

# Microsoft's new database modeling tool: Part 3

Terry Halpin  
Microsoft Corporation

**Abstract:** This is the third in a series of articles introducing the Visio-based database modeling component of Microsoft Visual Studio .NET Enterprise Architect. Part 1 discussed how to create a basic ORM source model, map it to a logical database model, and generate a DDL script for the physical database schema. Part 2 discussed how to use the verbalizer, mark an object type as independent, objectify an association, and add some other ORM constraints to an ORM source model. Part 3 shows how to add set-comparison constraints (subset, equality and exclusion) and how exclusive-or constraints are obtained by combining exclusion and disjunctive mandatory constraints.

## Introduction

This is the third in a series of articles introducing the database modeling solution in Microsoft Visio for Enterprise Architects (VEA), which is included in the Enterprise Architect edition of Visual Studio .NET. This article discusses how to add set-comparison constraints to an ORM source model, and how to form exclusive-or constraints by combining exclusion and inclusive-or constraints. Familiarity with ORM and relational database modeling is assumed. Overviews of ORM are downloadable [1, 2], and a thorough treatment of ORM and database modeling is discussed in my latest book [3]. For previous articles in this series, see [4] and [5].

## Set-comparison constraints

If two roles are played by the same object type, or their object types share a common supertype, they are said to be *compatible*, and it is meaningful to compare their populations. The same is true for role-sequences (ordered lists of roles). For databases, only three set-comparison operators are relevant: subset ( $\subseteq$ ), equality ( $=$ ) and mutual exclusion ( $\otimes$ ).

A *subset constraint* from a source role sequence to a target role sequence asserts that the population (set of instances) of the source role sequence must always be a subset of the population of the target role sequence. The constraint is displayed graphically as a circled " $\subseteq$ " connected by a dotted arrow running from the source to the target (e.g. see Figure 1). In previous versions of our ORM tool, the dotted arrow by itself designated the subset constraint. In VEA, the circled subset symbol is always shown explicitly. This clarifies the meaning of the constraint notation, supports rare cases where source and target belong to the same association, and allows use of the dotted arrow within other directed constraints (e.g.  $\leq$ ) for which a graphical constraint notation might later be added.

