# Reference Relationships

John Cartmell

November 27, 2014

## 1 Reference Relationships

Reference relationships are a second class of relationships and are distinguished from composition relationships by being depicted connecting to the side edges of the boxes representing entity types rather than to the upper and lower edges. Without distinction, connection to the left or right hand side of the boxes can be made; the side can be chosen for the convenience for the drawing. A text label at one end of a relationship line serves to describe the relationship from the point of view of the subject entity at that end. For example, we can say a chicken *lays* an egg or that an egg *is laid by* a chicken. Though there is a single relationship between chicken and egg there are two ways of articulating it and accordingly the relationship is shown labelled at each end:
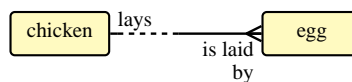
For convenience of the diagramming we can, without changing the meaning, turn this about so:
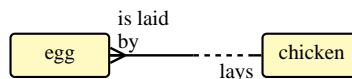

A chicken hatches from an egg too and we can add this to the diagram like so:
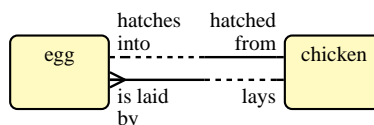

Of course, what we see here represented on the diagram are types of things not actual things - so the diagram does *not* express that a chicken lays the very same egg that it hatches from: rather that a chicken lays things *of the type* egg and that it is hatched from a thing *of the same type*.

The notation introduced for composition relationships has been descriptive of properties of relationships including cardinality and optionality (the crows foot and dashing of lines), exclusion (the exclusion arc) and use of abstraction (nested boxes). These notations apply equally
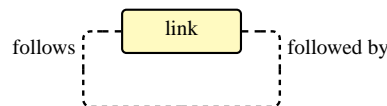
to the representation of reference relationships. A reference relationship can be looped around to relate instances of a single type and it is then said to be recursive. If it is many-one then a hierarchical[1] system of entities is implied such as a command hierarchy:
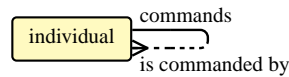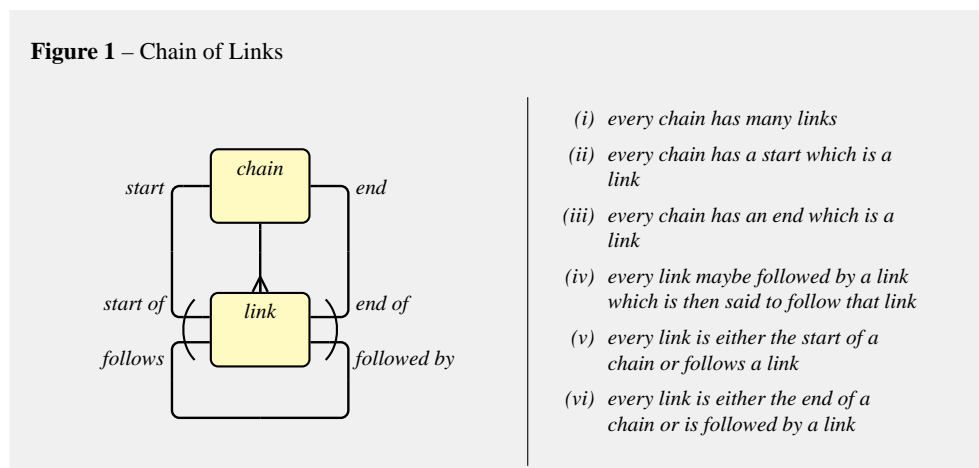

A reference relationship can be looped around to enter the opposite edge with no difference in meaning so the following is equivalent:
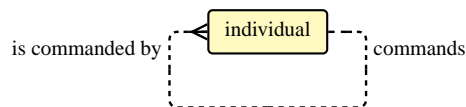

Consider also a chain composed of links each connected to the next. Each link of a chain is followed by the next and each link follows at most one other which is to say there is a recursive relationship:

For a more complete model see figure 1.

