Algebraic semantics of ER-models in the context of the calculus of relations. II: Dynamic view.

Ernst-Erich Doberkat
Department of Computer Science
Universität Dortmund
Dortmund, Germany

Eugenio G. Omodeo
Dipartimento Di Matematica Pura ed Applicata
Università degli Studi di L'Aquila
L'Aquila, Italy

30th July 2001

Contents

Abstract

We provide a detailed analysis of the insertion and deletion operations for an ER-model represented in terms of the map calculus. This continues our previous study of compiling an ER model into the abstract setting of what might be called logic without variables.

1 Introduction

Entity relationship modelling (*ER modelling* for short) is a widespread and powerful technique for data modelling. An ER model captures all the relationships between data using entities and relations together with attributes on them. The very popular modelling approach through the Unified Modelling Language (*UML*, see [12]) is partially based on it. A formal semantics of ER modelling, however, is not easy to come by: as usual, a popular technique is described more or less informally, and this is notoriously difficult to model formally.

There are several approaches at formally describing the semantics of this modelling technique. They are mainly based on algebraic modelling techniques and capitalize on the semantic framework that comes with them. Hettler [7] gives a translation of these models into the specification language SPECTRUM, essentially modelling entities as records with attributes as entries, but not taking inheritance into account. The formal semantics of an extended ER-model is investigated in [6, 8] from a database point of view, proposing the semantics of a database signature as the set of all interpretations; this work does not mention algebraic specifications explicitly. In [5] it is shown how to generate an algebraic specification from an ER-model, hereby carrying the model based semantics of such a specification over to ER-models.

The present paper proposes formalizing ER modelling through *relational algebra* (which is different from *relation algebra* so useful in data base programming languages!), a branch of Logic brought to flourish through the work of Ernst Schröder (see the historical introduction in [1]). Relational algebra has been used for decomposing relations in a database according to functional dependencies in [9], these methods have not yet be utilized for a systematic investigation of the dynamic behavior of a data base.

We separate the static structure (the topology) of the ER model from its dynamic counterpart, and we have shown already how to model the static view using relational algebra in a companion paper [11]. This is obviously not enough, because the dynamic nature of an ER model cannot be described using the static structure alone. Let us have a look at abstract data types for just conveying the flavor of our arguments.

1.1 The ADT view

An abstract data type (ADT) encapsulates data and the operations (usually called *methods*) on it. This notion of an ADT is fundamental in object oriented software construction, classes may be considered as special cases of ADTs. This notion is fundamental because it supports data abstraction and permits keeping data and their operation in one physically well defined place. ADTs serve as templates, they are instantiated, and the instances of an ADT are the living capsules data and operations are kept in. The state of an instance is just the collection of specific values the data of this instance are having. The approach *Design by Contract*, so forcefully advocated by Bertrand Meyer [10], and realized in his language Eiffel, goes one step beyond, associating with each ADT specific properties called *invariants*. Operations on an (instance of an) ADT have to respect these invariants in the sense that each operation that starts on an instance which satisfies the invariant leaves the instance in a state which also satisfies it. Each method m of an ADT is associated with a precondition pre_m and a postcondition post_m indicating a contract: entering m such that pre_m is satisfied guarantees leaving m with post_m satisfied. In Hoare's notation of predicate transforms,

$$\{\mathsf{inv} \wedge \mathsf{pre}_m\}\ m\ \{\mathsf{inv} \wedge \mathsf{post}_m\}.$$

Actually, Design by Contract entails more, because inheritance comes into the game through rather involved co- and contravariant rules relating methods from subclasses to superclasses, but this will not concern us here.
Call an ADT an *Eiffel ADT* iff it has invariants and pre- as well as postconditions, and if the Design by Contract rules are imposed on its methods.
An ER model $\mathcal{M}$ may be considered as an ADT. The data to be stored in an instance are composed of the data stored in the entities, relations and attributes, and the invariant is provided by the conditions imposed on the model's validity (see Definition 5.4). We should look for three families of operations:

- initializing an instance of $\mathcal{M}$,
- inserting elements into entities and relations,
- deleting elements from entities and relations.

Note that we do not talk about operations but rather about families of them; this is so since an operation like inserting an element into a relation R usually entails other operations (like inserting elements into the domain, and into the codomain of R); there may be more subtle dependencies as well, as we will see.
The invariant to be maintained by these operations is the validity of the model; this means that the model before and after one of these families of operations has to conform to the model's declaration. The postconditions are in every case empty, because the operations are all geared towards maintaining the ADT's invariant. The preconditions are sketched as follows:

Initialization: Since this operation initializes every entity and every relation to the empty set, no precondition needs to be provided. The assumption is that we always start from an empty model, so we do not cater for this operation.

Insertion: The insertion of elements into an entity or a relation requires a set of conditions which will force the invariant to hold after the insertions took place. This will provide the precondition, see Proposition 5.2.

Deletion: Similarly, the deletion of elements requires a set of conditions which will help maintaining the invariant. The conditions imposed there form the precondition, they are formulated in Proposition 6.4.

Hence $\mathcal{M}$ forms in fact an Eiffel ADT.

1.2 Overview

What needs to be done then is to formulate the invariant and the precondition using the language we have chosen for our formalization. After we discuss the version of ER modelling we want to work with in Section 2, we introduce relational algebra (or *map algebra*, as we will call it usually) briefly in Section 3, there we will also provide some abbreviations that are helpful for the discussions to follow. Section 4 formulates essential pieces of an ER model in map algebra, borrowing freely from [11]. Section 5 deals with a formulation of the preconditions for insertions; for reasons of reducing the complexity, this is split into the bare bones version of an ER model which does not entertain attributes. This leads to the notion of a *weakly valid* ER model, and it is shown under which conditions weak validity is maintained. Attributes are added to the discussions at that point, leading to the notion of a valid model, and strengthening the preconditions towards keeping validity invariant. A very similar procedere is observed when discussing deletions in Section 6, which quite surprisingly turns out to be easier to handle than insertions. This is mainly due to the fact that most of the interesting properties are downward closed: if a map expression W observes it, then all map expressions $V \subseteq W$ do, too. Section 7 proposes further investigations along the lines suggested here, discussing for example how model checking as a technique to ascertain properties of an ER model could be incorporated.

Acknowledgements. This work was done while the first author was visiting the Computer Science Group of the Dipartimento di Matematica Pura ed Applicata at the University of L'Aquila. This research was in part supported through grants from the *Exchange Programme for Scientists between Italy and Germany* from the Italian Minstry of Foreign Affairs/*Deutscher Akademischer Austauschdienst* and from *Progetto speciale I.N.D.A.M./GNIM "Nuovi paradigmi di calcolo: Linguaggi e Modelli"*.

2 Entity Relationship Models

Entity Relationship modelling [4] is a popular and widespread technique for data modelling which we assume the reader to be familiar with. Many variants have been discussed (Thalheim's encyclopedic book [15] provides an overview).

2.1 The variant to be considered

We will restrict ourselves to a rather basic variant in which

- All relations are binary, and the only cardinality restriction that may be imposed on a relation is that it is left- or right- unique.
- Inheritance is restricted to single inheritance.
- Relations are assumed to be total. In fact, in the presence of inheritance non-total relations may be transformed into total ones by introducing additional entities for the domain, and for the range, respectively,
- Attributes are defined on entities only.

This is the version of ER modelling investigated in [11] and a bit more restrictive than the one investigated in [5]. These restrictions can be removed or refined at the cost of a more complicated technical development. We feel, however, that the methods we develop here provide a way of modelling these more complicated situations.

2.2 The process model

We are given an instance $\mathcal{M}$ of an ER model which is *valid*, so all constraints formulated in the declaration of the model are satisfied. We want to investigate *change*, namely we want to investigate under which conditions insertions and deletions into $\mathcal{M}$ lead to a valid model again. In order to investigate this for insertions, we assume that we have complete information

about the items to be inserted. Thus, if E is an entity, we know the items δ^+E to be inserted into E, yielding $E \cup \delta^+E$ as the new version of this entity. Similarly, we know for relations R the tuples δ^+R to be inserted, and we know for attributes α the changes in $\delta^+\alpha$. What we want to know is, under which conditions for $E, \delta^+E, R, \delta^+R$ and $\alpha, \delta^+\alpha$ the invariance of validity of the instance is maintained. The question arises *mutatis mutandis* for deletions.
Note that the assumption that the change sets δ^+ and δ^- are given does not address the problem of constructing them. When insertion is done interactively, and is not done with care, situations may arise when an infinite sequence of insertions may be necessary; this can be demonstrated through easily found examples. We bypass these complications by postulating that complete information is available from the outset.

3 Map Calculus

ER models will be formulated in terms of relational algebras. These algebras formalize axiomatically the usual operations on binary relations (like composition or inversion), so that binary relations appear as one of several models that are possible for these algebras. We will provide a very brief introduction to these algebras, and we will fix some notations for the reader's convenience.
A relational algebra (or *map algebra*) is defined as a Boolean algebra with additional properties that are imposed because a composition relation is available. The version of relational algebras we want to use is defined below; for variants and further developments the reader is encouraged to consult [2, Ch. 2] or [1, Ch. 1].

Definition 1 $\langle \emptyset, \mathbb{1}, \cap, \overline{\cdot}, \iota, \circ, \cdot^{-1} \rangle$ *is called a* relational algebra *iff*

1. $\langle \emptyset, \mathbb{1}, \cap, \overline{\cdot} \rangle$ *is a Boolean algebra with smallest element* $\emptyset$*, largest element* $\mathbb{1}$*, intersection (meet)* $\cap$*, and complementation* $\overline{\cdot}$*; the associated order relation and union (join) are denoted by* $\subseteq$*, and* $\cup$*, resp.*

2. $\circ$ *is a binary associative operation on the Boolean algebra with* ι *as the left- and right-neutral element,*

3. $\cdot^{-1}$ *is a unary idempotent operation on the Boolean algebra,*

4. *the following properties hold:*

   (a) $(P \circ Q)^{-1} = Q^{-1} \circ P^{-1}$,
   (b) $(P \cap Q)^{-1} = P^{-1} \cap Q^{-1}$,
   (c) $P \circ (Q_1 \cap Q_2) \subseteq P \circ Q_1 \cap P \circ Q_2$ *($\cap$-subdistributivity),*
   (d) $P \circ (Q_1 \cup Q_2) = P \circ Q_1 \cup P \circ Q_2$ *($\cup$-distributivity),*

5. $P \subseteq Q$ *implies* $P \circ R \subseteq Q \circ R$*,*

6. $(P \circ Q) \cap R = \emptyset$ *implies* $(P^{-1} \circ R) \cap Q = \emptyset$ *(Schröder's Rule).*

Map algebra consists of map equalities $P = Q$, where P and Q are map expressions:

Definition 2 *Map expressions are terms of the signature according to the table below, where we have added union* $\cup$ *as an associative operation, and the left-associative set difference* $\setminus$ *for convenience:*

Symbol	$\emptyset$	$\mathbb{1}$	ι	r_i	$\cap$	$\circ$	$\cdot^{-1}$	$\overline{\cdot}$	$\cup$	$\setminus$
Degree	0	0	0	0	2	2	1	1	2	2
Priority					5	6	7	2	2	2

Here r_i *is one of the countably many* map letters *which we assume to be available.*

Map letters are used to customizing map algebra by attaching additional properties through additional axioms for the relational algebra, as we will see in the sequel.
An *interpretation* $\mathcal{I}$ over a universe $\mathcal{U}$ maps each map expression to a subset of the Cartesian square $\mathcal{U}^2 \equiv_{\mathrm{Def}} \mathcal{U} \times \mathcal{U}$ such that e.g.

$$\begin{array}{rcl|rcl} \emptyset^{\mathcal{I}} & = & \emptyset & (P \cap Q)^{\mathcal{I}} & = & P^{\mathcal{I}} \cap Q^{\mathcal{I}}, \\ \mathbb{1}^{\mathcal{I}} & = & \mathcal{U}^2 & (P \circ Q)^{\mathcal{I}} & = & P^{\mathcal{I}} \circ Q^{\mathcal{I}} \\ \iota^{\mathcal{I}} & = & \Delta & \left(Q^{-1}\right)^{\mathcal{I}} & = & \left(Q^{\mathcal{I}}\right)^{-1} \end{array}$$

Here Δ is the diagonal $\{\langle a, a\rangle \mid a \in \mathcal{U}\}$ of $\mathcal{U}$, and the operations on the right-hand side are the familiar ones manipulating relations over sets. Hence e.g. $\cup$-distributivity translates into the set equality $R \circ (S_1 \cup S_2) = R \circ S_1 \cup R \circ S_2$. that is familiar for the relations $R \subseteq A \times B$ and $S_1, S_2 \subseteq B \times C$ for sets A, B and C.
Adding new axioms through fixing properties of map letters has the effect of restricting interpretations: they have to satisfy the additional properties for the interpretation of the map letters, which in turn also have to be provided.
For convenience, we use some abbreviations which are listed in the table below.