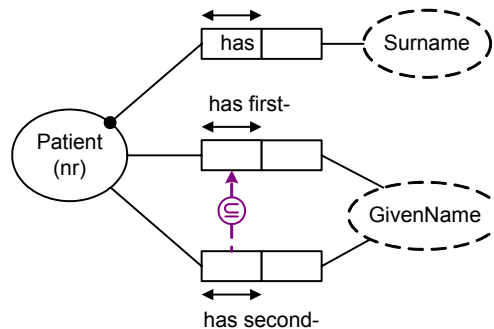
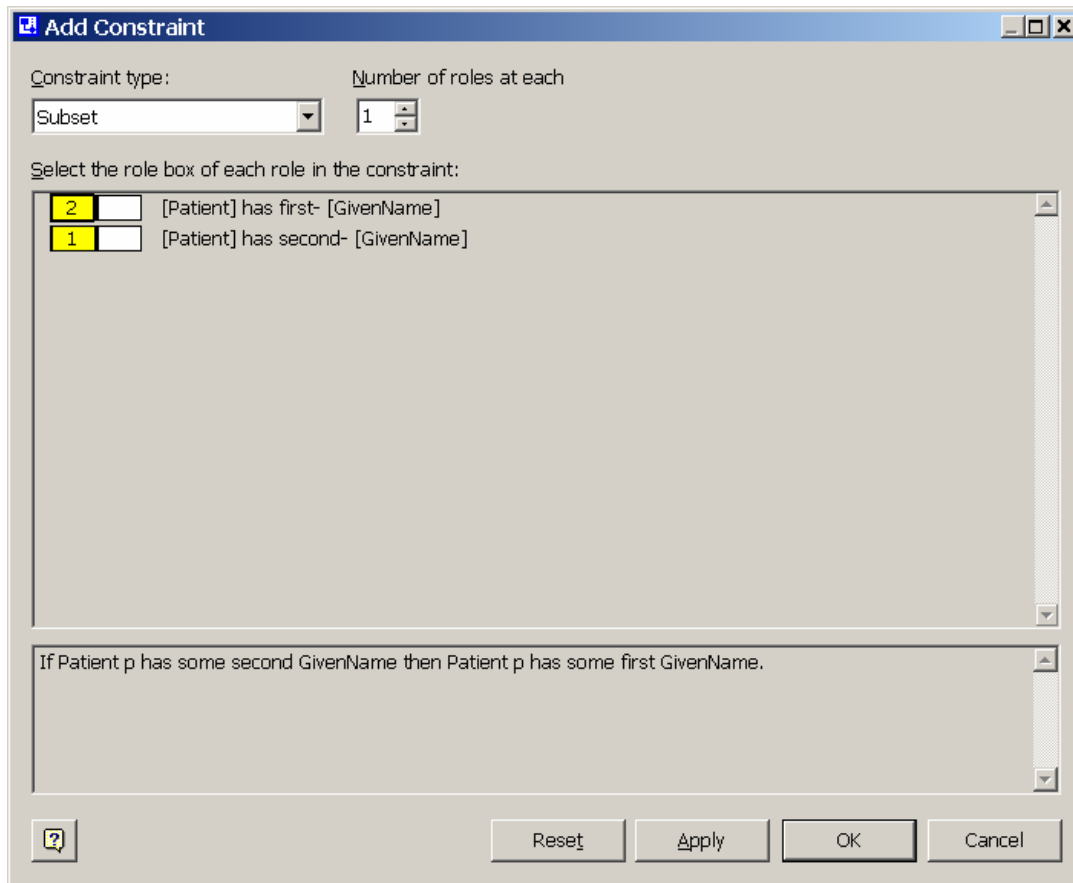


Figure 1 A subset constraint between single roles

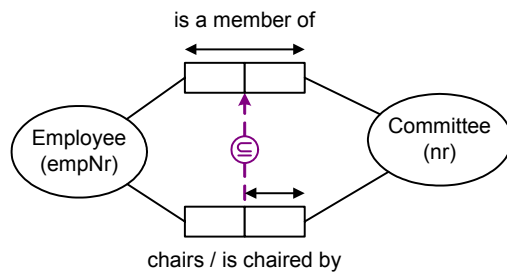
The simplest case of a role sequence is just a single role. In Figure 1 the subset constraint between two single roles means that the set of patients with a second given name must be a subset of the set of patients with a first given name. In other words, if a patient has a second given name, then he/she must also have a first given name. To add this subset constraint in VEA, first enter the two fact types (e.g. by adding them in the business rules editor and then dragging them onto the drawing window). Now hold the Shift key down as you select the relevant predicates, right-click, and choose Add Constraints from the right-click menu. When the Add Constraint dialog appears, select Subset in the Constraint type field, and then select the source role for the constraint followed by the target role. As you select role boxes in the dialog, they are numbered 1, 2 etc. in the order of selection. If you do this correctly, the dialog box should now appear as in Figure 2. The constraint is automatically verbalized in the lower section of the dialog box. If this is the only constraint you wish to apply right now to those fact types, hit OK. If you want to add other constraints now to one or more of these fact types, hit Apply and then add the other constraints.



**Figure 2** Choose the subset constraint type, then select the source role [1] then the target role [2]

Note the use of the *hyphen* in the predicates “has first-” and “has second-”. This binds the adjectives “first” and “second” to GivenName when constraints on those predicates are verbalized, so that keywords like “some” are inserted before the adjective instead of after it. For example, if you omitted the hyphens in our current example, the verbalization would instead read “If Patient p has second some GivenName then Patient p has first some GivenName”, which is at best awkward and at worst unintelligible.

