changing the unnest sequence and obtain the following:

$$\mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\cdots(\mu_{S_{Y_1}}(r_1 \cup^e r_2))\ldots))$$
$$= \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\cdots(\mu_{S_{Y_1}}(r_1))\ldots)) \cup \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\cdots(\mu_{S_{Y_1}}(r_2))\ldots)) \quad \square$$

3.3.2 Extended Difference

Extended Difference of RKS

Difference is similar to *union* in the sense that it does not need restructuring of the relation structures. To be able to find the difference of two structures $\mathcal{R}_1 = < R_1, r_1 >$ and $\mathcal{R}_2 = < R_2, r_2 >$, their schemes R_1 and R_2 must be equal. The structure of the resultant relation is $< R_3, r_1 -^e r_2 >$, where R_3 is equal to R_1 and R_2. The *extended difference* is defined by RKS as follows.

Let X range over the zero-order names in E_{R_1} and Y range over the higher-order names in E_{R_1}. Then,

$r_1 -^e r_2 = \{t \mid (\exists t_1 \in r_1 \land \exists t_2 \in r_2 \land \exists Y \in E_R :$

$$(\forall X, Y \in E_{R_1}\colon t[X] = t_1[X] = t_2[X]$$
$$\wedge\ t[Y] = (t_1[Y] -^e t_2[Y]) \wedge t[Y] \neq \emptyset))$$
$$\vee\ (t \in r_1 \wedge (\exists t' \in r_2 : (\forall X \in E_{R_1}\colon t[X] \neq t'[X]))))\}$$

This definition of [6] should be corrected as follows:

$$r_1 -^e r_2 = \{t \mid (\exists t_1 \in r_1 \wedge \exists t_2 \in r_2 \wedge \exists Y \in E_R :$$
$$(\forall X, Y \in E_{R_1}\colon t[X] = t_1[X] = t_2[X]$$
$$\wedge\ t[Y] = (t_1[Y] -^e t_2[Y]) \wedge t[Y] \neq \emptyset))$$
$$\vee\ (t \in r_1 \wedge (\forall t' \in r_2 : (\exists X \in E_{R_1}\colon t[X] \neq t'[X]))))\}$$

The examples of *extended difference* in [6] are interpreted with respect to this corrected definition. If they were interpreted with respect to the original definition of RKS, it would not be possible to obtain the results in [6].

Example: In the following the corrected *extended difference* definition of [6] is applied to the relations r_1 and r_2 (Figure 3.5). The result $r_1 -^e r_2$ and the flat form of this result $\mu_B(\mu_D(r_1 -^e r_2))$ are shown in Figure 3.13. If we compare the flattened result with the desired-result (Figure 3.14), we see that they are equal.□

Although it is not mentioned in [6], the *extended difference* operator produces correct results for only nested relations that are purely hierarchical as the *extended union* operator does. If a nested relation is not purely hierarchical, then the *extended difference* operator loses some of the tuples that must be in the result.

Example: Now let us illustrate this last claim. *Extended difference* operator is applied to the relations in Figure 3.8. $r_1 -^e r_2, \mu_X(\mu_Y(r_1 -^e r_2))$, $\mu_X(\mu_Y(r_1))$, $\mu_X(\mu_Y(r_2))$, and $\mu_X(\mu_Y(r_1)) - \mu_X(\mu_Y(r_2))$ are shown in Figures 3.15 and 3.16. $\mu_X(\mu_Y(r_1 -^e r_2))$ loses some tuples that's in *desired-result*, e.g. $< a_2 b_1 k_1 c_2 d_2 >$ and $< a_2 b_7 k_7 c_1 d_1 >$ which are in $\mu_X(\mu_Y(r_1))$ but not in $\mu_X(\mu_Y(r_2))$. As a result, the *extended difference* operator of [6] is not information equivalent.□

The class of PNF relations is closed under *extended difference* of [6] which

$r_1 -^e r_2$

A	B		
	C	D	
		E	F
a_1	c_1	e_2	f_2
	c_2	e_3	f_3
a_2	c_3	e_4	f_4

$\mu_B(\mu_D(r_1 -^e r_2))$

A	C	E	F
a_1	c_1	e_2	f_2
a_1	c_2	e_3	f_3
a_2	c_3	e_4	f_4

Figure 3.13: Extended difference of r_1 and r_2

$\mu_B(\mu_D(r_1))$

A	C	E	F
a_1	c_1	e_1	f_1
a_1	c_1	e_2	f_2
a_1	c_2	e_3	f_3
a_2	c_3	e_4	f_4

$\mu_B(\mu_D(r_2))$

A	C	E	F
a_1	c_1	e_1	f_1
a_1	c_1	e_7	f_7
a_1	c_4	e_4	f_4
a_3	c_5	e_5	f_5

$\mu_B(\mu_D(r_1)) - \mu_B(\mu_D(r_2))$

A	C	E	F
a_1	c_1	e_2	f_2
a_1	c_2	e_3	f_3
a_2	c_3	e_4	f_4

Figure 3.14: The *desired-result*

$r_1 -^e r_2$

A	X		Y	
	B	K	C	D
a_1	b_1	k_1	c_1	d_1
	b_2	k_2		
a_2	b_7	k_7	c_2	d_2

$\mu_X(\mu_Y(r_1 -^e r_2))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_7	k_7	c_2	d_2