Figure 1 – Chain of Links

(i) every chain has many links

(ii) every chain has a start which is a link

(iii) every chain has an end which is a link

(iv) every link maybe followed by a link which is then said to follow that link

(v) every link is either the start of a chain or follows a link

(vi) every link is either the end of a chain or is followed by a link
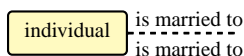
Some recursive relationships cannot be given a different role at either end for they are symmetric. For example x1 is married to x2 precisely if x2 is married to x1. Marriage is a symmetric binary relationship. When this relationship is shown in an entity model then it must be given the same the name at either end as so:
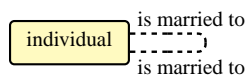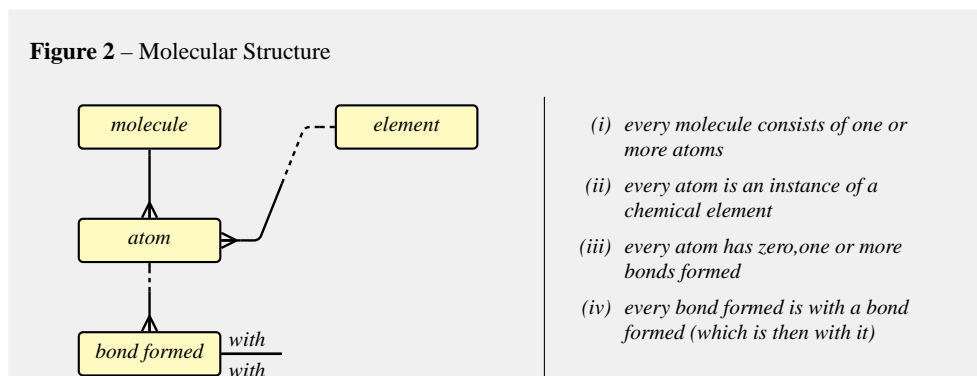

---
[1] we use the term somewhat loosely because there is nothing in the notation which implies an absence of cycles

As a shorthand we can compress this to a single line:

The example in figure 2 illustrates this notation in use.

Figure 2 – Molecular Structure

(i)   every molecule consists of one or more atoms

(ii) every atom is an instance of a chemical element

(iii) every atom has zero,one or more bonds formed

(iv) every bond formed is with a bond formed (which is then with it)

2   The Distinction between Composition and Reference

The entity modelling notation in one form or another is part of the core syllabus in the information sciences but invariably the distinction made here between composition and reference is hardly made[2]. It is surprising that the distinction between composition and reference is not taught more frequently because one of the prime motivations for teaching the entity relationship model is as a precursor technique to database design and, as we shall see later, concepts of context, reference and scope are of prime importance to database design and without them there remains a fudged step of the 'here is one I prepared earlier' variety. The concept of relationship scope is the key missing concept and it is introduced in this chapter.

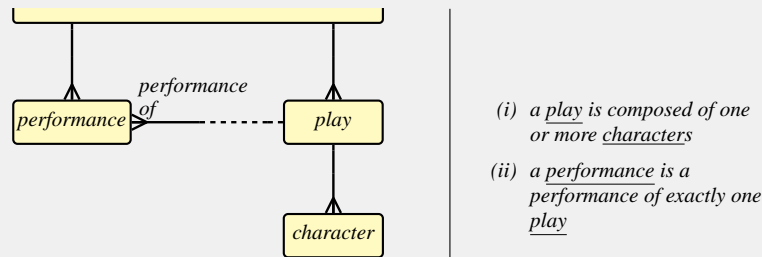Of these two superficially similar types of relationship:

- the relationship between a play and the characters within the play,

- the relationship between a play and performances of that play.

The first of these would generally be classified as a composition relationship for we can say that a play is in part composed of all the characters within it, whereas the second would generally be classified as a reference relationship for we would not say that a play is in part composed of all of it's performances. For this reason an entity model describing just these three entity types, 'play', 'performance' and 'character', contains both vertical composition relationships and an orthogonal reference relationship, as shown below in figure 3.