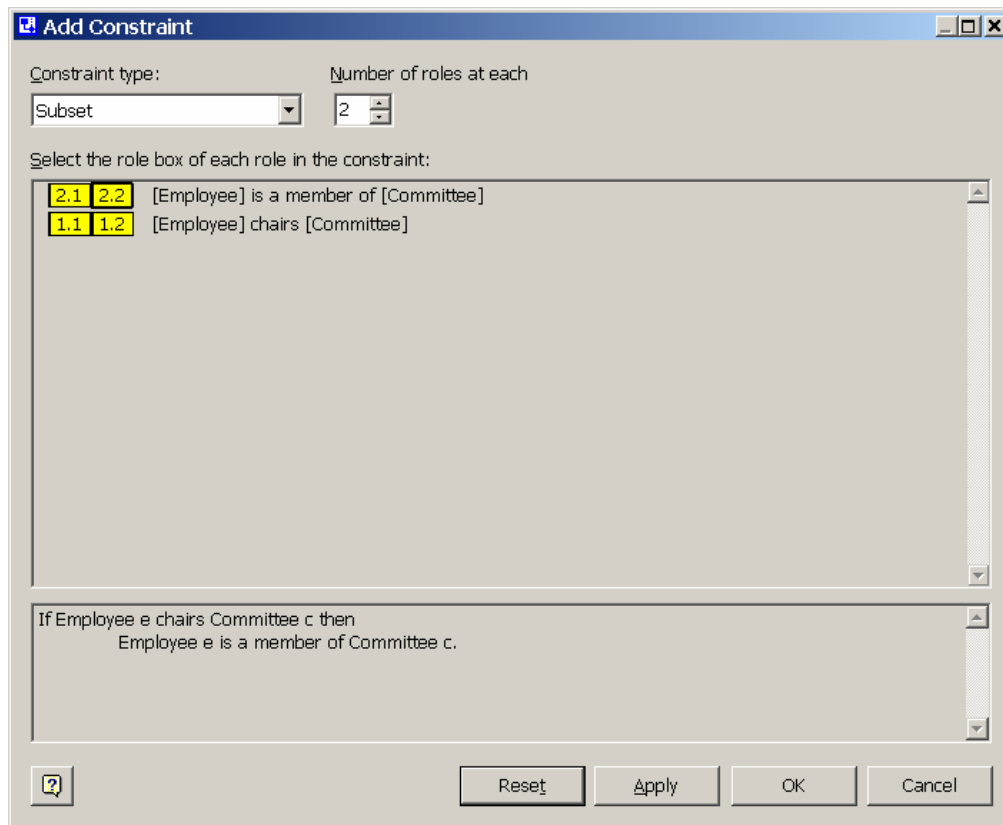
Figure 3 illustrates a subset constraint between role-pairs (each role-sequence contains two roles). If, as here, the roles in a pair are contiguous, the constraint connects to the junction of the roles. This constraint means that the population of Employee-Committee pairs instantiating the chairperson association must be a subset of the population of the membership association. That is, each person who chairs a committee must be a member of that very same committee.



**Figure 3** A subset constraint between role-pairs

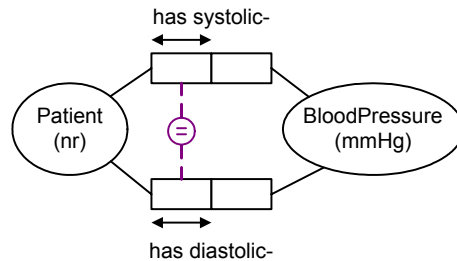
To add this subset constraint, add the two fact types to the diagram window, hold the Shift key down as you select the predicates, right-click, and choose Add Constraints from the right-click menu. When the Add Constraint dialog appears, select Subset in the Constraint type field. Notice that the lower section now prompts: If there is more than one role at each end of the constraint, increase the indicated "Number of roles at each end" to show this. By default, the number of roles at each end of the constraint is set to 1. Since there are two roles at each end of this constraint, change this setting to 2 (as in Figure 4). The caption "Number of roles at each" should actually read "Number of roles at each end".