Figure 3.15: Extended difference of r_1 and r_2

$\mu_X(\mu_Y(r_1))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2

$\mu_X(\mu_Y(r_2))$

A	B	K	C	D
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_3	d_3
a_2	b_8	k_8	c_1	d_1
a_2	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

$\mu_X(\mu_Y(r_1)) - \mu_X(\mu_Y(r_2))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2

Figure 3.16: The *desired-result*

is stated as a theorem (**Theorem 6.1**) in [6]. This theorem states that the structure $\mathcal{R}_3 = < R, r_1 -^e r_2 >$ is in PNF, given that the structures $\mathcal{R}_1 = < R, r_1 >$ and $\mathcal{R}_2 = < R, r_2 >$ are in PNF. We think that the PNF restriction on the resultant structure makes the *extended difference* definition non information equivalent as in *extended union*. Dropping this restriction on the resultant relation structures provides us with a new *extended difference*. The class of PNF relations is not closed under the new *extended difference*.

Extended Difference of AB

Before defining the new *extended difference* operator, let us go through the *extended difference* of [2].

Let f be a format and I, J two instances over f. Then the *difference* of I and J is the instance over f, denoted $I \ominus J$, defined by:

1. if $f \equiv X$, where X is nonempty, then $I \ominus J = I - J$, and
2. if $f \equiv X(f_1)^* \dots (f_n)^*$, where $f_1, \dots, f_n$ are nonempty, then :

$$I \ominus J = \left\{ < u(I_1 \ominus J_1) \dots (I_n \ominus J_n) > \;\middle|\; \begin{array}{l} < uI_1 \dots I_n > \in I \text{ and} \\ < uJ_1 \dots J_n > \in J \text{ and} \\ \text{for some } i, \; I_i \ominus J_i \neq \emptyset \end{array} \right\}$$
$$\cup \left\{ < uI_1 \dots I_n > \;\middle|\; \begin{array}{c} < uI_1 \dots I_n > \in I \text{ and} \\ \forall J_1 \dots J_n, < uJ_1 \dots J_n > \notin J \end{array} \right\}$$

The *extended difference* of [2] is similar to that of [6] and produces the same results with the previous examples. It produces correct results only for purely hierarchical relations, therefore it's not information equivalent.

The New Extended Difference

In the following *extended difference* definition, HA, E_{Y_i}, HA_{Y_i}, and X represent the same things as they do in the new *extended union* definition. Given two relation structures $\mathcal{R}_1 = < R, r_1 >$ and $\mathcal{R}_2 = < R, r_2 >$ in PNF, the *extended difference* with the structure $< R, r_1 -^e r_2 >$ is defined as follows at the instance level:

$$r_1 -^e r_2 \;=\; \{t \mid (\exists t_1 \in r_1, \exists t_2 \in r_2 :$$

$$
\begin{aligned}
&(\forall X, Y \in E_{R_1}, |HA| \leq 1 : t[X] = t_1[X] = t_2[X] \\
&\qquad \wedge t[Y] = (t_1[Y] -^e t_2[Y]) \wedge t[Y] \neq \emptyset)) \\
\bigvee&(\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
&(\forall X, \exists Y_i \in E_{R_1}, 1 \leq i \leq |HA|, |HA| > 1 : t[X] = t_1[X] = t_2[X] \\
&\wedge t[Y_i] = \{t_y | (\exists t'_{y_i} \in t_1[Y_i] : t_y = t'_{y_i} \wedge (\forall t''_{y_i} \in t_2[Y_i] : \\
&\qquad (\exists X \in E_{Y_i} : t'_{y_i}[X] \neq t''_{y_i}[X])))\} \\
&\wedge t[HA - \{Y_i\}] = t_1[HA - \{Y_i\}])) \\
\bigvee&(\exists t_1 \in r_1, \exists t_2 \in r_2 : \\
&(\forall X, \exists Y_i \in E_{R_1}, 1 \leq i \leq |HA|, |HA| > 1 : t[X] = t_1[X] = t_2[X] \\
&\wedge X_{Y_i} =_{def} \{X | X \in E_{Y_i}\} \\
&\wedge t[X_{Y_i}] = \{t_{y_i} | (\exists t'_{y_i} \in t_1[Y_i], \exists t''_{y_i} \in t_2[Y_i] : \\
&\qquad (\forall X \in E_{Y_i} : t_{y_i}[X] = t'_{y_i}[X] = t''_{y_i}[X]))\} \\
&\wedge HA =_{def} (HA - \{Y_i\}) \cup HA_{Y_i} \\
&\wedge [(|HA| > 1 : t[HA] \in (t_1[HA] -^e t_2[HA]) \\
&\qquad \wedge (t_1[HA] -^e t_2[HA]) \neq \emptyset) \\
&\quad \vee (|HA| \leq 1 : t[HA] = (t_1[HA] -^e t_2[HA]) \wedge t[HA] \neq \emptyset)])) \\
\bigvee&(t \in r_1 \wedge (\forall t' \in r_2 : (\exists X \in E_{R_1} : t[X] \neq t'[X])))\}
\end{aligned}
$$