Since it is a distinction rarely made many readers may be sceptical of whether there is a credible distinction between composition and reference; in the circumstances such doubts are reasonable and much of this chapter will be devoted to examples and implications of the distinction. So far much weight has rested on appeal to a sense of what constitutes a part and of what parts something can reasonably be said to be composed. We said in the introduction

---

[2]The one exception to this would be in teaching of the UML notation wherein there is a further classification of composition relationships resulting in three subclasses of the core relationship concept rather than two as here

Figure 3 – Composition and reference

(i) a <u>play</u> is composed of one or more <u>characters</u>

(ii) a <u>performance</u> is a performance of exactly one <u>play</u>

that entity modelling was concerned with what could be known of an entity; now, another way of asking what can be known of an entity is to ask what description can be given of an entity or what of an entity can be communicated.

If focusing on 'parts' and 'composition' doesn't clarify the distinction between composition and reference or, for that matter, to convince of the credibility of the distinction, then another ways of clarifying relies on a focus on 'full description' or 'communication' and this in turns leads to the idea of copying the full description of an entity - for to communicate an entity is to copy it in some way from source to destination.
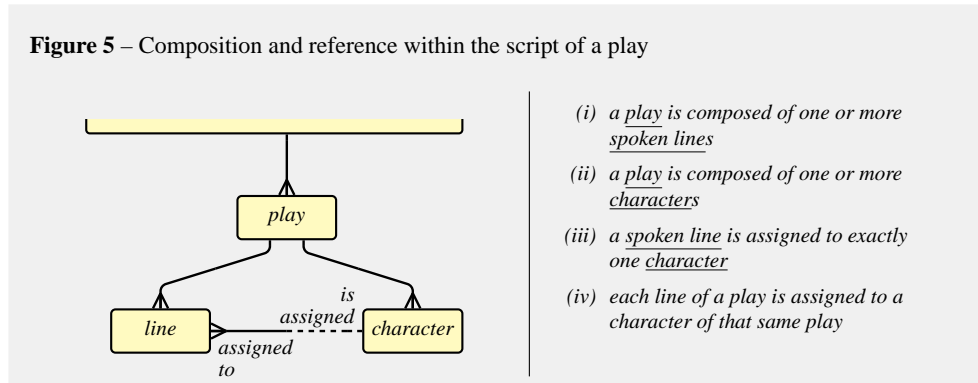
Therefore we ask what would be communicated in a full description of a play - surely it would include a full description of each of the characters? The play-characters relationship therefore passes the full description test and is classified as a composition relationship. The play-performances relationship on the other hand fails this test - it is not necessary to describe every performance of a play in order to fully describe the play - and therefore it fails the test. The matter will not rest however - there are many relationships which can be modelled either way and then models containing them are subtly different and are appropriate in different circumstances.

Figure 4 – The 'folder' example is an excellent example of a composition relationship. I cannot delete a folder on my computer without deleting all the folders and files contained within it (of course I can move the contained items first and then delete the parent folder). Shortcuts are different - I can delete a shortcut to a file or folder without deleting the file or folder. Therefore the relationship between a shortcut and that which it is a short cut to is a reference relationship.
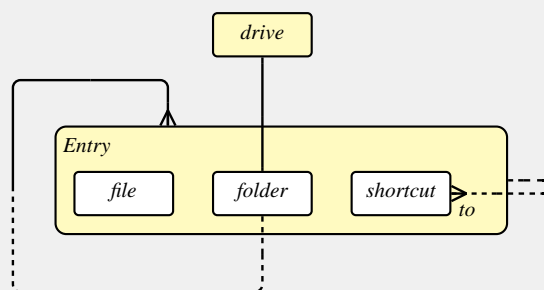
3 The Scope Concept

The example in figure **??** models another aspect of play; it shows two composition relationships (which have been left unlabelled) and one reference relationship doubly labelled.