Notation	Expression	Note
$\mathsf{Coll}(R)$	$R \subseteq \iota$	
$\mathsf{Total}(R)$	$R \circ \mathbb{1} = \mathbb{1}$	
$\mathsf{dom}(R)$	$R \circ \mathbb{1} \cap \iota$	domain of R
$\mathsf{img}(R)$	$\mathbb{1} \circ R \cap \iota$	range/image of R
$\mathsf{LUniq}(R)$	$\mathsf{Coll}\left(R^{-1} \circ R\right)$	$R^{\mathcal{I}}$ is a partial map
$\mathsf{RUniq}(R)$	$\mathsf{LUniq}\left(R^{-1}\right)$	$\left(R^{-1}\right)^{\mathcal{I}}$ is a partial map
$\mathsf{NonVoid}(R)$	$\mathsf{Total}(\mathbb{1} \circ R)$	$R^{\mathcal{I}} \neq \emptyset$
$\mathsf{Snglt}(R)$	$\mathsf{NonVoid}(R)$ & $\mathsf{LUniq}(\mathbb{1} \circ R)$ & $\mathsf{LUniq}\left(R^{-1}\right)$	$R^{\mathcal{I}}$ is a singleton
$\mathsf{DomSub}(R, S)$	$R \circ \mathbb{1} \subseteq S \circ \mathbb{1}$	domain containment
$\mathsf{ImgSub}(R, S)$	$\mathbb{1} \circ R \subseteq \mathbb{1} \circ S$	range containment

For example, $\mathsf{Coll}(E)$ says that $E^{\mathcal{I}}$ is supposed to consist of pairs of the form $\langle a, a\rangle$, $\mathsf{Total}(E)$ indicates that $E^{\mathcal{I}}$ is (left-) total, hence that for each $a \in \mathcal{U}$ there is some $b \in \mathcal{U}$ with $\langle a, b\rangle \in E^{\mathcal{I}}$. The reader is invited to formulate these expressions in terms of set-theoretic relations.
Some identities and inequalities will be particularly helpful in the sequel; we collect them here for convenience, and refer the reader to [14] and to [13].

Lemma 1 *Let* P, Q *and* R *be map expressions, then*

1. $\mathsf{LUniq}(R) \Leftrightarrow R \circ (P \cap Q) = R \circ P \cap R \circ Q$,

2. $(P \cup Q)^{-1} = P^{-1} \cup Q^{-1}$,
3. $(P \circ Q) \setminus (R \circ Q) \subseteq (P \setminus R) \circ Q$
4. $P, Q \subseteq \iota$, *then*
   (a) $P^{-1} \circ Q = \emptyset$, *provided* $P \cap Q = \emptyset$,
   (b) $(P \setminus Q) \circ \mathbb{1} = (P \circ \mathbb{1}) \setminus (Q \circ \mathbb{1})$,
5. $Q \circ R \subseteq S \Leftrightarrow Q^{-1} \circ \overline{S} \subseteq \overline{R} \Leftrightarrow \overline{S} \circ R^{-1} \subseteq \overline{Q}$ *(Schröder's Cycle Rules),*
6. $\overline{P^{-1}} = \left(\overline{P}\right)^{-1}$. ■

Map Letters We assume that we have a countably infinite provision of map letters $\mathsf{r}_1, \mathsf{r}_2, \ldots$ at our disposal of which we reserve the first T for system purposes.

4 Preparations

Now let an ER model $\mathcal{M}$ be given. All information concerning $\mathcal{M}$ can be found in a declaration which represents the static information about the model, and which permits stating the validity of an instantiation for $\mathcal{M}$. For the time being we concentrate on entities and relations, attributes will be added later on.

4.1 The Basic Model

If E is the domain of relation R with F as its co-domain, then we will assume that E and F are tight, i.e., that for each entity e there exists f such that $\langle e, f\rangle$ is in R, similarly for F. This is indicated by $E \bullet\!\!- R -\!\!\bullet F$. In what follows, entities and relations will be considered an element of a fixed (but anonymous) map algebra. An entity E is then represented through $\mathsf{Coll}(E)$, hence consists of pairs the first and the second component of which agree, and

$$E \bullet\!\!- R -\!\!\bullet F$$

translates to

$$\mathsf{DotDot}(E, R, F) \equiv_{\text{Def}} \mathsf{Coll}(E) \;\&\; \mathsf{Coll}(F) \;\&\; \mathsf{DomSub}(E, R) \;\&\; \mathsf{ImgSub}(F, R).$$

Either relation $\bullet\!\!-$ or $-\!\!\bullet$ may be tightened to $\bullet\!\!\overset{1}{-}$ and $\overset{1}{-}\!\!\bullet$, resp., indicating uniqueness. Thus $E \bullet\!\!\overset{1}{-} R$ means in additional to $E \bullet\!\!- R$ that

$$\langle x, y\rangle \in R \wedge \langle x', y\rangle \in R \Rightarrow x = x'$$

holds, which may be translated conveniently into $\mathsf{LUniq}(R)$. Similarly, $R \overset{1}{-}\!\!\bullet F$, which means

$$\langle x, y\rangle \in R \wedge \langle x, y'\rangle \in R \Rightarrow y = y'$$

is translated into $\mathsf{RUniq}(R)$.
Note that either of these conditions depends only on the relation, not on the domain or the codomain.

The different way a relation relates to its domain and its codomain may be captured through the suitable combination of macros which are comprehensively listed in the table below

Situation	Characterization
$E \bullet\!\overset{1}{-}\, R \overset{1}{-}\!\bullet F$	$\mathsf{DotDot}(E, R, F)$ & $\mathsf{LUniq}(R)$ & $\mathsf{RUniq}(R)$
$E \bullet\!\overset{1}{-}\, R -\!\bullet F$	$\mathsf{DotDot}(E, R, F)$ & $\mathsf{LUniq}(R)$
$E \bullet\!-\, R \overset{1}{-}\!\bullet F$	$\mathsf{DotDot}(E, R, F)$ & $\mathsf{RUniq}(R)$

4.2 Adding Place Holders

It may sometimes happen that information is incomplete: an element x is inserted into entity E, and $E \bullet\!- R -\!\bullet F$ holds, but there is no y in F so that $\langle x, y\rangle$ is to be inserted into R. This then would violate the condition $E \circ \mathbb{1} \subseteq R \circ \mathbb{1}$. There may even occur some unpleasant situations when place holders are not admitted. Consider Fig. 1, where E and F are assumed to be different entities. Insert *one* into E, then $\langle one, two\rangle$ into R; then *two* must be new to F. Insert it into F, then it will be inserted into F' which requires the insertion of a pair $\langle three, two\rangle$ into S; *three* must be new to E. In this way a loop is created which will not terminate

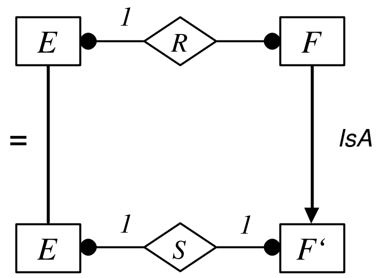

Figure 1: Possible Circularity

For enabling insertions also under somewhat problematic conditions, we postulate the existence of place holders which are collected in a relation P, so that in the situation considered $\langle x, *\rangle$ with $* \in P$ would be inserted into R. We assume that $\mathsf{Coll}(P)$ holds, and that the entities are free of place holders, thus $E \cap P = \emptyset$ is true for each entity E (note that this implies both $\mathbb{1} \circ E \cap \mathbb{1} \circ P = \emptyset$ and $E \circ \mathbb{1} \cap P \circ \mathbb{1} = \emptyset$ by Lemma 1). Let

$$\mathsf{Entity}(P, E) \equiv_{\mathrm{Def}} \mathsf{Coll}(E) \;\&\; E \cap P = \emptyset$$

denote that E is an entity.

Inheritance Immediate inheritance between entities is given through the IsA- relation. Hence E IsA F translates into

$$\mathsf{Inherits}(P, E, F) \equiv_{\mathrm{Def}} \mathsf{Entity}(P, E) \;\&\; \mathsf{Entity}(P, F) \;\&\; E \subseteq F.$$

There are some restrictions to be observed concerning the IsA-relation, mainly acyclicity and single inheritance, and the reader is referred to the companion paper [11] for details.

Constraints on place holders In [11] some constraints on the use of place holders were formulated:

1. No placeholder occurs twice as the first or the second component of a pair in a relation R. Put
$$\mathsf{NoTwice}(P, R) \equiv_{\mathrm{Def}} P \cap (R \cap R \circ \overline{\iota}) \circ 1\!\!1 = \emptyset,$$
then
$$\mathsf{NoTwice}(P, R) \,\&\, \mathsf{NoTwice}\big(P, R^{-1}\big)$$
should hold,

2. No placeholder occurs in two different relations R, S as the first components of a pair, which is formulated as
$$\mathsf{NoBoth}(P, R, S) \equiv_{\mathrm{Def}} P \cap R \circ 1\!\!1 \cap S \circ 1\!\!1 = \emptyset.$$

3. No placeholder occurs both as the first component in relation R and as the second component in relation S, hence
$$\mathsf{NoFirstSecond}(P, R, S) \equiv_{\mathrm{Def}} \mathsf{NoBoth}\big(P, R, S^{-1}\big).$$

4. No pair in a relation has place holders on both sides, thus
$$\mathsf{NoSamePair}(P, R) \equiv_{\mathrm{Def}} R \cap P \circ 1\!\!1 \cap 1\!\!1 \circ P = \emptyset.$$

5. The situation $\langle *, y\rangle$ and $\langle x, y\rangle$ with $x \neq *$ (and, for symmetry, in the second component) does not occur; this is captured through
$$\mathsf{NoDoubleFirst}(P, R) \equiv_{\mathrm{Def}} R \circ R^{-1} \cap P \circ 1\!\!1 \cap 1\!\!1 \circ \overline{P} = \emptyset$$
and
$$\mathsf{NoDoubleSecond}(P, R) \equiv_{\mathrm{Def}} R^{-1} \circ R \cap 1\!\!1 \circ P \cap \overline{P} \circ 1\!\!1 = \emptyset$$

Summing up: If $\{R_1, \ldots, R_k\}$ are the identifiers for all the relations in play, the conjunction $\mathsf{PlaceHolder}(P, \{R_1, \ldots, R_k\})$ should hold, where

$$\begin{array}{rcl}
\mathsf{PlaceHolder}(P, \{R_1, \ldots, R_k\}) & \equiv_{\mathrm{Def}} & \\
 & & \&_{i=1}^{k} \mathsf{NoTwice}(P, R_i) \,\&\, \mathsf{NoTwice}\big(P, R_i^{-1}\big) \\
 & \& & \&_{i=1}^{k} \&_{j=i+1}^{k} \mathsf{NoBoth}(P, R_i, R_j) \\
 & \& & \&_{i=1}^{k} \&_{j=i}^{k} \mathsf{NoFirstSecond}(P, R_i, R_j) \\
 & \& & \&_{i=1}^{k} \mathsf{NoSamePair}(P, R_i) \\
 & \& & \&_{i=1}^{k} \mathsf{NoDoubleFirst}(P, R_i) \\
 & \& & \&_{i=1}^{k} \mathsf{NoDoubleSecond}(P, R_i)
\end{array}$$

We reserve the map letter π for place holders.