Example: When the newly defined *extended difference* operator is applied to the relations r_1 and r_2 in Figure 3.8, it is possible to obtain the results $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$ in Figure 3.17. If we compare the flattened forms $\mu_X(\mu_Y(r_1 -^e r_2))_{(1)}$ and $\mu_X(\mu_Y(r_1 -^e r_2))_{(2)}$ (Figure 3.18) of $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$ with the desired-result (Figure 3.16), we notice that these three are equal. The difference between $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$ is because of the same reason explained for *extended union.* □

Theorem 3.2 *The extended difference operator is information equivalent*
Proof The proof has several cases.

1. $|HA| = 0$ (flat relations).
2. nesting-depth $= n$ ($\in N^+$), for all nesting-depths i, $1 \leq i \leq n$: $|HA| = 1$ (purely hierarchical relations).

$r_1 -^e r_{2(1)}$

A	X		Y	
	B	K	C	D
a_1	b_1	k_1	c_1	d_1
	b_2	k_2		
a_2	b_7	k_7	c_1	d_1
			c_2	d_2
a_2	b_1	k_1	c_2	d_2

$r_1 -^e r_{2(2)}$

A	X		Y	
	B	K	C	D
a_1	b_1	k_1	c_1	d_1
	b_2	k_2		
a_2	b_1	k_1	c_2	d_2
	b_7	k_7		
a_2	b_7	k_7	c_1	d_1

Figure 3.17: $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$

$\mu_X(\mu_Y(r_1 -^e r_2))_{(1)}$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2

$\mu_X(\mu_Y(r_1 -^e r_2))_{(2)}$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_2	d_2
a_2	b_7	k_7	c_1	d_1

Figure 3.18: Flat forms of $r_1 -^e r_{2(1)}$ and $r_1 -^e r_{2(2)}$

3. $|HA| > 1$, and each higher-order attribute Y in E_R is a flat relation.

4. $|HA| = n$ ($\in \mathrm{N}^+$) and $\exists Y \in E_R : |HA_Y| = m$ ($\in \mathrm{N}^+$).

(1) In this case r_1 and r_2 are flat relations, so we show that $r_1 -^e r_2 = r_1 - r_2$.

$\subseteq$ *part:* Let $t \in r_1 -^e r_2$, then t can only satisfy the following disjunct of the $-^e$ definition: $(t \in r_1 \wedge (\forall t' \in r_2 : (\exists X \in E_{R_1} : t[X] \neq t'[X])))$. This disjunct states is that t is a tuple only in r_1, so t is obviously in $r_1 - r_2$.

$\supseteq$ *part:* Let $t \in r_1 - r_2$, then t is only in r_1, and there is at least one atomic attribute that differentiates t from all the tuples in r_2. If this statement is formalized, we obtain the disjunct of $-^e$ mentioned in the $\subseteq$ part. Since t satisfies a disjunct of $-^e$ definition, $t \in r_1 -^e r_2$

(2) In this case we show that

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))$$
$$= \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) - \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)),$$

where Y_i is the higher-order attribute of the i^{th} nesting level. The proof is by induction on the nesting-depth n.

Basis: We show that $\mu_Y(r_1 -^e r_2) = \mu_Y(r_1) - \mu_Y(r_2)$, where $n = 1$ and $Y = X_1 \ldots X_m$.

