

The Third Manifesto Revisited

Maurice Gittens <maurice at gittens dot nl>

March 25, 2007

Abstract

According to the authors, Hugh Darwen and C.J. Date, of the third edition of the Third Manifesto entitled “Databases, types and the relational model; The Third Manifesto” the maxim: *All logical differences are big differences* and its corollary *All logical mistakes are big mistakes* has been central to their work on this book. Respecting the standard set by this maxim and its corollary, this paper will proceed to identify a number of issues with the logical consistency of the dissertation presented in The Third Manifesto. The conclusion is drawn that Date and Darwen seem to be regressing relative to Codd’s RM/T work.

The copyright of this document belongs to its author. Making complete and unmodified copies of this document is allowed for personal use.

Status: draft

Revision history

- 25 March 2007; Do my best to fix-up my use of the English language.
- 04 February 2007; minor cleanups
- 22 January 2007; Add a few sentences on D&D violating their own prescriptions
- Sept 29 2006; Minor cleanups
- Sept 13 2006; Add example of alternative semantics accommodating the *concept* of an unknown value which still rejects *instances* of null values.
- Sept 12 2006; Minor editing
- Sept 8 2006; Added section on nulls and relation valued attributes
- Sept 7 2006; Add section about “Treating operators as relations”
- Sept 6 2006; A first draft of this document

Contents

| | |
|---|----------|
| 1 No more Great Blunders | 2 |
| 2 Treating operators as relations without rigour | 2 |
| 3 No adherence to the principle of semantic compositionality | 3 |
| 4 No semantic integrity in the presence of relational assignment | 4 |
| 5 Undermining issues with relation valued attributes | 6 |
| 6 No sound substantiation for the rejection of unknown values | 7 |
| 7 Conclusions | 9 |

1 No more Great Blunders

Relative to the second edition of The Third Manifesto, the third edition has changed, on many points. This edition gives a better impression of the combined position of Date and Darwen on databases, types and the relational model than the previous version. There is no longer much ado about the so-called *great blunders* so prominently present in the second edition of the The Third Manifesto.

Still, there remain issues with the logical consistency of the dissertation presented in the third edition of The Third Manifesto. In my opinion, the standard set by the maxim: *All logical difference are big differences* and its corollary *All logical mistakes are big mistakes* that Date and Darwen present as guiding principle in their work, has not yet been met by The Third Manifesto. Key issues which have influenced my position are:

- Treating operators as relations without rigour
- No adherence to Frege's principle of semantic compositionality
- No semantic integrity in the presence relational assignment
- Undermining issues with relation valued attributes
- No sound substantiation for the rejection of unknown values

An elaboration on these issues is presented in the following.

2 Treating operators as relations without rigour

In appendix A of The Third Manifesto, Date and Darwen, repeatedly make the unqualified statement that operators can be treated as relations. This idea, which they claim as

their own¹, is used as grounds for dispensing with a few operators from Codd's algebra as can be read in the following quote taken from Appendix A.

We now claim that, given the fact that operators can be treated as relations, and given also the availability of the A operators AND, REMOVE, and RENAME (the latter two still to be discussed), it is indeed the case that we can *dispense* with restrict, EXTEND, and SUMMARIZE. We will justify this claim in the next section but one².

First, it seems fair to assume that many people familiar with, for example, truth tables as they are known in the context of boolean algebra, have for many years appreciated the fact that *commutative* operators can be treated as relations. So, Date and Darwen seem to have discovered hot water here. Second and more significantly, while making claims about operators in general, Date and Darwen attempted to motivate their dispensing of specific operators from Codd's algebra based on an example involving the *commutative* operator PLUS. Rigour would require them to show that pertinent operators from Codd's algebra are also commutative. Alternatively, rigour would require an elaboration on non-commutative operators such SUBTRACT as well. Attempting to view non-commutative operators as relations one soon encounters the problem that it is necessary not only to identify the operands of non-commutative operators but it is also necessary to designate their respective roles. Information about the roles of operands is lacking in the relational representation of *non-commutative* operators and Date and Darwen would need to illustrate how this information could be catered for without violating their proscriptions and prescriptions.