5 Insertions: Validity

This section formulates the validity of an ER model. For this, we first have a look at the available map letters and at their arrangement. Then we discuss the validity of the model at first without taking attributes into account. This leads to the notion of *weak validity*. Conditions are formulated under which the weak validity of an ER model is preserved. Then we add attributes to our discussion, and the notion of validity is formulated. Again, conditions are given under which the attributes of the model arising from insertions satisfy the constraints, this time leading to the instance of a valid ER model.

5.1 Map Letters

We assume that we have countably many map letters $\mathsf{r}_1, \mathsf{r}_2, \ldots$ at our disposal, of which we reserve the first T initially for system purposes. We have reserved already π for place holders. Some additional reservations will have to be done.
The map letters with indices beyond T will be used for the ER model under consideration in the following way. $\mathsf{r}_{T+1}, \ldots, \mathsf{r}_{T+S}$ will be reserved for entities, the next block of B map letters $\mathsf{r}_{T+S+1}, \ldots, \mathsf{r}_{T+S+B}$ will be reserved for relations, and finally we will reserve the next block of A map letters for attributes. In case of an insertion or a deletion, we reserve the next block of S map letters for the δ^+ resp. δ^--values for entities, the next block of size B for those values for relations, and finally the next block A map letters for attributes. We continue the sequence with the results, according to the following scheme (with $\Sigma := S + B + A$): if entity E corresponds to map letter r_{T+i} with δ^+E corresponding to $\mathsf{r}_{T+\Sigma+i}$, then $E \cup \delta^+E$ will be deposited at $\mathsf{r}_{T+2\cdot\Sigma+i}$. In the same linear way — proceeding in a block wise fashion — we deposit the changed values for relations and attributes. The arrangement of map letters is indicated in Fig. 2.

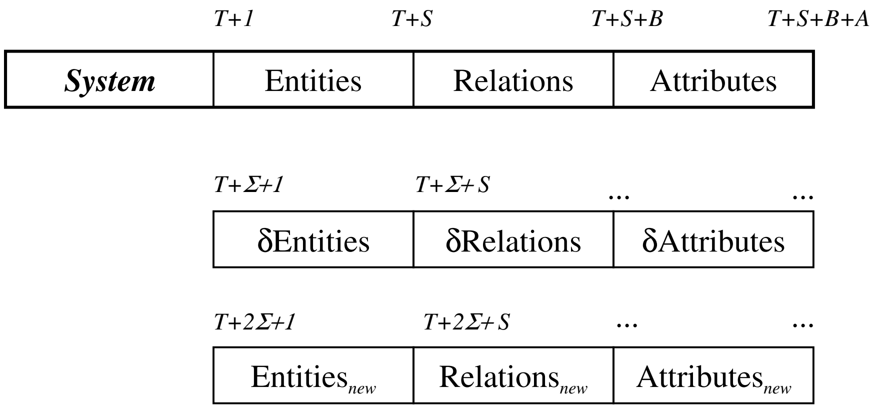

Figure 2: Arrangement of Map Letters

5.2 Keeping track

We keep record the respective relations between entities and relations through a set-valued map
$$\mathsf{Track} : \{T+S+1, \ldots, T+S+B\} \to 2^{\{T+1,\ldots,T+S\}\times\{T+1,\ldots,T+S\}}$$
upon setting
$$\langle i, j\rangle \in \mathsf{Track}(t) \Leftrightarrow \mathsf{r}_i \bullet\!\!- \mathsf{r}_t -\!\!\bullet \mathsf{r}_j.$$
Define for the relational index $t \in \{T+S+1, \ldots, T+S+B\}$

$$\begin{aligned} t \in \mathsf{LeftOne} &\quad\Leftrightarrow\quad \exists i, j : \mathsf{r}_i \bullet\!\!\overset{1}{-} \mathsf{r}_t -\!\!\bullet \mathsf{r}_j \\ t \in \mathsf{RightOne} &\quad\Leftrightarrow\quad \exists i, j : \mathsf{r}_i \bullet\!\!- \mathsf{r}_t \overset{1}{-}\!\!\bullet \mathsf{r}_j \end{aligned}$$

Through these sets we get access to left- and right-unique relations. Note that $\mathsf{r}_i \bullet\!\!\overset{1}{-} \mathsf{r}_t -\!\!\bullet \mathsf{r}_j$ implies $\mathsf{r}_k \bullet\!\!\overset{1}{-} \mathsf{r}_t -\!\!\bullet \mathsf{r}_\ell$ for all r_k that form the domain of r_t.
Again, $\mathsf{Track}, \mathsf{LeftOne}$ and $\mathsf{RightOne}$ can be shifted linearly along each Σ-block of indices.
The reflexive and transitive closure IsA^* of the inheritance relation is recorded through a reflexive and transitive relation Up on the set $\{T+1, \ldots, T+S\}$; note that this relation may be shifted linearly to the sets $\{T+k\cdot\Sigma+1, \ldots, T+k\cdot\Sigma+S\}$. The necessary properties of IsA^* are described in [11].

Attributes If entity E is represented by map letter r_i with $i \in \{T+1, \ldots, T+S\}$, then
$$\mathsf{Attributes}(i) \subseteq \{T+S+B+1, \ldots, T+S+B+A\}$$
is the set of map letters that are associated with E's attributes. Clearly,
$$\{\mathsf{Attributes}(i) \mid T+1 \le i \le T+S\}$$
forms a partition of the set $\{T+S+B+1, \ldots, T+S+B+A\}$. The set
$$\mathsf{Mandatory}(i) \subseteq \mathsf{Attributes}(i)$$
contains the indices of all mandatory attributes (those attributes which are defined on all of E), and the set
$$\mathsf{Key}(i) \subseteq \mathsf{Mandatory}(i)$$
contains all indices of the key attributes. We assume having only one set of key attributes per entity. It would be easy to work with a varying number of sets of keys for each entity, but this would only complicate the notation, without adding any new ideas.
When we execute an insertion or a deletion, we change the contents of the map letters by manipulating the extension of the corresponding data containers. Our block oriented scheme ensures that this process can be repeated without much ado by simply changing the base address where it all begins from T to $T+2\cdot\Sigma$.

5.3 Weak Validity

An instance $\mathcal{M}$ of the ER model under consideration is *weakly valid* iff it satisfies all the constraints imposed on the entities and the relations laid down in the model's declaration. This can be described now formally:

Definition 3 *The instance $\mathcal{M}$ is called* weakly valid *iff*

$$
\begin{array}{lll}
\&_{T+S+1\leq t\leq T+S+}\&_{\langle i,j\rangle\in\mathsf{Track}(t)}\mathsf{DotDot}(\mathsf{r}_i,\mathsf{r}_t,\mathsf{r}_j) & \& & \&_{t\in\mathsf{LeftOne}}\mathsf{LUniq}(\mathsf{r}_t)\\
& \& & \&_{t\in\mathsf{RightOne}}\mathsf{RUniq}(\mathsf{r}_t)\\
& \& & \&_{T+1\leq i\leq T+S}\ \mathsf{Entity}(\pi,\mathsf{r}_i)\\
& \& & \mathsf{PlaceHolder}(\pi,\{\mathsf{r}_{T+1},\ldots,\mathsf{r}_{T+S}\})\\
& \& & \&_{\langle i,j\rangle\in\mathsf{Up}}\ \mathsf{r}_i\subseteq\mathsf{r}_j
\end{array}
$$

Note that weak validity is formulated using a fixed base address T, which, however, has not been incorporated into the notation that is already cluttered enough.

5.4 Maintaining Weak Validity

The insertions to be performed start from a weakly valid ER model and should of course maintain weak validity as an invariant; this issue is discussed in 1.1 at greater length. We will need some preconditions. Before formulating them, however, we elaborate on the insertions proper. If E is an entity, and δ^+E contains the insertions into E, then $E\cup\delta^+E$ will be formed, and this will be the new version of this entity. It is a bit more complicated with a relation R, since we cannot simply form $R\cup\delta^+R$ without running the risk of violating $\mathsf{NoDoubleFirst}(P,R\cup\delta^+R)$ or $\mathsf{NoDoubleSecond}(P,R\cup\delta^+R)$. Hence we have to clean up R by removing candidates for violations; they are easily seen to belong to

$$(\mathbb{1}\circ\pi\cap\delta^+R\circ\mathbb{1})\cup(\pi\circ\mathbb{1}\cap\mathbb{1}\circ\delta^+R).$$

Thus we work with

$$\left[R,\delta^+R\right]\equiv_{\text{Def}}R\setminus\left((\mathbb{1}\circ\pi\cap\delta^+R\circ\mathbb{1})\cup(\pi\circ\mathbb{1}\cap\mathbb{1}\circ\delta^+R)\right)$$

instead of R and form $[R,\delta^+R]\cup\delta^+R$ as the new version of relation R. Occasionally we will replace the map letter π by the free variable P; the expression then will be denoted by $[R,\delta^+R]_P$.
For describing under which conditions weak validity is maintained, we need preparations.