$\supseteq$ *part:* We show that if $t \in \mu_Y(r_1) -^e \mu_Y(r_2)$, then $t \in \mu_Y(r_1 -^e r_2)$. t is only in $\mu_Y(r_1)$ and it is unnested from some u_1 in r_1. Since t is not in $\mu_Y(r_2)$, t cannot be unnested from any u_2 in r_2. We can say that $t[X_1 \ldots X_m] \in u_1[Y]$ and $\forall u_2 \in r_2 : t[X_1 \ldots X_m] \notin u_2[Y]$. In the *extended difference* of r_1 and r_2, u_1 will be included either completely as u_1 or partially as a new tuple u, where $u[Y] = u_1[Y] -^e u_2[Y]$. In any case t will be included in $\mu_Y(r_1 -^e r_2)$.

$\subseteq$ *part:* We show that if $t \in \mu_Y(r_1 -^e r_2)$, then $t \in \mu_Y(r_1) - \mu_Y(r_2)$. If we partition $\mu_Y(r_1 -^e r_2)$ on $E_R - X_1 \ldots X_m$ and obtain the partitions $u_1, \ldots, u_k$, then we must show that all tuples $t_1, \ldots t_n$ in any partition of $\mu_Y(r_1 -^e r_2)$ are in $\mu_Y(r_1) - \mu_Y(r_2)$. The tuples $t_1, \ldots, t_n$ are obtained by unnesting the set of tuples

$u_1, \ldots, u_k$, each of which is a partition on $E_R - Y$ in $r_1 -^e r_2$. This means that for all i, $1 \le i \le n$, $\exists j$, $1 \le j \le k$, such that $t_i[X_1 \ldots X_m] \in u_j[Y]$, and $\bigcup_{j=1}^{k} u_j[Y] = \{t_i[X_1 \ldots X_m] \mid 1 \le i \le n\}$. Each u_j is created by the extended difference of two tuples, $u_j{}^1 \in r_1$ and $u_j{}^2 \in r_2$. Since Y is a purely hierarchical relation, $\bigcup_{j=1}^{k} u_j[Y] \in (\bigcup_{j=1}^{k} u_j{}^1[Y] -^e \bigcup_{j=1}^{k} u_j{}^2[Y])$. When the tuples $u_j{}^1$ and $u_j{}^2$ are unnested into tuples $v_l{}^1$, $(1 \le i \le p_1)$ and $v_l{}^2$, $(1 \le l \le p_2)$, we have $\bigcup_{j=1}^{k} u_j{}^1[Y] = \{v_l{}^1[X_1 \ldots X_m] \mid 1 \le l \le p_1\}$ and $\bigcup_{j=1}^{k} u_j{}^2[Y] = \{v_l{}^2[X_1 \ldots X_m] \mid 1 \le l \le p_2\}$, and we can say $\{t_i[X_1 \ldots X_m] \mid 1 \le i \le n\} \subseteq \{v_l{}^1[X_1 \ldots X_m] \mid 1 \le l \le p_1\} - \{v_l{}^2[X_1 \ldots X_m] \mid 1 \le l \le p_2\}$. Therefore $\mu_Y(r_1) - \mu_Y(r_2)$ contains all the tuples $t_1, \ldots, t_n$ in $\mu_Y(r_1 -^e r_2)$.

Induction Step: By the induction hypothesis, we know that

$$\begin{aligned}&\mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))\\&\qquad = \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1))\ldots)) -^e \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_2))\ldots))\end{aligned}$$

for the first $(n-1)$ nesting levels, where Y_i is the higher-order attribute at the i^{th} nesting level, $1 \le i \le n-1$. We now show that this is also true for n nesting levels. If we unnest both sides of the last equation with Y_n, we obtain

$$\begin{aligned}&\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))\\&\qquad = \mu_{Y_n}[\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots) -^e \mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)]\end{aligned}$$

Let $r_1' = \mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)$ and $r_2' = \mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)$,

now we have

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots)) = \mu_{Y_n}(r_1' -^e r_2').$$

Since r_1' and r_2' are relations whose nesting-depths are 1, $\mu_{Y_n}(r_1' -^e r_2') = \mu_{Y_n}(r_1') - \mu_{Y_n}(r_2')$, which is proved to be true in the basis step. If we substitute r_1' and r_2' by their equivalents, we have

$$\begin{aligned}&\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))\\&\qquad = \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) - \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots))\end{aligned}$$

(3) In this case we show that

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))$$
$$= \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) - \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots))$$

The proof is by induction on the number of the higher-order attributes at the first and only nesting level.

Basis: We show that $\mu_{Y_1}(\mu_{Y_2}(r_1 -^e r_2)) = \mu_{Y_1}(\mu_{Y_2}(r_1)) - \mu_{Y_1}(\mu_{Y_2}(r_2))$, where $|HA| = 2$ and $Y_1 = X_1 \ldots X_m$, $Y_2 = X_l \ldots X_k$.