Now select the source role pair, and then the target role pair, ordering the roles within each pair to match their corresponding role from the other pair. As you select role boxes in the dialog, they are numbered 1.1, 1.2, 2.1, 2.2 in the order of selection. The first part of each number denotes the role sequence, and the second part denotes the position within that sequence. If you do this correctly, the dialog box should now appear as in Figure 4. The constraint is automatically verbalized in the lower section of the dialog box. Hit OK to have the constraint accepted and added to the diagram.



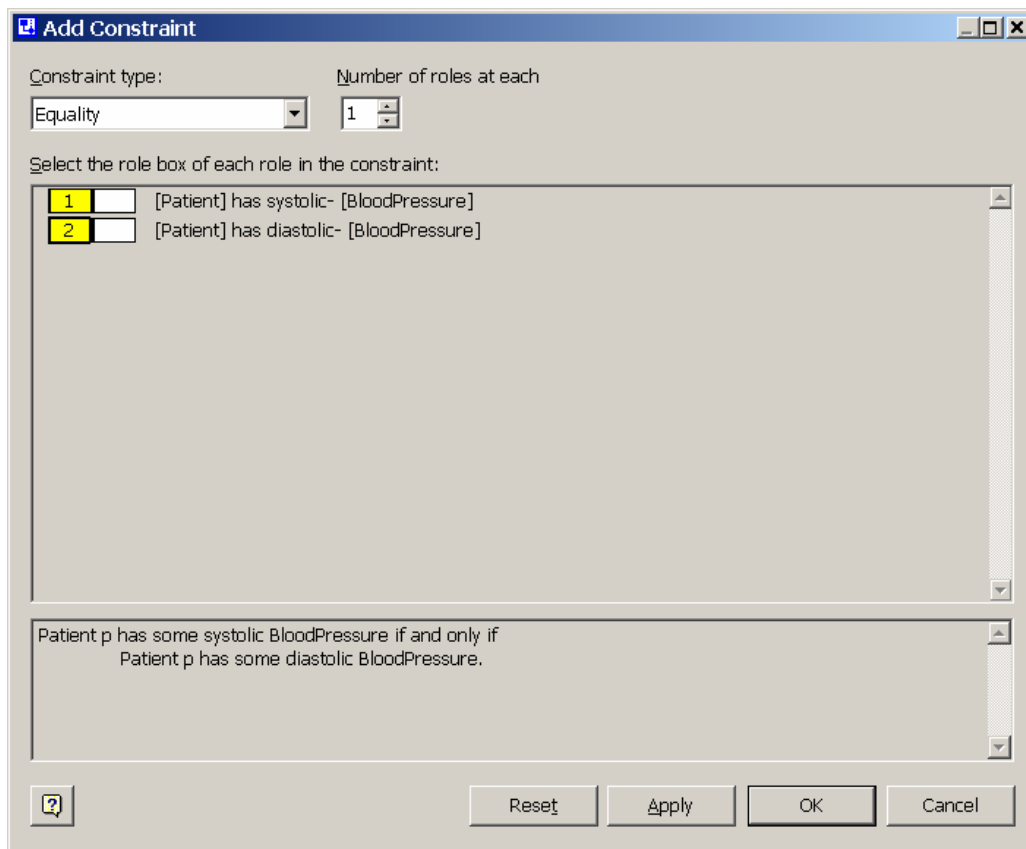
**Figure 4** Adding a subset constraint between role-pairs

An *equality constraint* between role sequences indicates their populations must always be equal. This is depicted as a circled “=” connected by a dotted line to the role sequences. Our earlier ORM tool instead used a dotted line with an arrowhead at each end, which reflected the equivalence between an equality constraint and a pair of subset constraints in either direction. This old notation is no longer supported.