Lemma 2 *Let R be a relation, and assume* $\mathsf{Entity}(P,E)$. *Then these implications hold:*

1.
$$\frac{\begin{array}{c}\mathsf{DomSub}(E,R)\ \ \mathsf{Entity}(P,E)\\ \mathsf{DomSub}(\delta^+E,\delta^+R)\ \ \mathsf{Entity}(P,\delta^+E)\\ (R\cap\mathbb{1}\circ P\cap\delta^+R\circ\mathbb{1})\circ\mathbb{1}\cap(E\cup\delta^+E)\circ\mathbb{1}=\emptyset\end{array}}{\mathsf{DomSub}(E\cup\delta^+E,[R,\delta^+R]_P\cup\delta^+R)}$$

2.

$$\frac{\begin{array}{c}\mathsf{ImgSub}(F,R)\ \mathsf{Entity}(P,F)\\ \mathsf{ImgSub}(\delta^+F,\delta^+R)\ \mathsf{Entity}(P,\delta^+F)\\ \mathbb{1}\circ(R\cap P\circ\mathbb{1}\cap\mathbb{1}\circ\delta^+R)\cap\mathbb{1}\circ(F\cup\delta^+F)=\emptyset\end{array}}{\mathsf{ImgSub}(F\cup\delta^+F,[R,\delta^+R]_P\cup\delta^+R)}$$

Proof: Apparently,

$$(E\cup\delta^+E)\circ\mathbb{1}\subseteq(R\cup\delta^+R)\circ\mathbb{1},$$

and

$$(R\cup\delta^+R)\circ\mathbb{1}=\big([R,\delta^+R]_P\cup\delta^+R\big)\circ\mathbb{1}\cup A,$$

where

$$A:=\big(R\cap\mathbb{1}\circ\pi\cap\delta^+R\circ\mathbb{1}\big)\circ\mathbb{1}\cup\big(R\cap\pi\circ\mathbb{1}\cap\mathbb{1}\circ\delta^+R\big)\circ\mathbb{1}.$$

Now

$$\begin{array}{rcl}(E\cup\delta^+E)\circ\mathbb{1}\cap A&=&(E\cup\delta^+E)\circ\mathbb{1}\cap\big(R\cap\mathbb{1}\circ P\cap\delta^+R\circ\mathbb{1}\big)\circ\mathbb{1}\\ &\subseteq&(E\cup\delta^+E)\circ\mathbb{1}\cap P\circ\mathbb{1}\\ &=&\emptyset.\end{array}$$

This establishes *1.* In order to prove *2.*,

$$\mathbb{1}\circ(R\cup\delta^+R)$$

is decomposed similarly into

$$\mathbb{1}\circ([R,\delta^+R]_P\cup\delta^+R)$$

and a part that is shown to be disjoint from $F\cup\delta^+F$. ■

In a similar way, we can make sure that the new relation maintains its properties as a map, or as the inverse of a map:

Lemma 3 *Let R be a relation, then the following implications hold:*

1.

$$\frac{\begin{array}{c}\mathsf{LUniq}(R)\ \mathsf{LUniq}(\delta^+R)\\ [R,\delta^+R]\subseteq\overline{\overline{\iota}\circ\delta^+R}\\ \delta^+R\subseteq\overline{\overline{\iota}\circ[R,\delta^+R]}\end{array}}{\mathsf{LUniq}([R,\delta^+R]\cup\delta^+R)}$$

2.

$$\frac{\begin{array}{c}\mathsf{RUniq}(R)\ \mathsf{RUniq}(\delta^+R)\\ [R,\delta^+R]\subseteq\overline{\delta^+R\circ\overline{\iota}}\\ \delta^+R\subseteq\overline{[R,\delta^+R]\circ\overline{\iota}}\end{array}}{\mathsf{RUniq}([R,\delta^+R]\cup\delta^+R)}$$

Proof: From $\cup$-distributivity it is inferred that

$$\big([R,\delta^+R]\cup\delta^+R\big)\circ\big([R,\delta^+R]\cup\delta^+R\big)^{-1}\subseteq\iota,$$

since Schröder's Rule implies

$$\begin{array}{rcl}[R, \delta^+R] \circ (\delta^+R)^{-1} & \subseteq & \iota \\ \delta^+R \circ ([R, \delta^+R])^{-1} & \subseteq & \iota.\end{array}$$

This establishes property *1.* The other property is proved similarly. ■

It may be noted that both implications above can be reversed.

Use in what follows as abbreviations

$$\begin{array}{rcl}\Gamma(A, B, C) & \equiv_{\mathrm{Def}} & A \cap (B \cap (C \circ \overline{\iota})) \circ \mathbb{1} = \emptyset, \\ \Pi(A, B, C) & \equiv_{\mathrm{Def}} & A \cap (B \circ \mathbb{1} \cap C \circ \mathbb{1}) = \emptyset \\ \Psi(A, B, C) & \equiv_{\mathrm{Def}} & A \circ B^{-1} \cup B \circ A^{-1} \cup B \circ B^{-1} \subseteq C\end{array}$$

Lemma 4 *The following implications hold for relation R:*

1.

$$\frac{\mathsf{NoTwice}(P, R) \;\; \mathsf{NoTwice}(P, \delta^+R, \delta^+R) \quad \Gamma(P, [R, \delta^+R]_P, \delta^+R) \;\; \Gamma(P, \delta^+R, [R, \delta^+R]_P)}{\mathsf{NoTwice}(P, [R, \delta^+R]_P \cup \delta^+R)}$$

2.

$$\frac{\mathsf{NoBoth}(P, R, S) \;\; \mathsf{NoBoth}(P, \delta^+R, \delta^+S) \quad \Pi(P, \delta^+R, [S, \delta^+S]_P) \;\; \Pi(P, [R, \delta^+R]_P, \delta^+S)}{\mathsf{NoBoth}(P, [R, \delta^+R]_P \cup \delta^+R, [S, \delta^+S]_P \cup \delta^+S)}$$

3.

$$\frac{\mathsf{NoSamePair}(P, R) \;\; \mathsf{NoSamePair}(P, \delta^+R)}{\mathsf{NoSamePair}(P, [R, \delta^+R]_P \cup \delta^+R)}$$

4.

$$\frac{\mathsf{NoDoubleFirst}(P, R) \;\; \mathsf{NoDoubleFirst}(P, \delta^+R) \quad \delta^+R \circ [R, \delta^+R]_P^{-1} \subseteq \overline{P \circ \mathbb{1} \cap \mathbb{1} \circ \overline{P}}}{\mathsf{NoDoubleFirst}(P, [R, \delta^+R]_P \cup \delta^+R)}$$

5.

$$\frac{\mathsf{NoDoubleSecond}(P, R) \;\; \mathsf{NoDoubleSecond}(P, \delta^+R) \quad [R, \delta^+R]_P^{-1} \circ \delta^+R \subseteq \overline{\mathbb{1} \circ P \cap \overline{P} \circ \mathbb{1}}}{\mathsf{NoDoubleSecond}(P, [R, \delta^+R]_P \cup \delta^+R)}$$

6.

$$\frac{\mathsf{Inherits}(P, E, F) \;\; \mathsf{Entity}(P, \delta^+E) \;\; \mathsf{Entity}(P, \delta^+F) \;\; \delta^+E \subseteq F \cup \delta^+F}{\mathsf{Inherits}(P, E \cup \delta^+E, F \cup \delta^+F)}$$

Proof: The proofs depend on the algebraic laws imposed for a relational algebra. We give prototypical examples for proving these implications.

Regarding *1.*, $\cup$-distributivity implies

$$\begin{array}{rcl}P \cap (((X \cup Y) \cap (X \cup Y) \circ \overline{\iota}) \circ \mathbb{1})) & = & (P \cap ((X \cap X \circ \overline{\iota}) \circ \mathbb{1}))) \\ & & \cup \;\; (P \cap ((X \cap Y \circ \overline{\iota}) \circ \mathbb{1}))) \\ & & \cup \;\; (P \cap ((Y \cap X \circ \overline{\iota}) \circ \mathbb{1}))) \\ & & \cup \;\; (P \cap ((Y \cap Y \circ \overline{\iota}) \circ \mathbb{1})))\end{array}$$

Matching this against the definition, and against Γ yields the result. In a similar way *2.* is established. The distributive law for $\cup$ implies *3.* directly.
For establishing *4.*, put $\tau_\ell := P \circ \mathbb{1} \cap \mathbb{1} \circ \overline{P}$ as an abbreviation, then the condition together with Schröder's Rule yields

$$\delta^+ R \circ \left[R, \delta^+ R\right]_P^{-1} \cap \tau_\ell = \emptyset.$$

Consequently, by $\cup$-distributivity,

$$\left[R, \delta^+ R\right]_P \circ \delta^+ R^{-1} \cap \tau_\ell = \emptyset$$

needs to be established. Schröder's Rule again shows this to be equivalent to

$$\tau_\ell \circ \delta^+ R \subseteq \overline{[R, \delta^+ R]_P},$$

which in turn may be seen from

$$\begin{aligned} \tau_\ell \circ \delta^+ R &\subseteq P \circ \mathbb{1} \circ \delta^+ R \cap \mathbb{1} \circ \overline{P} \circ \delta^+ R \\ &\subseteq P \circ \mathbb{1} \circ \mathbb{1} \cap \mathbb{1} \circ \mathbb{1} \circ \delta^+ R \\ &= P \circ \mathbb{1} \cap \mathbb{1} \circ \delta^+ R. \end{aligned}$$

The inference *5.* is established in a very similar way. Finally, *6.* is obvious. ■
Define the set $\mathsf{Related}(t)$ as the smallest subset K of $\{T+1, \ldots, T+S+B\}$ with these properties:

- $t \in K$,
- if $u \in K$ and $\langle u, v \rangle \in \mathsf{Up}$, then $v \in K$,
- if $\mathsf{r}_i \bullet\!\!- \mathsf{r}_j$ or $\mathsf{r}_j -\!\!\bullet \mathsf{r}_i$, then $i \in K$ iff $j \in K$.

Thus if we want to insert something into, say, entity E, and E corresponds to map letter r_i, then $\mathsf{Related}(i)$ contains the indices of exactly those entities and relations which are affected by this insertion.
Now let an entity or a relation correspond to map letter r_t. An insertion or a deletion is called *local at t* iff $\mathsf{r}_s = \emptyset$ whenever $s \in \{T+\Sigma, \ldots, T+2\cdot\Sigma+1\} \setminus \mathsf{Related}(t)$. Introducing this guard prevents the insertion or the deletion from violating the invariants for the model by letting properties creeping in that are not really controlled through our safety measures.
From the instance $\mathcal{M}$ a new instance $\mathcal{M}'$ is generated by performing the insertions. Put for each $j \in \{1, \ldots S\}$

$$\mathsf{r}_{T+2\cdot\Sigma+j} := \mathsf{r}_{T+j} \cup \mathsf{r}_{T+\Sigma+j}.$$

This accounts for insertions into entities. As far as relations are concerned, we set for each $j \in \{S+1, \ldots, B\}$

$$\mathsf{r}_{T+2\cdot\Sigma+j} := \left[\mathsf{r}_{T+j}, \mathsf{r}_{T+\Sigma+j}\right] \cup \mathsf{r}_{T+\Sigma+j},$$

accounting for the peculiar way we insert into a relation.
Upon shifting the base address from T to $T+2\cdot\Sigma$, the weak validity of $\mathcal{M}'$ can be investigated:

Proposition 1 *Let $\mathcal{M}$ be a weakly valid ER model, assume that an insertion is local at some index t, then the ER model arising from the insertions is weakly valid, provided the following conditions are all satisfied:*

1. $\&_{s\in\{1,\ldots,S\}}\ \mathsf{Entity}(\pi,\mathsf{r}_{T+\Sigma+s})$,

2. $\&_{s\in\mathsf{Related}(t)\cap\{T+1,\ldots,T+B\}}$
$\&_{\langle i,j\rangle\in\mathsf{Track}(s)}\ \mathsf{DomSub}(\mathsf{r}_{\Sigma+i},\mathsf{r}_{\Sigma+s})$
$\&\ \mathsf{Entity}(\mathsf{r}_{\Sigma+i})\ \&\ \mathsf{ImgSub}(\mathsf{r}_{\Sigma+j},\mathsf{r}_{\Sigma+s})\ \&\ \mathsf{Entity}(\mathsf{r}_{\Sigma+j})$
$\&\ (\mathsf{r}_s\cap\mathbb{1}\circ\pi\cap\mathsf{r}_{\Sigma+s}\circ\mathbb{1})\circ\mathbb{1}\cap\mathsf{r}_{2\cdot\Sigma+i}\circ\mathbb{1}=\emptyset$
$\&\ \mathbb{1}\circ(\mathsf{r}_s\cap\pi\circ\mathbb{1}\cap\mathbb{1}\circ\mathsf{r}_{\Sigma+s})\cap\mathbb{1}\circ\mathsf{r}_{2\cdot\Sigma+i}=\emptyset$