$\supseteq$ *part:* We show that if $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) - \mu_{Y_1}(\mu_{Y_2}(r_2))$, then $t \in \mu_{Y_1}(\mu_{Y_2}(r_1 -^e r_2))$. Since $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) - \mu_{Y_1}(\mu_{Y_2}(r_2))$, we know that t is only in $\mu_{Y_1}(\mu_{Y_2}(r_1))$ and it is unnested from some $u_1 \in r_1$. Then we can say that $(t[X_1 \ldots X_m] \in u_1[Y_1] \wedge t[X_l \ldots X_k] \in u_1[Y_2]) \wedge \forall u_2 \in r_2 : t \notin u_2)$. In the *extended difference* of u_1 and u_2, u_1 will be included either completely as u_1, or partially as a new tuple u. Since $\forall u_2 \in r_2, t \notin u_2, t \in u_1$ or $t \in u$. Therefore $t \in \mu_{Y_1}(\mu_{Y_2}(r_1 -^e r_2))$.

$\subseteq$ *part:* We show that if $t \in \mu_{Y_1}(\mu_{Y_2}(r_1 -^e r_2))$, then $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) - \mu_{Y_1}(\mu_{Y_2}(r_2))$. In this case, t is unnested from some u in $r_1 -^e r_2$. u satisfies one of the disjuncts in the $-^e$ definition and all the disjuncts in this definition include those tuples only in r_1, so

$(\forall u' \in \mu_{Y_1}(\mu_{Y_2}(u)) : u' \in \mu_{Y_1}(\mu_{Y_2}(r_1)) \wedge (\forall t' \in \mu_{Y_1}(\mu_{Y_2}(r_2)) : u' \neq t'))$.

The last statement is the definition of the standard set difference, therefore $t \in \mu_{Y_1}(\mu_{Y_2}(r_1)) - \mu_{Y_1}(\mu_{Y_2}(r_2))$.

Induction Step: By the induction hypothesis, we know that

$$\mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))$$
$$= \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_1))\ldots)) -^e \mu_{Y_{n-1}}(\mu_{Y_{n-2}}(\ldots(\mu_{Y_1}(r_2))\ldots)),$$

for the first $(n-1)$ higher-order attributes of E_R, where $n \geq 3$. Now we show that this is also true for n:

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))$$
$$= \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) -^e \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)).$$

The proof is similar to the proof of induction step of case (2). If $\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}$ $(r_1 -^e r_2))\ldots)$ is unnested with Y_n and r_1' and r_2' are substituted as in case (2), we obtain,

$$\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots)) = \mu_{Y_n}(r_1' -^e r_2').$$

Since r_1' and r_2' are relations which have one higher-order attribute and one nesting level, $\mu_{Y_n}(r_1' -^e r_2') = \mu_{Y_n}(r_1') - \mu_{Y_n}(r_2')$, which is proved to be true in the basis of case (2). Therefore

$$\begin{aligned}&\mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1 -^e r_2))\ldots))\\ &\quad = \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_1))\ldots)) - \mu_{Y_n}(\mu_{Y_{n-1}}(\ldots(\mu_{Y_1}(r_2))\ldots)).\end{aligned}$$

(4) This is the most general case of a nested relation, that is a nested relation with n higher-order attributes, each of which is also a nested relation with a finite number of higher-order attributes and nesting levels.

We show that the *extended difference* operator is information equivalent to this kind of relation structures in several steps. In these steps, using a recursive procedure, we obtain the most general nested structure and show that the *extended difference* operator is information equivalent to this structure.

Now let the relation structures of r_1 and r_2 have $n \in N^+$ higher-order attributes, where each has a relation structure which is equal to that of (1), (2), or (3) and let this new structure be (4.a). To show that *extended difference* is information equivalent in this case, we show that

$$\begin{aligned}&\mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots))\\ &= \mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots)) - \mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))\end{aligned}$$

where S_{Y_i} is the unnest sequence (a set of higher-order names in E_{Y_i}) required to flatten the i^{th} higher-order attribute in E_R. The proof is by induction on the number of higher-order attributes in E_R.

Basis: In this case, $|HA| = 1$ and there is only one higher-order attribute in E_R. The structure of this higher-order attribute is equal to that of (1), (2), or (3). Since we have shown that the *extended difference* operator is information equivalent to the structures of (1), (2), and (3), we conclude that

$$\mu_Y(\mu_{S_Y}(r_1 -^e r_2)) = \mu_Y(\mu_{S_Y}(r_1)) - \mu_Y(\mu_{S_Y}(r_2))$$

Induction Step: By the induction hypothesis we know that

$$\begin{aligned}&\mu_{(Y_{n-1},Y_{n-2},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots))\\ &= \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots)) -^e \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))\end{aligned}$$

for the first $(n-1)$ higher-order attributes of E_R. We now show that this is also true for all the higher-order attributes of E_R, which is stated as follows

$$\begin{aligned}&\mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots))\\ &= \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots)) -^e \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))\end{aligned}$$