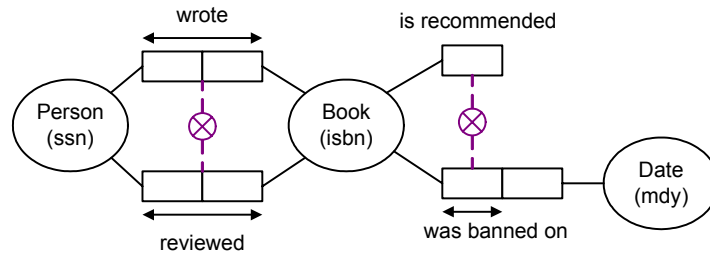
**Figure 5** An equality constraint between two single roles

To add this equality constraint, add the two fact types to the diagram window, hold the Shift key down as you select both predicates, right-click, and choose Add Constraints from the right-click menu. When the Add Constraint dialog appears, select Equality in the Constraint type field, and then select the role sequences (in this example, each sequence has only one role). If you do this correctly, the dialog box should appear as in Figure 6. Actually, the order of the role-sequences in an equality constraint doesn't matter, since equality is symmetric (unlike subset). The constraint is verbalized in the lower section of the dialog box. Equality constraints between longer role sequences may be added in a similar way as for subset constraints.



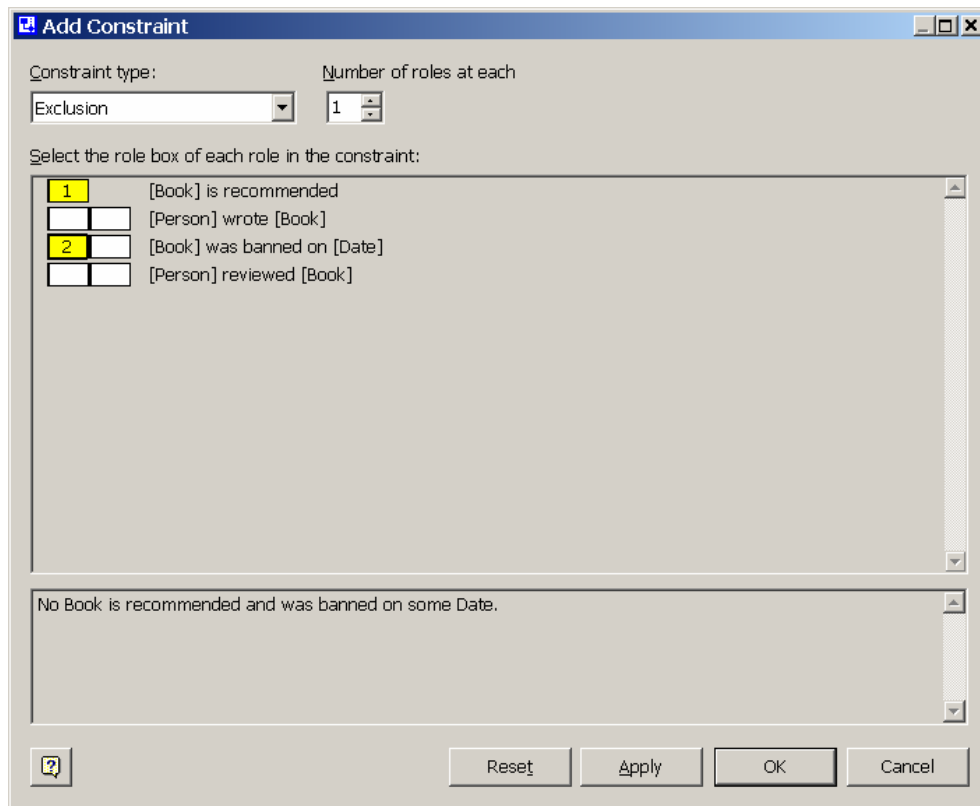
**Figure 6** Adding a simple equality constraint between roles

An *exclusion constraint* between role sequences indicates their populations must always be disjoint (mutually exclusive). This is depicted as a circled “X” connected by a dotted line to the role sequences. Figure 7 includes a pair-exclusion constraint (no person wrote and reviewed and the same book) and a simple exclusion constraint (no book can be both recommended and banned).