3. $\&_{s\in\mathsf{LeftOne}\cap\mathsf{Related}(t)}\ \mathsf{r}_{\Sigma+s}\circ\mathsf{r}_{\Sigma+s}^{-1}\subseteq\boldsymbol{\iota}\ \&\ [\mathsf{r}_s,\mathsf{r}_{\Sigma+s}]\subseteq\overline{\overline{\boldsymbol{\iota}}\circ\mathsf{r}_{\Sigma+s}}\ \&\ \mathsf{r}_{\Sigma+s}\subseteq\overline{\overline{\boldsymbol{\iota}}\circ[\mathsf{r}_s,\mathsf{r}_{\Sigma+s}]}$

4. $\&_{s\in\mathsf{RightOne}\cap\mathsf{Related}(t)}\ \mathsf{r}_{\Sigma+s}^{-1}\circ\mathsf{r}_{\Sigma+s}\subseteq\boldsymbol{\iota}\ \&\ [\mathsf{r}_s,\mathsf{r}_{\Sigma+s}]\subseteq\overline{\mathsf{r}_{\Sigma+s}\circ\overline{\boldsymbol{\iota}}}\ \&\ \mathsf{r}_{\Sigma+s}\subseteq\overline{[\mathsf{r}_s,\mathsf{r}_{\Sigma+s}]\circ\overline{\boldsymbol{\iota}}}$

5. $\&_{s\in\mathsf{Related}(t)\cap\{T+S+1,\ldots,T+S+B\}}$
$\pi\cap(\mathsf{r}_{\Sigma+s}\cap\mathsf{r}_{\Sigma+s}\circ\overline{\boldsymbol{\iota}}\circ\mathbb{1})=\emptyset\ \&\ \pi\cap\left(\mathsf{r}_{\Sigma+s}^{-1}\cap(\mathsf{r}_{\Sigma+s}^{-1}\circ\overline{\boldsymbol{\iota}})\circ\mathbb{1}\right)=\emptyset$
$\&\ \Gamma(\pi,[\mathsf{r}_s,\mathsf{r}_{\Sigma+s}],\mathsf{r}_{\Sigma+s})\ \&\ \Gamma(\pi,[\mathsf{r}_s^{-1},\mathsf{r}_{\Sigma+s}^{-1}],\mathsf{r}_{\Sigma+s}^{-1})$
$\&\ \Gamma(\pi,\mathsf{r}_{\Sigma+s},[\mathsf{r}_s,\mathsf{r}_{\Sigma+s}])\ \&\ \Gamma(\pi,\mathsf{r}_{\Sigma+s}^{-1},[\mathsf{r}_s^{-1},\mathsf{r}_{\Sigma+s}^{-1}])$

6. $\&_{s\in\mathsf{Related}(t)\cap\{T+S+1,\ldots,T+S+B\}}$
$\&_{v\in\mathsf{Related}(t)\cap\{s,\ldots,T+S+B\}}$
$\&\ \pi\cap(\mathsf{r}_{\Sigma+s}\circ\mathbb{1})\cap(\mathsf{r}_{\Sigma+v}\circ\mathbb{1})=\emptyset$
$\Pi(\pi,\mathsf{r}_{\Sigma+s},[\mathsf{r}_v,\mathsf{r}_{\Sigma+v}])\ \&\ \Pi(\pi,[\mathsf{r}_s,\mathsf{r}_{\Sigma+s}],\mathsf{r}_v)$

7. $\&_{s\in\mathsf{Related}(t)\cap\{T+S+1,\ldots,T+S+B\}}\ \mathsf{r}_{\Sigma+s}\cap\pi\circ\mathbb{1}\cap\mathbb{1}\circ\pi=\emptyset$

8. $\&_{s\in\mathsf{Related}(t)\cap\{T+S+1,\ldots,T+S+B\}}$
$\mathsf{r}_{\Sigma+s}\circ\mathsf{r}_{\Sigma+s}^{-1}\cap\pi\circ\mathbb{1}\cap\mathbb{1}\circ\overline{\pi}=\emptyset$
$\&\ \mathsf{r}_{\Sigma+s}\circ[\mathsf{r}_s,\mathsf{r}_{\Sigma+s}]^{-1}\subseteq\overline{\pi\circ\mathbb{1}\cap\mathbb{1}\circ\overline{\pi}}$

9. $\&_{s\in\mathsf{Related}(t)\cap\{T+S+1,\ldots,T+S+B\}}$
$\mathsf{r}_{\Sigma+s}^{-1}\circ\mathsf{r}_{\Sigma+s}\cap\mathbb{1}\circ\pi\cap\overline{\pi}\circ\mathbb{1}=\emptyset$
$\&\ [\mathsf{r}_s,\mathsf{r}_{\Sigma+s}]^{-1}\circ\mathsf{r}_{\Sigma+s}\subseteq\overline{\mathbb{1}\circ\pi\cap\overline{\pi}\circ\mathbb{1}}$

10. $\&_{\langle i,j\rangle\in\mathsf{Up}\cap\mathsf{Related}(T)\times\mathsf{Related}(T)}\ \mathsf{Entity}(\pi,\mathsf{r}_{\Sigma+i})\ \&\ \mathsf{Entity}(\pi,\mathsf{r}_{\Sigma+j})\ \&\ \mathsf{r}_{\Sigma+i}\subseteq\mathsf{r}_j\cup\mathsf{r}_{\Sigma+j}$

Proof: 0. This looks at first like a confusing bag of details. So let us sort them out by providing a brief table which permits some correspondence between the properties stated in the Proposition, and the properties of an ER model.

Item#	Property addressed
1	Entities are preserved
2	Domain properties are preserved
3	Left uniqueness
4	Right uniqueness
5	NoTwice
6	NoBoth
7	NoSamePair
8	NoDoubleFirst
9	NoDoubleSecond
10	Inheritance is preserved

1. Property *1.* establishes together with the assumption

$$\&_{T+1\leq j\leq T+S}\ \mathsf{Entity}(\mathsf{r}_j)$$

that

$$\&_{T+2\cdot\Sigma+1\leq j\leq T+2\cdot\Sigma+S}\ \mathsf{Entity}(\mathsf{r}_j)$$

is true.
2. Take $s \in \mathsf{Related}(t)$, and assume that $\langle i,j\rangle \in \mathsf{Track}(s)$. Because $\mathcal{M}$ is weakly valid, we know that $\mathsf{DotDot}(\mathsf{r}_i, \mathsf{r}_s, \mathsf{r}_j)$ holds. In particular,

$$\mathsf{DomSub}(\mathsf{r}_i, \mathsf{r}_s)\ \&\ \mathsf{Entity}(\mathsf{r}_i)$$

are true. This implies together with

$$\mathsf{DomSub}(\mathsf{r}_{\Sigma+i}, \mathsf{r}_{\Sigma+s})\ \&\ \mathsf{Entity}(\mathsf{r}_{\Sigma+i})\ \&\ (\mathsf{r}_s \cap \mathbb{1} \circ \pi \cap \mathsf{r}_{\Sigma+s} \circ \mathbb{1}) \circ \mathbb{1} \cap \mathsf{r}_{2\cdot\Sigma+i} \circ \mathbb{1} = \emptyset$$

through Lemma 2 (Property 1.) that $\mathsf{DomSub}(\mathsf{r}_{2\cdot\Sigma+i}, \mathsf{r}_{2\cdot\Sigma+s})$ holds. In a similar way (by appealing to Lemma 2, *2.*), $\mathsf{ImgSub}(\mathsf{r}_{\Sigma+j}, \mathsf{r}_s)$ is established. Collecting things, we have established that

$$\mathsf{DotDot}(\mathsf{r}_{2\cdot\Sigma+i}, \mathsf{r}_{2\cdot\Sigma+s}, \mathsf{r}_{2\cdot\Sigma+j})$$

is true.
3. Let $s \in \mathsf{LeftOne} \cap \mathsf{Related}(t)$, then we know from $\mathcal{M}'s$ validity that $\mathsf{r}_s \circ \mathsf{r}_s \subseteq \boldsymbol{\iota}$ holds. From Lemma 3, Property *1.* we now see that

$$\mathsf{LUniq}(\mathsf{r}_{2\cdot\Sigma+s})$$

is true, provided *3.* holds. In a very similar manner,

$$\mathsf{RUniq}(\mathsf{r}_{2\cdot\Sigma+s})$$

is deduced from *4.* for $s \in \mathsf{RightOne} \cap \mathsf{Related}(t)$.
4. From *5.* we infer that

$$\&_{T+2\cdot\Sigma+S+1\leq j\leq T+2\cdot\Sigma+S+B}\ \mathsf{NoTwice}(\pi, \mathsf{r}_j)\ \&\ \mathsf{NoTwice}\Big(\pi, \mathsf{r}_j^{-1}\Big)$$

holds (where we use Lemma 1 to establish that the identity $[\mathsf{r}_{T+j}, \mathsf{r}_{T+\Sigma+j}]^{-1} = \Big[\mathsf{r}_{T+j}^{-1}, \mathsf{r}_{T+\Sigma+j}^{-1}\Big]$ holds.
5. In similar ways one establishes the desired properties, resorting to Lemma 4 for establishing the necessary conditions. ∎

The conditions formulated above look certainly very technical, so let us interpret the second and the last of them. The former one states conditions under which

$$E \cup \delta^+E \bullet\!\!- R \cup \delta^+R -\!\!\bullet F \cup \delta^+F$$

holds, i.e., under which conditions $E \cup \delta^+E$ and $F \cup \delta^+F$ remain the tight domain and the tight codomain, resp., of $[R, \delta^+R] \cup \delta^+R$, provided E was the tight domain, and F was the tight codomain of R before the insertion, i.e., provided

$$E \bullet\!\!- R -\!\!\bullet F$$

holds. The conditions state that δ^+E needs to be an entity such that

$$\mathsf{dom}\big(\delta^+E\big) \subseteq \mathsf{dom}\big(\delta^+R\big)$$

is true, hence each element to be inserted into E should be the first component of a pair to be inserted into R. In the same way δ^+F is required to be an entity such that

$$\mathsf{img}\big(\delta^+F\big) \subseteq \mathsf{img}\big(\delta^+R\big)$$

holds. In addition we make sure that the required conditions on place holders are not violated, so that

$$\begin{aligned}\big(R \cap \mathbb{1} \circ P \cap \delta^+R \circ \mathbb{1}\big) \circ \mathbb{1} \cap (E \cup \delta^+E) \circ \mathbb{1} &= \emptyset. \\ \mathbb{1} \circ \big(R \cap P \circ \mathbb{1} \cap \mathbb{1} \circ \delta^+R\big) \cap \mathbb{1} \circ (F \cup \delta^+F) &= \emptyset\end{aligned}$$

holds, as we have discussed above.
The last condition simply states that for $E \cup \delta^+E$ to inherit from $R \cup \delta^+R$ it is sufficient that E inherits from F, and that δ^+E is a subset of $F \cup \delta^+F$, and that the new sets are entities indeed. Similar interpretations are given for the other conditions; this is left to the reader.