If we nest both sides of the equality introduced by the induction hypothesis with Y_n and S_{Y_n}, we obtain

$$\begin{aligned}&\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots)))) = \mu_{Y_n}(\mu_{S_{Y_n}}\\ &[\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots)) -^e \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))])\end{aligned}$$

Let $r_1' = \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))$ and
$r_2' = \mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))$

If we replace $\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1))\ldots))$ and $\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_2))\ldots))$ with r_1' and r_2' respectively, we obtain

$$\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\ldots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots)))) = \mu_{Y_n}(\mu_{S_{Y_n}}(r_1' -^e r_2')$$

The structure of r_1' and r_2' contains one higher-order attribute which is in one of the forms (1), (2), or (3). Since it is shown in the basis step that *extended difference* is information equivalent to the structures of (1), (2), and (3), we conclude that

$$\mu_{Y_n}(\mu_{S_{Y_n}}(r_1{}' -^e r_2{}')= \mu_{Y_n}(\mu_{S_{Y_n}}(r_1{}')) - \mu_{Y_n}(\mu_{S_{Y_n}}(r_2{}')).$$

With the introduction of this equation, we obtain the following equality

$$\begin{aligned}&\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots))))\\&= \mu_{Y_n}(\mu_{S_{Y_n}}(r_1{}')) - \mu_{Y_n}(\mu_{S_{Y_n}}(r_2{}'))\end{aligned}$$

If $r_1{}'$ and $r_2{}'$ are substituted with their equivalents, we have

$$\begin{aligned}&\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots))))\\&= \mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_1))\ldots)))) -\\&\mu_{Y_n}(\mu_{S_{Y_n}}(\mu_{(Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_{n-1}}}(\cdots(\mu_{S_{Y_1}}(r_2))\ldots))))\end{aligned}$$

With respect to **Theorem 8.1.b** of [6], the order of unnest is not important, so we can reorganize the previous equality by changing the unnest sequence and obtain the following equality:

$$\begin{aligned}&\mu_{(Y_n,Y_{n-1},\ldots,Y_1)}(\mu_{S_{Y_n}}(\cdots(\mu_{S_{Y_1}}(r_1 -^e r_2))\ldots))\\&= \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\cdots(\mu_{S_{Y_1}}(r_1))\ldots)) - \mu_{(Y_n,\ldots,Y_1)}(\mu_{S_{Y_n}}(\cdots(\mu_{S_{Y_1}}(r_2))\ldots))\ \square\end{aligned}$$

3.3.3 Extended Intersection

Extended intersection is another set operator that does not need restructuring. As with *union* and *difference*, to be able to find the intersection of two structures $\mathcal{R}_1 = < R_1, r_1 >$ and $\mathcal{R}_2 = < R_2, r_2 >$, their schemes R_1 and R_2 must be equal. The structure of the resultant relation is $< R_3, r_1 -^e r_2 >$, where R_3 is equal to R_1 and R_2. The *extended intersection* is defined as follows [6]:

$$\begin{aligned}r_1 \cap^e r_2 = \{t \mid (&\exists t_1 \in r_1,\ \exists t_2 \in r_2:\\&(\forall X, Y \in E_{R_1}:\ t[X] = t_1[X] = t_2[X]\\&\wedge\ t[Y] = (t_1[Y] \cap^e t_2[Y]) \wedge t[Y] \neq \emptyset))\}\end{aligned}$$

The *extended intersection* in [2] is defined as follows:

Let *f* be a format and *I*, *J* two instances over *f*. Then the *intersection* of *I*

$r_1 \cap^e r_2$

A	X		Y	
	B	K	C	D
a_1	b_1	k_1	c_1	d_1

$\mu_X(\mu_Y(r_1 \cap^e r_2))$

A	B	K	C	D
a_2	b_1	k_1	c_1	d_1

Figure 3.19: Extended intersection of r_1 and r_2

$\mu_X(\mu_Y(r_1))$

A	B	K	C	D
a_1	b_1	k_1	c_1	d_1
a_1	b_2	k_2	c_1	d_1
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_2	d_2
a_2	b_7	k_7	c_1	d_1
a_2	b_7	k_7	c_2	d_2

$\mu_X(\mu_Y(r_2))$

A	B	K	C	D
a_2	b_1	k_1	c_1	d_1
a_2	b_1	k_1	c_3	d_3
a_2	b_8	k_8	c_1	d_1
a_2	b_8	k_8	c_3	d_3
a_4	b_4	k_4	c_4	d_4