**Figure 7** A pair-exclusion constraint and a simple exclusion constraint

To quickly add these exclusion constraints, add the four fact types to the diagram window, hold the Shift key down as you select all four predicates, right-click, and choose Add Constraints from the right-click menu. When the Add Constraint dialog appears, select Exclusion in the Constraint type field, and then select the recommended and banned roles (as in Figure 8). Now hit Apply to have the constraint accepted and displayed, while leaving the dialog box open, ready to add the other exclusion constraint.



**Figure 8** Adding a simple exclusion constraint

Now select Exclusion in the Constraint type field, increment the number of roles at each end to 2, and then select the role pairs in the writing and review fact types (as in Figure 9). Now hit OK to have the constraint accepted and exit the dialog.

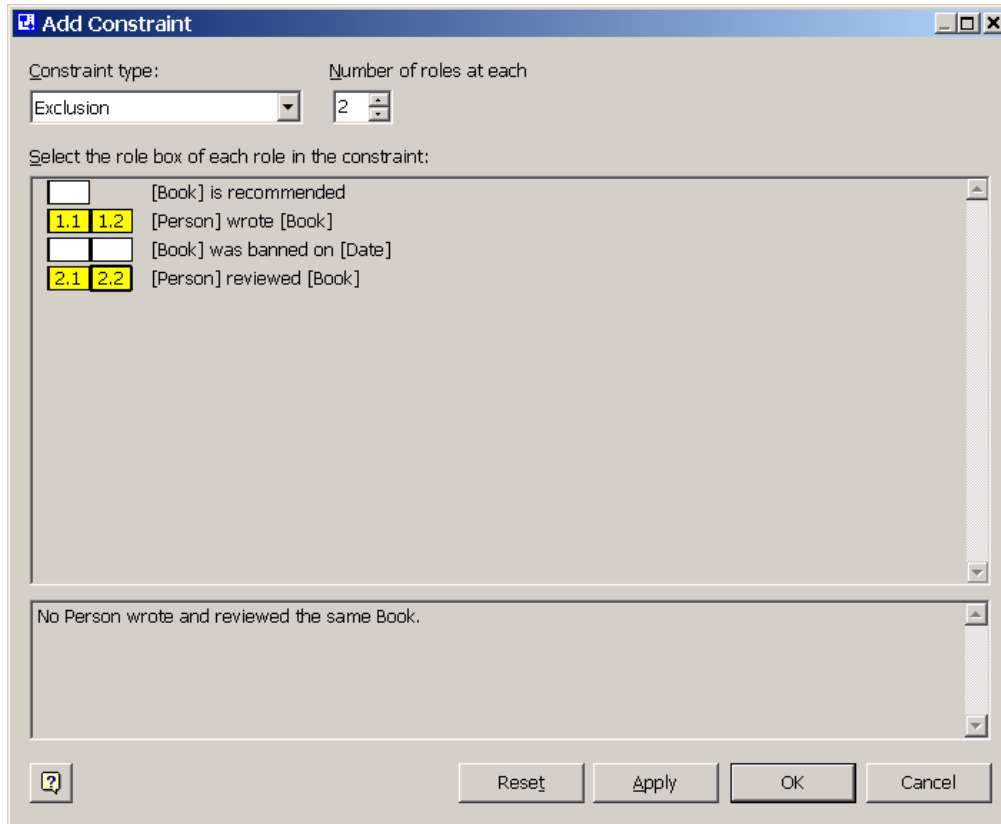


Figure 9 Adding a pair-exclusion constraint

### Exclusive-or constraints

In ORM, an *exclusive-or* constraint is simply an orthogonal combination of an inclusive-or (disjunctive mandatory) constraint and an exclusion constraint. By default, these two constraints are overlaid as shown on the left of Figure 10, where superimposing a circled dot (disjunctive mandatory) and circled “X” (exclusion) results in a lifebuoy symbol. To create this example, add the two fact types to the diagram window, select them, right click to bring up the Add Constraint dialog, and add first one constraint and then the other to the same roles (hit Apply after adding the first constraint, and OK after the second constraint).