3 No adherence to the principle of semantic compositionality

It is a general principle of language design that the substitution of variables for their corresponding values should not change the meaning of expressions containing them. In the third edition of The Third Manifesto Date and Darwen are in violation of this principle. Consider that the type of a relvar is determined by the header of the relvar. The candidate keys associated with a relvar are *not* part of their type according to The Third Manifesto.³ This choice by Date and Darwen, represents a serious logical error because it causes variables and values of the *same* type to not be interchangeable. This is evident when one considers that a relation value C of type T may not be assignable to a relational variable V of type T . More specifically, the assignment of C to V is not allowed when there are candidate key constraints defined on V to which C is not in adherence. Put another way, even though V and C share the same type, the assignment $V = C$ may or may not be allowed depending on whether or not C is in adherence with all candidate key constraints defined for V .

¹See page 366

²See page 364

³See RM prescription 13 and 15 on page 85, point 13.

Similarly, nested relation *values* and non nested relation *values*, cannot play the role of *parent* in foreign-key relations using facilities provided by the The Third Manifesto. This is obviously true because, according to The Third Manifesto, relation *values* have no⁴ associated candidate keys and foreign-keys are defined in terms of candidate keys of the parent relation *variable*. Thus, The Third Manifesto', relvars cannot be in general be replaced by their values and are consequently not *referentially transparent*.

Finally, it can also be noted that Date and Darwen are in violation the principle of *conceptual integrity* and of their own RM prescription number 21⁵. This is obvious because the assignment of a value v to a variable V denoted $V := v$, where both V and v share the same type T , does not *in general* imply that the equality expression $V = v$ yields true. Again the reason for this inconsistency is that the type of relation variables does not include their associated candidate keys.

4 No semantic integrity in the presence of relational assignment

Breaches of semantic integrity invalidates trust in data stored, thus seriously compromises value. Consider the relvar predicate for a relation *Author* with one candidate key (based on surname) and two tuples.

| SURNAME | first name |
|---------|------------|
| Date | Chris |
| Darwen | Hugh |

The *relvar predicate* (see page 29) for this relation might read:

Author with surname "Date" has first name "Chris".
 Author with surname "Darwen" has first name "Hugh".

Consider the following multiple assignment statement where the key values for both tuples are swapped.

```
update Author where surname = "Date" { surname = "Darwen" },
update Author where surname = "Darwen" { surname = "Date" };
```

Since integrity checking is postponed until after both assignments have been performed⁶, this assignment statement is allowed by the The Third Manifesto. Our example relation now becomes:

| SURNAME | first name |
|---------|------------|
| Darwen | Chris |
| Date | Hugh |

Which reads:

⁴The TTM does assign implicit candidate keys to virtual relvars. But not to relation values.

⁵See page: 177

⁶See page 179.

Author with surname "Darwen" has first name "Chris".

Author with surname "Date" has first name "Hugh".

Given the relation values of the *Author* relvar *before* and *after* this single assignment and the knowledge that only one assignment has taken place, consider a forensic application which needs to find out what was *changed* by this assignment statement. Not appreciating this fundamental breach of integrity facilitated by The Third Manifesto will likely lead to seriously erroneous conclusions like:

The first name of the Author with surname "Date" was changed to "Hugh"

and

The first name of the Author with surname "Darwen" was changed to "Chris".

What, could be done to avoid more realistic⁷ variants of this example to cause compromise of the semantic integrity of relational databases adhering to The Third Manifesto? In essence the problem is the fact that assignment in the The Third Manifesto, by design, lacks facilities for keeping track of specific tuples.⁸ Because Date and Darwen, contrary to Codd, have chosen to reject the concept of *tuple identity*. In practical systems this requirement can be met by:

- using Codd's immutable, system assigned, surrogates to serve this purpose
- providing facilities to specify which key values are immutable in an assignment
- providing some other means of keeping track of individual tuples in assignments

How ever Date and Darwen choose to address this issue, the fact remains that it represents a deviation from the principle of conceptual integrity and a serious regression relative to Codd's RM/T because in the presence of a *single* assignment statement, the semantic integrity of a complete database can be compromised.

Consequences Violation of the semantic integrity, as facilitated by The Third Manifesto's lack of support for tuple identity, is likely to be among the most severe forms of compromise for an information system, because it compromises trust. Codd's immutable surrogates⁹ allow relation variables to be understood as sets of tuple variables with a common type, thus providing the ability to trace change to individual tuple variables. Consequently, Codd style databases facilitate business rules, accountability, traceability and forensics, among others, in ways not possible with Date and Darwen styled databases which provide no facilities that allow individual tuples to be reliably tracked or constrained over time.