$\mu_X(\mu_Y(r_1)) \cap \mu_X(\mu_Y(r_2))$

A	B	K	C	D
a_2	b_1	k_1	c_1	d_1

Figure 3.20: The *desired-result*

and J is the instance over f, denoted $I \circledcap J$, defined by:

1. if $f \equiv X$, where X is nonempty, then $I \circledcap J = I \cap J$, and
2. if $f \equiv X(f_1)^* \ldots (f_n)^*$, where $f_1, \ldots, f_n$ are nonempty, then:

$$I \circledcap J = \left\{ < u(I_1 \circledcap J_1) \ldots (I_n \circledcap J_n) > \;\middle|\; \begin{array}{l} < uI_1 \ldots I_n > \in I \text{ and} \\ < uJ_2 \ldots J_n > \in J \end{array} \right\}$$

Both of these *extended intersection* operators are information equivalent. Since we use the model of RKS, we use their *extended intersection* as well.

Example: If the *extended intersection* operator of [6] is applied to the relations r_1 and r_2 in Figure 3.8, we obtain the result $r_1 \cap^e r_2$ in Figure 3.19. The flat form of $r_1 \cap^e r_2$ is also shown in the same figure. This flattened result is equal to the *desired-result* depicted in Figure 3.20. □

The class of PNF relations is closed under *extended intersection* which is stated in Theorem 6.1 of [6]. What this theorem states is that the structure $\mathcal{R}_3 = < R, r_1 \cap^e r_2 >$ is in PNF, given that the structures $\mathcal{R}_1 = < R, r_1 >$ and $\mathcal{R}_2 =$

$< R, r_2 >$ are in PNF.

Theorem 3.3 *The extended intersection operator is information equivalent,* that is

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = \mu_{Y_1}(\ldots(\mu_{Y_n}(r_1))\ldots) \cap \mu_{Y_1}(\ldots(\mu_{Y_n}(r_2))\ldots),$$

where $Y_1 \ldots Y_n$ is the unnest sequence (the set of higher-order attributes in the relation structure) required to flatten the relations r_1, r_2, and $r_1 \cap^e r_2$.

Proof In this proof we use **Theorem 8.2.a** of RKS. This theorem is stated as follows in [6].

Given two relation structures $\mathcal{R}$ and $\mathcal{S}$, the following property holds

$$\mu_A(\mathcal{R} \cap^e \mathcal{S}) = \mu_A(\mathcal{R}) \cap^e \mu_A(\mathcal{S}).$$

(A is an higher-order attribute in E_R, $\mathcal{R} = < R, r >$, and $\mathcal{S} = < S, s >$.)

Let us flatten $r_1 \cap^e r_2$ by unnesting it with the sequence $Y_1 \ldots Y_n$. We know that $\mu_{Y_n}(r_1 \cap^e r_2) = \mu_{Y_n}(r_1) \cap^e \mu_{Y_n}(r_2)$ (by **Theorem 8.2.a** [6]), so we have

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = \mu_{Y_1}(\ldots(\mu_{Y_{n-1}}[\mu_{Y_n}(r_1) \cap^e \mu_{Y_n}(r_2)])\ldots)$$

If we let ${r_1}^1 = \mu_{Y_n}(r_1)$ and ${r_2}^1 = \mu_{Y_n}(r_2)$, and replace $\mu_{Y_n}(r_1)$ and $\mu_{Y_n}(r_2)$ with ${r_1}^1$ and ${r_2}^1$ respectively, we obtain

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = \mu_{Y_1}(\ldots(\mu_{Y_{n-1}}({r_1}^1 \cap^e {r_2}^1))\ldots)$$

The class of PNF relations is closed under *unnesting* (**Theorem 5.1** [6]), and it is given that r_1 and r_2 are in PNF, so ${r_1}^1$ and ${r_2}^1$ are also in PNF, and we can apply *extended intersection* to ${r_1}^1$ and ${r_2}^1$. By **Theorem 8.2.a** [6], we know that $\mu_{Y_{n-1}}({r_1}^1 \cap^e {r_2}^1) = \mu_{Y_{n-1}}({r_1}^1) \cap^e \mu_{Y_{n-1}}({r_2}^1)$, so we have

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = \mu_{Y_1}(\ldots(\mu_{Y_{n-2}}[\mu_{Y_n}({r_1}^1) \cap^e \mu_{Y_n}({r_2}^1)])\ldots)$$

If we let ${r_1}^2 = \mu_{Y_{n-1}}({r_1}^1)$ and ${r_2}^2 = \mu_{Y_{n-1}}({r_2}^1)$, and replace $\mu_{Y_{n-1}}({r_1}^1)$ and $\mu_{Y_{n-1}}({r_2}^1)$ with ${r_1}^2$ and ${r_2}^2$ respectively, we obtain