5.5 Looking at Attributes

Attributes are defined on entities (this is one of our restrictions, cp. Sect. 2.1), they come in different flavors, as we will discuss now. An attribute α on entity E is a partial map, so $\mathsf{LUniq}(\alpha)$ should be satisfied, and its domain should be contained in (the domain of) E, thus

$$\mathsf{dom}(\alpha) \subseteq \mathsf{dom}(E)$$

should hold. Moreover we assume attributes to have atomic values.
This requirement will be modelled as follows: We assume our universe $\mathcal{U}$ to be structured as

$$\mathcal{U} = \mathcal{A} \cup \mathcal{A}^*,$$

where $\mathcal{A} \neq \emptyset$ are the atomic values, and $\mathcal{A}^*$ denotes the set of all words over the alphabet $\mathcal{A}$, hence

$$\mathcal{A} \cap \mathcal{A}^* = \emptyset$$

with ϵ as the empty word; as usual, we put

$$\mathcal{A}^+ := \mathcal{A}^* \setminus \{\epsilon\}.$$

We reserve a map letter $\varepsilon \in \{\mathsf{r}_1, \ldots, \mathsf{r}_T\}$ for representing ϵ (hence $\mathsf{SngIt}(\varepsilon)$ & $\mathsf{Coll}(\varepsilon)$) and permit only interpretations $\mathcal{I}$ that satisfy $\varepsilon^{\mathcal{I}} = \{\langle\epsilon, \epsilon\rangle\}$. The atomic entities in $\mathcal{A}$ are modelled through the map letter υ with

$$\mathsf{Coll}(\upsilon) \;\&\; \mathsf{NonVoid}(\upsilon) \;\&\; \upsilon \cap \varepsilon = \emptyset.$$

In addition we postulate that $\pi \subseteq \upsilon$ holds.
Interpretations are restricted further by postulating that

$$\upsilon^{\mathcal{I}} = \{\langle a, b\rangle \in \mathcal{A}^2 \mid a = b\}.$$

Moreover we assume the existence of canonic projections CAR and CDR separating the head from the tail of a non-empty word, hence

$$\begin{aligned} \mathsf{CAR} : \mathcal{A}^+ \ni t_1 \dots t_k &\mapsto t_1 \in \mathcal{A}, \\ \mathsf{CDR} : \mathcal{A}^+ \ni t_1 \dots t_k &\mapsto t_2 \dots t_k \in \mathcal{A}^*. \end{aligned}$$

These projections are represented through the map letters λ and ρ, corresponding to CAR and CDR, resp; their properties will not be discussed here, the reader is referred to [11, 3.1]. We abbreviate for later use the i^{th} projection (hence the operation of extracting the i^{th} component of a tuple) by

$$\begin{aligned} Z^{(i)} &\equiv_{\text{Def}} (i = 1 \ ?\ \mathsf{CAR} :\ Z^{(i-1)} \circ \mathsf{CDR}) \\ \tau^{(i)} &\equiv_{\text{Def}} (i = 1 \ ?\ \lambda :\ \tau^{(i-1)} \circ \rho), \end{aligned}$$

the latter abbreviation preparing for the use of map letters later on.
Returning to attributes: a mandatory attribute α on entity E is characterized through

$$\mathsf{dom}(\alpha) = \mathsf{dom}(E) \ \&\ \mathbb{1} \circ \pi \cap \mathsf{img}(\alpha) = \emptyset.$$

If $\{\alpha_0, \dots \alpha_w\}$ is a collection of key attributes on E, then [11] shows that this property means

$$\mathsf{LUniq}\left(\bigcap_{i=0}^{w} Z^{(i+1)} \circ \alpha_i^{-1}\right)$$

to hold.

Lemma 5 *The following properties hold:*

1.
$$\frac{\mathsf{LUniq}(\alpha)\,\Psi(\alpha, \delta^+\alpha, \boldsymbol{\iota})}{\mathsf{LUniq}(\alpha \cup \delta^+\alpha)}$$

2.
$$\frac{\begin{array}{c}\mathsf{dom}(\alpha) = \mathsf{dom}(E) \\ (\delta^+\alpha \setminus \alpha) \circ \mathbb{1} = (\delta^+E \circ \mathbb{1}) \setminus (E \circ \mathbb{1})\end{array}}{\mathsf{dom}(\alpha \cup \delta^+\alpha) = \mathsf{dom}(E \cup \delta^+E)}$$

Proof: Both parts follows directly from $\cup$-distributivity. Note that the implication in the first part can be reversed. ■

The conditions laid down in Lemma 5 permit stating conditions under which some attribute conditions persist under insertion. The exception is a condition which permits being a member of a family of key attributes stable under insertions. The criterion is formulated in Lemma 6. It requires some preparations.
Remember that in a map algebra the equality

$$\bigcap_{i \in I} (A_{1,i} \cup A_{2,i}) = \bigcup_{J \subseteq I} \left(\bigcap_{j \in J} A_{1,j} \cap \bigcap_{j \notin J} A_{2,j} \right)$$

holds, whenever I is finite, since a map algebra is a Boolean algebra, thus in particular a distributive lattice. Abbreviate for the map expressions $A_0, \ldots, A_k, B_0, \ldots, B_k$ and for $J, K \subseteq \{0, \ldots, k\}$

$$\begin{aligned}
\Lambda(J, \langle A_0, \ldots, A_k\rangle, \langle B_0, \ldots, B_k\rangle) &\equiv_{\mathrm{Def}} \bigcap_{j \in J} A_j \circ \left(Z^{(j+1)}\right)^{-1} \cap \bigcap_{j \notin J} (B_j \setminus A_j) \circ \left(Z^{(j+1)}\right)^{-1}, \\
\Gamma(J, K, \langle A_0, \ldots, A_k\rangle, \langle B_0, \ldots, B_k\rangle) &\equiv_{\mathrm{Def}} \Lambda(J, \langle A_0, \ldots, A_k\rangle, \langle B_0, \ldots, B_k\rangle) \\
&\quad \circ \Lambda(K, \langle A_0, \ldots, A_k\rangle, \langle B_0, \ldots, B_k\rangle)^{-1}
\end{aligned}$$

With these notations we may formulate:

Lemma 6 *Invariance of a key under insertion is maintained by the following condition:*

$$\frac{\begin{array}{c} \&_{i=0}^{k} \mathsf{LUniq}(\alpha_i) \\ \&_{i=0}^{k} \mathsf{LUniq}(\delta^+\alpha_i) \\ \&_{J \subseteq \{0,\ldots,k\}} \&_{K \subseteq \{0,\ldots,k\}} \Gamma(J, K, \langle \alpha_0, \ldots, \alpha_k\rangle, \langle \delta^+\alpha_0, \ldots, \delta^+\alpha_k\rangle) \subseteq \iota \end{array}}{\mathsf{LUniq}\left(\bigcap_{i=0}^{k} Z^{(i+1)} \circ (\alpha_i \cup \delta^+\alpha_i)^{-1}\right)}$$

It should be noted that the formulation above requires

$$\langle \alpha_0, \ldots, \alpha_k \rangle$$

as well as

$$\langle \delta^+\alpha_0 \setminus \alpha_0, \ldots, \delta^+\alpha_k \setminus \alpha_k \rangle$$

to have the properties of key attributes.
Proof: The distributive law (in the lattice), $\cup$-distributivity (with respect to composition), and Lemma 1 together show that

$$\left(\bigcap_{i=0}^{k} Z^{(i+1)} \circ \left(\alpha_i \cup \delta^+\alpha_i\right)^{-1}\right)^{-1} \circ \left(\bigcap_{i=0}^{k} Z^{(i+1)} \circ \left(\alpha_i \cup \delta^+\alpha_i\right)^{-1}\right)$$

equals

$$\bigcup_{J \subseteq \{0,\ldots,k\}} \bigcup_{K \subseteq \{0,\ldots,k\}} \Gamma(J, K, \langle \alpha_0, \ldots, \alpha_k\rangle, \langle \delta^+\alpha_0, \ldots, \delta^+\alpha_k\rangle).$$

This implies the desired result. ■
The condition just formulated is exponential in the size of the key, consequently, it is not very convenient for practical purposes. On the other hand, it is exact, because a key can be extended if and only if the condition above is satisfied. It would be desirable to develop a more practical, if only sufficient condition for the invariance under insertions of the property being a key.
Now call an ER model $\mathcal{M}$ valid iff it is weakly valid, and if the conditions on attributes that have been laid down in the model's declaration are satisfied. Formally:

Definition 4 *The ER model $\mathcal{M}$ is called valid iff*

1. *$\mathcal{M}$ is weakly valid,*

2. *the attributes satisfy*

$$\&_{T+1\leq i\leq T+S}\&_{j\in\mathsf{Attributes}(i)}\ \mathsf{LUniq}(r_j)\ \&\ \mathsf{dom}(r_j)\subseteq\mathsf{dom}(r_i)\ \&\ \mathsf{img}(r_j)\subseteq\upsilon$$
$$\&$$
$$\&_{T+1\leq i\leq T+S}\&_{j\in\mathsf{Mandatory}(i)}\ \mathsf{dom}(r_j)=\mathsf{dom}(r_i)\ \&\ \mathbb{1}\circ\pi\cap\mathsf{img}(r_j)=\emptyset$$
$$\&$$
$$\&_{T+1\leq i\leq T+S}\ \textbf{\textit{let}}\ \{i_1,\ldots,i_j\}{=}\mathsf{Key}(i)\ \textbf{\textit{in}}\ \mathsf{LUniq}\Big(\bigcap\nolimits_{\ell=1}^{j}Z^{(\ell+1)}\circ\mathsf{r}_{i_\ell}^{-1}\Big)$$

We will state now conditions under which the attributes of the changes ER model $\mathcal{M}'$ will cater for the model's validity after the construction process is extended to attributes in the obvious way. Investigating validity requires us to exploit properties of the change sets $\delta^{+}\cdot$ for attributes in the context of their relations to the change sets for entities (note that we do for the time being without attributes on the relations on $\mathcal{M}$).

Proposition 2 *Suppose that the ER model $\mathcal{M}$ is valid, and that in addition to the properties 1 – 10 from Proposition 1 the following properties are satisfied, when performing an insertion that is local at some index t:*

1. $\&_{i\in\mathsf{Related}(t)\cap\{T+1,\ldots,T+B\}}\&_{j\in\mathsf{Attributes}(i)}\ \mathsf{r}_{\Sigma+j}^{-1}\circ\mathsf{r}_{\Sigma+j}\subseteq\boldsymbol{\iota}\ \&\ \Psi(\mathsf{r}_j,\mathsf{r}_{\Sigma+j},\boldsymbol{\iota})\ \&\ \mathbb{1}\circ\mathsf{r}_{\Sigma+j}\subseteq\upsilon$
2. $\&_{i\in\mathsf{Related}(t)\cap\{T+1,\ldots,T+B\}}\&_{j\in\mathsf{Mandatory}(i)}\ (\mathsf{r}_{\Sigma+j}\setminus\mathsf{r}_j)\circ\mathbb{1}=(\mathsf{r}_{\Sigma+i}\circ\mathbb{1})\setminus(\mathsf{r}_i\circ\mathbb{1})$
3. $\&_{i\in\mathsf{Related}(t)\cap\{T+1,\ldots,T+B\}}\&_{j\in\mathsf{Mandatory}(i)}\ \mathbb{1}\circ\pi\cap\mathbb{1}\circ\mathsf{r}_{\Sigma+j}=\emptyset$
4. $\&_{i\in\mathsf{Related}(t)\cap\{T+1,\ldots,T+B\}}$
$$\textbf{\textit{let}}\ \mathsf{Key}(i)=\{i_0,\ldots,i_k\}\ \textbf{\textit{in}}$$
$$\&_{J\subseteq\{0,\ldots,k\}}\&_{K\subseteq\{0,\ldots,k\}}\ \Gamma(J,K,\langle\mathsf{r}_{i_0},\ldots,\mathsf{r}_{i_k}\rangle,\langle\mathsf{r}_{\Sigma+i_0},\ldots,\mathsf{r}_{\Sigma+i_k}\rangle)\subseteq\boldsymbol{\iota}$$

Then $\mathcal{M}'$ is a valid ER model.