If you wish to visually separate the two constraints making up the inclusive-or constraint, right-click on the lifebuoy symbol and chose the option Split X/OR constraint. The constraints will be now displayed separately as on the right of Figure 10. You can now work with either constraint individually (e.g. you could delete just one of them by selecting it and hitting Delete). If the two constraints appear separately you can merge them into the lifebuoy symbol, by selecting one of the constraints and dragging it over the other.

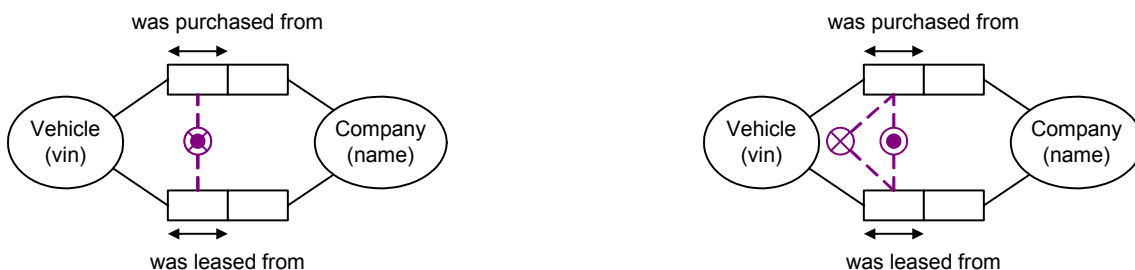


Figure 10 Exclusive-or is inclusive-or plus exclusion

## Conclusion

Apart from subtyping, you now have enough information to recreate the sample Employee ORM source model that ships with the product. The next two articles will discuss subtyping as well as other ORM constraints (frequency, ring and index constraints) and related issues such as constraint layers. If you have any constructive feedback on this article, please e-mail me at: [TerryHa@microsoft.com](mailto:TerryHa@microsoft.com).

## References

1. Halpin, T. A. 1998a, 'Object Role Modeling: an overview', white paper, (online at [www.orm.net](http://www.orm.net)).
2. Halpin, T.A. 1998b, 'Object-Role Modeling (ORM/NIAM)', *Handbook on Architectures of Information Systems*, Bernus, P., Mertins, K. & Schmidt (eds), Springer, Heidelberg, Ch. 4. (online at [www.orm.net](http://www.orm.net)).
3. Halpin, T.A. 2001a, *Information Modeling and relational Databases*, Morgan Kaufmann Publishers, San Francisco ([www.mkp.com/books\\_catalog/catalog.asp?ISBN=1-55860-672-6](http://www.mkp.com/books_catalog/catalog.asp?ISBN=1-55860-672-6)).
4. Halpin, T.A. 2001b, 'Microsoft's new database modeling tool: Part 1', *Journal of Conceptual Modeling*, June 2001 issue (online at [www.InConcept.com](http://www.InConcept.com) and [www.orm.net](http://www.orm.net)).
5. Halpin, T.A. 2001c, 'Microsoft's new database modeling tool: Part 2', *Journal of Conceptual Modeling*, August 2001 issue, (online at [www.InConcept.com](http://www.InConcept.com) and [www.orm.net](http://www.orm.net)).