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = \mu_{Y_1}(\ldots(\mu_{Y_{n-2}}({r_1}^2 \cap^e {r_2}^2))\ldots)$$

${r_1}^2$ and ${r_2}^2$ are in PNF and *extended intersection* can be applied to them because of the same reasons explained in the previous steps.
If we keep on applying the same procedure until the relation structures contain no more higher-order attributes (i.e., the relation structures are flat), we finally obtain

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = {r_1}^n \cap^e {r_2}^n, \text{ where}$$

$$\begin{array}{lclclcl}
{r_1}^n & = & \mu_{Y_1}({r_1}^{n-1}) & \text{and} & {r_2}^n & = & \mu_{Y_1}({r_2}^{n-1}) \\
{r_1}^{n-1} & = & \mu_{Y_2}({r_1}^{n-2}) & \text{and} & {r_2}^{n-1} & = & \mu_{Y_2}({r_2}^{n-2}) \\
\vdots & & & \vdots & & \vdots & \\
{r_1}^1 & = & \mu_{Y_2}(r_1) & \text{and} & {r_2}^1 & = & \mu_{Y_2}(r_2)
\end{array}$$

Using the above equations, we find that

$${r_1}^n = \mu_{Y_1}(\ldots(\mu_{Y_n}(r_1))\ldots) \text{ and } {r_2}^n = \mu_{Y_1}(\ldots(\mu_{Y_n}(r_2))\ldots).$$

Since ${r_1}^n$ and ${r_2}^n$ are flat relations, we have ${r_1}^n \cap^e {r_2}^n = {r_1}^n \cap {r_2}^n$ (which obviously follows from the *extended intersection* definition). By replacing ${r_1}^n$ and ${r_2}^n$ with their equivalents, we finally obtain the following equality, which is what we are trying to show

$$\mu_{Y_1}(\ldots(\mu_{Y_n}(r_1 \cap^e r_2))\ldots) = \mu_{Y_1}(\ldots(\mu_{Y_n}(r_1))\ldots) \cap \mu_{Y_1}(\ldots(\mu_{Y_n}(r_2))\ldots) \ \Box$$

Chapter 4

Conclusions

In this study, we presented the database models of RKS [6] and AB [2] to formalize $\neg 1NF$ relations with their extended relational algebra. In these models the notions of database and relation structures, database and relation schema, instance, domain, and attribute are extended for $\neg 1NF$ relations.

Extended relational algebra operators are defined recursively both in RKS and AB. We have restricted ourselves to only extended set operators *union, difference*, and *intersection*. We have introduced the notion of *information equivalent* set operator, which generates a result that is equal to the desired-result when it is flattened. (Hence, an information equivalent set operator does not lose any tuples in the desired-result or does not introduce extra tuples that are not in the desired-result.) We have shown that the extended set operators *union* and *difference* of RKS and AB are not information equivalent.

The extension we have introduced was the new extended *union* and *difference* operators which were shown to be information equivalent. The model of RKS is used in these definitions. Furthermore, we have proved that the *extended intersection* operator of RKS is information equivalent.

We did not consider all the extended relational algebra operators in this study. Further research may be carried out to define other extended relational algebra operators such as *selection, join*, etc.

References

[1] S. ABITEBOUL, C. BEERI, M. GYSSENS, and D. V. GUCHT. An introduction to the completeness of languages for complex objects and nested relations. In *Volume 361 of Lecture Notes in Computer Science*, pages 117–138, Berlin, 1989. Springer-Verlag.

[2] S. ABITEBOUL and N. BIDOIT. Non first normal form relations: An algebra allowing data restructuring. *Journal of Computer and System Sciences*, 33(4):361–393, 1986.

[3] L. S. COLBY. A recursive algebra and query optimization for nested relations. *ACM SIGMOD Record*, 18(2):273–283, June 1989.

[4] L. GARNETT and A. U. TANSEL. Equivalence of relational algebra and calculus languages for nested relations. Technical report, Baruch College, CUNY (City University of NewYork), May 1988.

[5] A. MAKINOUCHI. A consideration on normal form of not-necessarily-normalized relations in the relational data model. In *Proceedings of the Third International Conference on Very Large Data Bases*, pages 447–453, Tokyo, October 1977.

[6] M. A. ROTH, H. F. KORTH, and A. SILBERSCHATZ. Extended algebra and calculus for nested relational databases. *ACM Transactions on Database Systems*, 13(4):389–417, December 1988.

[7] H. J. SCHEK and M. H. SCHOLL. The relational model with relation-valued attributes. *Information Systems*, 11(2):137–147, 1986.

[8] J. D. ULLMAN. *Principles of Database and Knowledge-Base Systems.* Computer Science Press, Rockville, MD, 1988.