⁷If you are still having doubts, please consider a variant of the above based around bank accounts. Imagine a bank with a Date and Darwen styled database. What would happen in a financial system handling hundreds of transaction per second, if it were to lose track of account numbers, by means similar to those presented above? Think about it.

⁸Transitional constraints as defined by The Third Manifesto do not help in the case under discussion

⁹For the purposes of semantic integrity and information security, in the presence of change, there is at least one requirement in addition to immutability.

Transition constraints The Third Manifesto offers transition constraints. As can be read on page 220 in the section elaborating on very strong RM suggestion 4, these transition constraints allow constraints to be placed on the values *relvars* can take, given their current value. In this section an example is presented by Date and Darwen which follows the following approach.

- Join the current and previous value of a relvar
- Pick out desired tuples from the resulting relation
- Process the result

The fallacy under discussion is based on the incorrect assumption by Date and Darwen that tuples can, in general, be “picked out”, in a semantically valid way, without support for tuple identity.

5 Undermining issues with relation valued attributes

Contrary to Codd, Date and Darwen allow relation valued attributes. Please consider the following questions.

- What problem is solved by support for these attributes that could not be solved otherwise?
- What propositions can be represented by relvars including relation valued attributes? Can these propositions not be represented, by other relational means?
- What positive traits of alternative solutions are not available to solutions based on relation valued attributes?

These questions are specifically pertinent in the light of the fact that these relation valued attributes are *relation values* and as such they lack associated candidate keys. In addition relation valued attributes cannot play any role in foreign-key constraints given the facilities provided by The Third Manifesto. Which is to say that alternatives to relation valued attributes have more facilities to accurately constrain databases to adhere to requirements of business- and other applications.¹⁰

So, the question remains, why should the relational model be complicated for dubious gain relative to Codd’s alternative? Appendix B of The Third Manifesto, represents an elaboration of sorts on this topic. This chapter digresses much but provides little that is of substance. I quote¹¹:

What then is the criterion for making something a type and not a relvar?
In our opinion this question is still somewhat open.

¹⁰For example, the GROUP statement does not provide facilities for specifying candidate key constraints and foreign key constraints

¹¹See page 379

Put another way Date and Darwen do not seem to know (!), in any definite sense, what the advantages of relation valued attributes are relative to alternative relational solutions. The disadvantages however, are clear:

the employment of relation valued attributes as introduced by The Third Manifesto provides less opportunity for the expression of candidate key and foreign key constraints in databases, relative to alternatives not involving relation valued attributes.

Is it not up to Date and Darwen to provide proper arguments for adding relation valued attributes to the relational model? Until logically valid advantages of relation valued attributes can be illustrated, relative to alternatives, support for such attributes, as defined by The Third Manifesto, seem not only a solution in search of a problem but also a needless and pointless complication. Consequently, given the current state of affairs on this issue, I maintain that support for relation valued attributes as defined by The Third Manifesto represents a violation of at least the parsimony requirement of RM prescription 26.

About relation valued attributes and relational operators Adding insult to injury, it can be noted that the *algebra A* appears to have no operators which allow relation valued attributes to be singled out for manipulation. This entails that from the perspective of The Third Manifesto's *algebra A*, relation valued attributes are *not* relation values at all. This is obviously true because these relation valued attributes cannot be manipulated by the set of relational operators specifically defined to accommodate the transformation of relation values. Consequently, from the perspective of the transformations facilitated by *A*, relation valued attributes seem a purpose unto themselves. So from the perspective of the algebra *A* questions like the following can be asked:

- *In what way would the algebra A be logically different if the GROUP operator produced an XML value?*
- *More generally, in what way is support for relation valued attributes logically different from, for example, builtin support for XML valued attributes?*

6 No sound substantiation for the rejection of unknown values

Nulls in databases are a source of much discussion. Codd, together with many practitioners allow nulls to be used where pragmatically deemed appropriate. Date and Darwen disagree. This edition of the Third Manifesto provides little substantiation for their position. This is a pity because it would have been nice to have a solid presentation of their arguments which amounts to more than picking on SQL. Date and Darwen seem to have rejected the concept of *the unknown* based on issues with nulls encountered in SQL. For *example*, since many agree that nulls are not values, it just might be possible that a language with consistent semantics can be designed that accommodates

the *concept* of the unknown in databases while, at the same time, the same language would have no notion of an *instance* of a null value. Such a language would accommodate Codd's concept of an attribute whose value is not known without mandating three valued logic. Consider for example the following proposition corresponding to an example relvar predicate taken from The Third Manifesto¹².