Proof: Lemma 5 makes sure that condition *1.* implies that we indeed obtain attributes, and that by condition *2.* mandatory attributes remain mandatory. Condition *3.* caters for banning place holders from the image of mandatory attributes. The last condition helps together with Lemma 6 in ascertaining the properties of keys in the new model. ■

6 Deletions: Validity

Deletions are treated in a similar fashion: we formulate conditions under which deletions maintain the validity of the ER model. We will deal initially with entities and relations only, and in a second step extend our considerations to attributes. This separation of concerns will again first formulate conditions under which weak validity is preserved, and then upgrade these conditions with the goal of finding criteria for unconstrained validity.

We will use the same initial setup of map letters as in sect. 5.1, but now interpret the map letters between $T+\Sigma+1$ and $T+2\cdot\Sigma$ as the place where we store the values to be deleted; they are now prefixed with δ^{-}. If entity E corresponds to map letter r_{T+i} with $\delta^{-}E$ corresponding to $\mathsf{r}_{T+\Sigma+i}$, then $E\setminus\delta^{-}E$ will be deposited at $\mathsf{r}_{T+2\cdot\Sigma+i}$. In the same linear way, proceeding in a block wise fashion, we deposit the changed values for relations and attributes. The reader may wish to consult Fig. 2 again.

6.1 Weak validity

The following observation shows that for maintaining weak validity we need not consider place holders separately, that left or right uniqueness of relations is of no concern, and that the defining property of key attributes remains intact, when deleting elements from the maps constituting the key:

Lemma 7 *The following implications hold:*

1.
$$\frac{R_1 \subseteq R_2 \ \mathsf{LUniq}(R_2)}{\mathsf{LUniq}(R_1)}$$

2.
$$\frac{R_1 \subseteq R_2 \ \mathsf{RUniq}(R_2)}{\mathsf{RUniq}(R_1)}$$

3.
$$\frac{\begin{array}{c}\&_{i=1}^{k} R_{1,i} \subseteq R_{2,i} \\ \mathsf{PlaceHolder}(P, \{R_{2,1}, \ldots, R_{2,k}\})\end{array}}{\mathsf{PlaceHolder}(P, \{R_{1,1}, \ldots, R_{1,k}\})}$$

4.
$$\frac{\begin{array}{c}\&_{i=1}^{k} R_{1,i} \subseteq R_{2,i} \\ \mathsf{LUniq}\Big(\bigcap_{i=1}^{n} T^{(i+1)} \circ R_{2,i}^{-1}\Big)\end{array}}{\mathsf{LUniq}\Big(\bigcap_{i=1}^{n} T^{(i+1)} \circ R_{1,i}^{-1}\Big)}$$

Proof: Because the composition operator $\circ$ is monotone in both arguments, the first two assertions are immediate. The monotonicity of the converse operator (which sends R to R^{-1}) is used on top of that in establishing the third assertion. This is done by inspecting the auxiliary macros that constitute the conjunction defining the PlaceHolder-macro, and that are formulated in sect. 4.2. Monotonicity of both operations is finally used to establish the last implication. ■

Thanks to Lemma 7, the technical base for maintaining weak validity in the following statement (which corresponds to Lemma 4 for insertions), is rather easier to formulate:

Lemma 8 *The following implications hold:*

1.
$$\frac{\begin{array}{c}\mathsf{DomSub}(E, R) \\ \mathsf{Entity}(E) \ \mathsf{Entity}(\delta^{-}E) \\ \mathsf{DomSub}(R, \delta^{-}E \cup R \setminus \delta^{-}R)\end{array}}{\mathsf{DomSub}(E \setminus \delta^{-}E, R \setminus \delta^{-}R)}$$

2.
$$\frac{\begin{array}{c}\mathsf{ImgSub}(F, R) \\ \mathsf{Entity}(F) \ \mathsf{Entity}(\delta^{-}F) \\ \mathsf{DomSub}\big(R^{-1}, \delta^{-}F^{-1} \cup (R \setminus \delta^{-}R)^{-1}\big)\end{array}}{\mathsf{ImgSub}(F \setminus \delta^{-}F, R \setminus \delta^{-}R)}$$

Proof: Only the first implication needs to be established, since the second follows by inversion. Since

$$R \circ \mathbb{1} \subseteq \left(\delta^- E \cup (R \setminus \delta^- R)\right) \circ \mathbb{1},$$

an elementary calculation establishes

$$(R \circ \mathbb{1}) \setminus (\delta^- E \circ \mathbb{1}) \subseteq (R \setminus \delta^- R) \circ \mathbb{1}.$$

Consequently,

$$\begin{aligned} (E \setminus \delta^- E) \circ \mathbb{1} &= (E \circ \mathbb{1}) \setminus (\delta^- E \circ \mathbb{1}) \\ &\subseteq (R \circ \mathbb{1}) \setminus (\delta^- E \circ \mathbb{1}) \text{ (since } E \circ \mathbb{1} \subseteq R \circ \mathbb{1}) \\ &\subseteq (R \setminus \delta^- R) \circ \mathbb{1}. \end{aligned}$$

This establishes the claim. ■

From the instance $\mathcal{M}$ a new instance $\mathcal{M}'$ is generated by performing the deletions. This is very similar to the insertion discussed above: Put for each $j \in \{1, \ldots, S + B\}$

$$\mathsf{r}_{T+2\cdot\Sigma+j} := \mathsf{r}_{T+j} \setminus \mathsf{r}_{T+\Sigma+j}.$$

Upon shifting the base address from T to $T+2\cdot\Sigma$, the weak validity of $\mathcal{M}'$ can be investigated:

Proposition 3 *Let $\mathcal{M}$ be a weakly valid ER model, assume that a deletion is local at some index t, then the ER model arising from the deletions is weakly valid, provided the following conditions are all satisfied:*

1. $\&_{s \in \mathsf{Related}(t) \cap \{T+1,\ldots,T+B\}}$
   $\&_{\langle i,j\rangle \in \mathsf{Track}(s)}\ \mathsf{Entity}(\mathsf{r}_{\Sigma+i})\ \&\ \mathsf{Entity}(\mathsf{r}_{\Sigma+j})$
   $\&\ \mathsf{DomSub}(\mathsf{r}_s, \mathsf{r}_{\Sigma+i} \cup (\mathsf{r}_s \setminus \mathsf{r}_{\Sigma+s}))\ \&\ \mathsf{DomSub}\left(\mathsf{r}_s^{-1}, \mathsf{r}_{\Sigma+j}^{-1} \cup (\mathsf{r}_s \setminus \mathsf{r}_{\Sigma+s})^{-1}\right)$

2. $\&_{\langle i,j\rangle \in \mathsf{Up} \cap \mathsf{Related}(t) \times \mathsf{Related}(t)}\ \mathsf{r}_{\Sigma+j} \subseteq \mathsf{r}_{\Sigma+i}$

Proof: Because condition *2.* takes care of inheritance, and because of Lemma 7 we have to establish only $\mathsf{DotDot}(\mathsf{r}_{2\cdot\Sigma+i}, \mathsf{r}_{2\cdot\Sigma+s}, \mathsf{r}_{2\cdot\Sigma+j})$ for all indices $s \in \mathsf{Related}(t)$, and for all $\langle i,j\rangle \in \mathsf{Track}(s)$. But this follows through a straightforward calculation from the assumption together with Lemma 8. ■

6.2 Adding Attributes

Turning to attributes, we see that the functional character of attributes together with that of their domains is maintained when changing to a subset of each:

Lemma 9 *Let α be an attribute on entity E, then the following implications show how to maintain attribute conditions under deletion:*

1. $$\frac{\begin{array}{c}\mathsf{LUniq}(\alpha)\ \mathsf{LUniq}(\delta^-\alpha) \\ \mathsf{Entity}(E)\ \mathsf{dom}(\alpha) \subseteq \mathsf{dom}(E) \\ (\alpha \setminus \delta^-\alpha)^{-1} \circ \delta^- E = \emptyset\end{array}}{\mathsf{dom}(\alpha \setminus \delta^-\alpha) \subseteq \mathsf{dom}(E \setminus \delta^- E)}$$

2.

$$\frac{\begin{array}{c}\mathsf{LUniq}(\alpha)\ \mathsf{LUniq}(\delta^{-}\alpha)\\ \mathsf{Entity}(E)\ \mathsf{dom}(\alpha)\subseteq\mathsf{dom}(E)\\ (\alpha\setminus\delta^{-}\alpha)^{-1}\circ\delta^{-}E=\emptyset\\ \mathsf{dom}(\delta^{-}E)\subseteq\mathsf{dom}(\delta^{-}\alpha)\end{array}}{\mathsf{dom}(\alpha\setminus\delta^{-}\alpha)=\mathsf{dom}(E\setminus\delta^{-}E)}$$

Proof: 1. Schröder's Cycle Rule implies that $(\alpha \setminus \delta^{-}\alpha)^{-1} \circ \delta^{-}E = \emptyset$ is equivalent to $\mathsf{dom}(\alpha \setminus \delta^{-}\alpha) \cap \mathsf{dom}(\delta^{-}E) = \emptyset$, thus

$$\begin{aligned}\mathsf{dom}\big(\alpha\setminus\delta^{-}\alpha\big) &\subseteq \mathsf{dom}(E)\setminus\mathsf{dom}\big(\delta^{-}E\big)\\ &= \mathsf{dom}\big(E\setminus\delta^{-}E\big).\end{aligned}$$

This proves *1.*
2. It remains to show that the domain of $\alpha \setminus \delta^{-}\alpha$ contains $E \setminus \delta^{-}E$ under the conditions from *2:* Using Lemma 1, we see

$$\begin{aligned}\mathsf{dom}\big(E\setminus\delta^{-}E\big) &= \mathsf{dom}(E)\setminus\mathsf{dom}\big(\delta^{-}E\big)\\ &\subseteq \mathsf{dom}(\alpha)\setminus\mathsf{dom}\big(\delta^{-}\alpha\big)\\ &\subseteq \mathsf{dom}\big(\alpha\setminus\delta^{-}\alpha\big)\end{aligned}$$

■

Now we are able to state conditions under which deletions from an ER model maintain its validity:

Proposition 4 *Suppose that the ER model $\mathcal{M}$ is valid, and that in addition to the properties 1 and 2 from Proposition 3 the following properties are satisfied, when performing an insertion that is local at some index t:*

1. $\&_{i\in\mathsf{Related}(t)\cap\{T+1,\dots,T+B\}}\&_{j\in\mathsf{Attributes}(i)}\ \mathsf{r}_{\Sigma+j}^{-1}\circ\mathsf{r}_{\Sigma+j}\subseteq\iota\ \&\ (\mathsf{r}_j\setminus\mathsf{r}_{\Sigma+j})^{-1}\circ\mathsf{r}_{\Sigma+i}=\emptyset$
2. $\&_{i\in\mathsf{Related}(t)\cap\{T+1,\dots,T+B\}}\&_{j\in\mathsf{Mandatory}(i)}\ \mathsf{r}_{\Sigma+i}\circ\mathbb{1}\subseteq\mathsf{r}_{\Sigma+j}\circ\mathbb{1}$

Proof: Since the weak validity of $\mathcal{M}'$ is already being taken care of by Proposition 3, we have to cater for the integrity of the attributes. Condition *1.* maintains together with Lemma 9 the condition under which the property of being an attribute is preserved, the second condition states when a mandatory attribute remains one; this also makes use of the same lemma. It was noted already in Lemma 7 that key attributes are not sensitive to deletions. ■

It comes as a surprise that deletions are so much easier to handle, than insertions. Some data structures (like binary search trees or heaps) are rather sensitive to deletions in the sense that a deletion from them requires much more attention in maintaining the invariant, than an insertion does (and consequently makes the analysis a much harder and more unpleasant undertaking). In the case of the data structures just mentioned, however, intrinsic properties of the keys do not play a role, while in the case considered here the monotonicity together with a downward closeness of some properties played a simplifying role.