Supplier *S1* is under contract, is named *Smith*, has status *20*, and is located in city *London*.

Under the assumption that the city attribute of the corresponding relvar accommodates attributes whose value is unknown and that the city corresponding with supplier *S1* is unknown, here are two options for the corresponding proposition:

1. Supplier *S1* is under contract, is named *Smith*, has status *20*.
2. Supplier *S1* is under contract, is named *Smith*, has status *20*, and is located in an *unknown* city.

What would a language exploiting the first option look like? Cannot conceptually pure and consistent, language semantics be devised based on this the proposition?

Please consider the following Codd styled relation *R*¹³.

| SUPPLIERID | name | city (possibly null) |
|------------|------|----------------------|
| S1 | John | |
| S2 | Jane | Paris |
| S3 | Judy | |

As long as a language can *guarantee* that operators in the language would allow *no* relation values to be derived from our example relation *R*, which refer to the unknown, all would be well. Which in our example is to say that the statement:

```
select * from R
```

would produce an error. While the statements:

```
select SUPPLIERID, name from R
and
select * from R where city="Paris"
```

would execute without error because the resulting relation values would contain no tuples referencing the unknown. The *key* point to appreciate is that support for the *concept* of the unknown does not *require* support for the concept of a null value. Failing to appreciate this logical difference is a *big* mistake. In my opinion there is *absolutely* nothing wrong with intelligently supporting the *concept* of the unknown in databases. The Third Manifesto has provided no evidence that the logical difference between the

¹²See page 29

¹³Now, some might say that *R* is not a relation. My reply to them is that *R* is a relation according to Codd. Why would one assume that Codd had a faulty understanding of things relational?

concept of *the unknown* and the notion of an *unknown value* was well appreciated by Date and Darwen. Such an appreciation might have led to a more inclusive stance regarding support of Codd's notion of attributes whose value are not known. So, in the light of the above:

- *Is it a matter of fact that Date and Darwen have provided proper substantiation for the rejection of nulls?*
- *Is it a matter of fact that The Third Manifesto has provided evidence of properly researching the issue of nulls?*

I think not.

7 Conclusions

The relational model as presented by Date and Darwen in the third edition of The Third Manifesto represents a regression relative to Codd RM/T on the following grounds:

- “Treating operators as relations” approach lacks rigour.
- The Third Manifesto does not facilitate the semantic integrity of databases in the presence of a single assignment. This may facilitate different forms of information security breaches based on tuple identity theft.
- The Third Manifesto is inconsistent with regards to assignment of values and variables of the same type. Consequently, assignment as defined in the Third Manifesto is not referentially transparent, not in adherence with the Frege's principle of semantic composition and in violation of Date and Darwen's own RM prescription 21 and 26.
- Since, The Third Manifesto has provided no evidence of *added value* to be associated with relation valued attributes, as introduced by The Third Manifesto, and the existence of significant disadvantages relative to alternatives, support for relation valued attributes as defined by The Third Manifesto seems a service to none. This represents a violation of RM prescription 26.
- Nulls rejected without evidence of either having researched the topic well or having considers alternatives¹⁴.

Codd's work does not suffer these deficiencies.

References

- [1] L,T,F Gamut [1991] Logic, Language and Meaning Volume 2, University of Chicago Press.

¹⁴See RM Proscription 4 on page 193

- [2] E.F. Codd, [1979] “Extending the Database Relational Model to Capture more Meaning”
- [3] C.J. Date, Hugh Darwen [2007] Databases, Types and the Relational model, Addison-Wesley Publishing Company.
- [4] C.J. Date, Hugh Darwen [2000] Foundation for Future Database Systems, Addison-Wesley Publishing Company.
- [5] Maurice Gittens [2003], “A critical reading of the third manifesto”, <http://www.gittens.nl>