7 Further Work

One line of attack for further work is removing some restrictions we imposed for technical reasons. We want to point to some, as we feel, interesting application of our formal set up. Model checking may be done in ER models, as we have formulated them. We will sketch the approach and propose further investigations along these lines.

7.1 Removing restrictions

This work was performed under some simplifying assumptions: we did assume that we work only with attributes on entities, and that we have a rather scant selection of cardinality restrictions. Both assumptions are not essential for our approach, and we feel that they should be removed. Another technical issue addresses the fact that we work with binary relations only. The discussions concerning projections shows, however, that it should not be too difficult to extend our set up for incorporating n-ary relations (although the notation then becomes slightly unbearable). From a modelling point of view, we work here in a somewhat untyped environment: we do not have sorts for different entities, but rather assume that one sort fits all. This is fairly problematic in applications, and not entirely practical. Introducing sorts is another step we feel should be undertaken (and a more detailed comparison of both approaches).

7.2 Model checking

We indicate briefly how the basic mechanisms of model checking can be applied by casting an ER model into the mold of model checking, following [3]. For this, the following ingredients are needed:

1. Properties to establish. These properties will be formed from a set AP of atomic propositions,
2. A Kripke structure for modelling different paths of the computations under consideration,
3. A temporal logic for formulating the properties. Here usually one of the subsets CTL or LTL of the Computational Tree Logic CTL^* is used. We assume that the reader is familiar with these logics and their semantics (a good starting point is [3, Ch. 3]).

7.2.1 A Simple Example

A student is examined orally by a professor, the exam results in a grade which is taken from the set $\{1, \ldots, 5\}$, 5 indicating *fail.* Figure 3 displays the ER model.
Each student's knowledge of a subject is modelled through a finite set of knowledge items taken from a set $\mathfrak{K}$, each professor's questions requires the student presenting some of these items. Associate with each student s a set $\kappa(s) \subseteq \mathfrak{K}$, and with each professor p a set $\pi(p) \subseteq \mathfrak{K}$. Apparently, the knowledge exchanged during one exam modelled through $\langle s, p\rangle$ is then $\kappa(s) \cap \pi(p)$. We are interested in exams that may be compared all over the state, say, hence we introduce an *objectivity function* $\omega : 2^{\mathfrak{K}} \to \{1, \ldots, 5\}$, so that the objective weight of exam $\langle s, p\rangle$ is just $\sigma(\langle s, p\rangle) := \omega(\kappa(s) \cap \pi(p))$. Now an exam may be measured against ω. Call the exam t

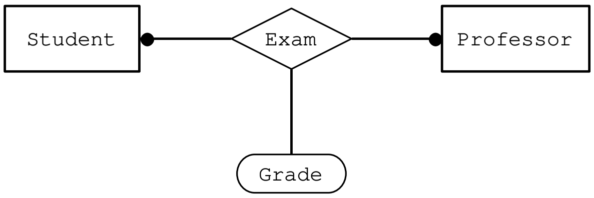

Figure 3: Oral examinations (simplified)

- *fair* iff $grade(t) = \sigma(t)$,
- *nice* iff $grade(t) < \sigma(t)$,
- *nasty* iff $grade(t) > \sigma(t)$.

Temporal statements for an instance of this ER model then include

1. Professor A gives only fair exams.
2. Sometimes Professor B gives a nasty exam.
3. Professor C gives fair exams until he examines a red haired cryptologist to whom he gives a nice exam.

7.2.2 Kripke structure

We recall the definitions of a Kripke structure:

Definition 5 $\mathcal{K} = (AP, S, R, S_0, L)$ *is called a* Kripke structure *with atomic propositions AP iff S is a finite set of states, $R \subseteq S \times S$ is a left-total relation on S, $S_0 \subseteq S$ is a set of initial states, and $L : S \to 2^{AP}$ is a function that labels states with atomic propositions.*

A path π in the Kripke structure $\mathcal{K}$ is an infinite sequence $\pi = s_0, s_1, \ldots$ of states such that $(s_i, s_{i+1}) \in R$ holds for each index i, hence represents an infinite branch in the computation tree that corresponds to the Kripke structure.
The *Computational Tree Logic* CTL^* describes properties of computation trees. Such a tree is given by (the states of) a Kripke structure: an initial state of $\mathcal{K}$ is taken as the root, and the structure expands into an infinite tree according to the relation R. For arguing about this tree, CTL^* offers path quantifiers (describing all, or only some, computation paths) and temporal operators (like *next time*, *eventually*, *always* or *until*). The semantics describes under which conditions a temporal formula is valid in a state, e.g. $\mathcal{K}, s \models \mathbf{A}[f_1 \mathbf{U} f_2]$ is true iff on all paths ("**A**") starting from state s the formula f_1 holds until formula f_2 is true ("**U**").

Model Checking Given a Kripke structure $\mathcal{K}$ and a temporal formula f, the problem consists of determining the set of all states that satisfy f, hence of the computation of

$$\{s \in S \mid \mathcal{K}, s \models f\}.$$

7.2.3 Translating

Fix a universe $\mathcal{U}$ and an interpretation for the ER model (technically, an interpretation for the map algebra is required). We may and do assume that we need only a finite subset of $\mathcal{U}$, and that we deal only with a finite number of instances to $\mathcal{M}$. These instances may arise through a finite sequence of insertions and deletions from the empty instance σ_0, in which all entities and all relations are empty. They form the state space Σ.
To be more specific: The state space is formed by all blocks

$$\sigma_k := \langle \mathsf{r}_{T+2\cdot k\cdot\Sigma+1}, \ldots, \mathsf{r}_{T+(2\cdot k+1)\cdot\Sigma} \rangle$$

($k \geq 0$) of map letters forming a valid ER model. Transitions are provided through the map letters in the block

$$\rho_k := \langle \mathsf{r}_{T+(2\cdot k+1)\cdot\Sigma+1}, \ldots, \mathsf{r}_{T+(2\cdot k+2)\cdot\Sigma} \rangle$$

which carry the changes, as described in Sections 5 and 6. There are apparently only finitely many possible choices for these transition vectors, when one identifies values coming from an infinite domain like the reals by suitable representatives.
The relation Γ is defined through: $\sigma\Gamma\sigma'$ iff there exists a transition vector ρ_k for $\sigma = \sigma_k$ which, when executed on σ_k, will result in $\sigma' = \sigma_{k+1}$. Apparently, Γ is a left-total relation.
We assume that each element u in the universe $\mathcal{U}$ comes with a finite set $e_u \subseteq AP_0$ of elementary properties. Let $\mathcal{H}_\sigma$ be the finite set of elements in $\mathcal{U}$ that take part in the construction of $\sigma \in \Sigma$, and put

$$\mathcal{H}_{\mathcal{M}} := \bigcup_{\sigma\in\Sigma} \mathcal{H}_\sigma$$

Call a partial map $q : \mathcal{H}_{\mathcal{M}} \to AP_0$ *admissible* iff $q(u) \in e_u$ holds for all $u \in \mathsf{dom}(q)$. This will define the atomic propositions: define for an admissible map q the proposition

$$q^{\#} := \bigwedge_{u\in\mathsf{dom}(q)} q(u)$$

and let

$$AP := \{q^{\#} | q \text{ is an admissible map}\}.$$

Finally, define $\Lambda : \Sigma \to 2^{AP}$ through

$$\Lambda(\sigma) := \{q^{\#} | \mathsf{dom}(q) \subseteq \mathcal{H}_\sigma\}$$

Apparently this denotes the set of all atomic propositions which are valid in instance σ.
This Kripke structure can then be used for checking properties along insertion- or deletion paths in the ER model.

References

[1] C. Brink, W. Kahl, and G. Schmidt, editors. *Relational Methods in Computer Science*. Advances in Computing. Springer-Verlag, Wien, New York, 1997.

[2] D. Cantone, E. G. Omodeo, and A. Policriti. *Set Theory for Computing*. Springer-Verlag, 2001.

[3] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. The MIT Press, Cambridge, MA, 2000.

[4] E. F. Codd. A relational model for large shared data banks. *Communications of the ACM*, 13(6):377–387, 1970.

[5] E.-E. Doberkat. Generating an algebraic specification from an ER-model. *International Journal of Software Engineering and Knowledge Engineering*, 7(4):525–552, 1997.

[6] M. Gogolla and U. Hohenstein. Towards a semantic view of an extended entity-relationship model. *ACM Transactions on Database Systems*, 16:369–416, 1991.

[7] R. Hettler. Zur Übersetzung von E/R-Schemata nach SPECTRUM. Technical Report TUM I-9333, Technical University, Munich, 1993.

[8] U. Hohenstein. *Formale Semantik eines erweiterten Entity-Relationship Modells*. B. G. Teubner, Stuttgart und Leipzig, 1993.

[9] A. Jaouna, N. Belkhiter, H. Ounalli, and T. Moukam. Databases. In *[1]*, pages 197 – 210. Springer-Verlag, 1997.

[10] B. Meyer. *Object-oriented Software Construction*. Prentice-Hall, Englewood Cliffs, NJ, 5 edition, 1998.

[11] E. G. Omodeo and E.-E. Doberkat. Algebraic semantics of ER-models in the context of the calculus of relations. Part I: Static view. *Electronic Notes in Theoretical Computer Science*, 2001.

[12] M. Page-Jones. *Fundamentals of Object-Orientied Design in UML*. Dorset House Publishing & Addison-Wesley, New York and Boston, 2000.

[13] G. Schmidt, C. Hattensperger, and M. Winter. Heterogeneous relation algebra. In *[1]*, pages 39 – 53. Springer-Verlag, 1997.

[14] A. Tarski and S. Givant. *A formalization of set theory without variables*, volume 41 of *Colloquium Publications*. American Mathematical Society, 1987.

[15] B. Thalheim. *Entity-Relationship Modeling: Foundations of Database Technology*. Springer-Verlag, 2000.