Figure 5 – Composition and reference within the script of a play

(i) *a play is composed of one or more spoken lines*

(ii) *a play is composed of one or more characters*

(iii) *a spoken line is assigned to exactly one character*

*(iv)* *each line of a play is assigned to a character of that same play*

In terms of our understanding of a domain of discourse it is significant that reference relationships may be limited in scope - the instances of the reference relationship shown in figure 5 are local to the context of individual plays: as part of the structure of a play, a line is never assigned to a character of a different play - we might summarise the situation by saying that assignment is intra-play not inter-play. We have chosen to express this *fact of limited scope* in bullet (iv) of figure 5 and it is significant that the diagram *alone* does not express this fact[3]. Most significantly, the Entity Relationship diagram notation is able to express the types and cardinalities of reference relationships and how they are articulated but not their scopes. Nonetheless every reference relationship has a scope, the scope is fundamentally important, it can be expressed in words, or, as we see later, in equations, or in a subordinate diagram called a scope diagram.

Contrast the situation of figure 5 with that of figure 6. Figure 6 is similar in shape but the difference is that the reference relationship shown is *not* limited in scope. Rather the point of the relationship *translation* is to cross languages - it establishes an inter-language relationship rather than intra-language relationship. This again shows that the scope of a relationship - the extent to which it is global or local - 'inter' or 'intra' - cannot be deduced from the entity relationship diagram. Other means of expression must be used. Gaining an understanding of scope and the means of its expression is an important part of learning entity modelling.

A further example is given in figure 7 in which there is an example of a reference relationship between types at different levels in a composition hierarchy.

---

[3]Bullet (iv) of figure 5 adds to the information given in the diagram and is different in this respect to bullets (i),(ii),(iii) for these simply restate what is already asserted by the diagram

Figure 6 – Translations of a book
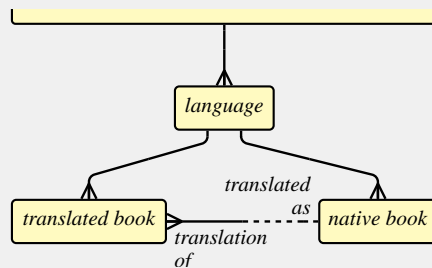
(i) *a language has one or more native books*

(ii) *a language has one or more translated books*

(iii) *a translated book is a translation of exactly one native book*

(iv) *a native book is translated as zero, one or more translated books*

Figure 7 – Recipes and their ingredients. In this figure bullets (i) to (iv) restate what is said in the diagram. Bullet (v) adds to the diagram - it is a scope statement for relationship 'use of'.
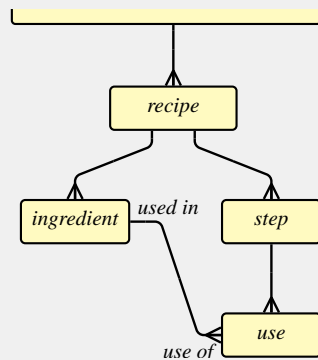
(i) *a recipe has one or more ingredients*

(ii) *a recipe has one or more steps*

(iii) *a step has one or more full or partial uses of zero, one or more ingredients*

(iv) *a thing used makes reference to zero or one ingredients*

(v) *if a thing used in a step of a recipe makes reference to an ingredient then the ingredient is an ingredient within the very same recipe*
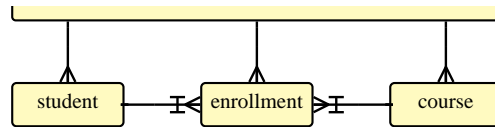
4 Many-Many Relationships and Their Avoidance

A many-many relationship is one depicted with a crows foot at each end whereas a many-one relationship is one with a crows foot at one end but not the other. An example many-many would be:
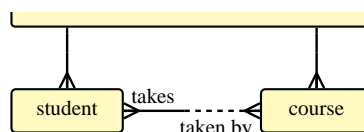
An alternative to using a many-many relationship is to use two many-one relationships and an entity type. The result is shown in figure 8. Any entity type introduced to eliminate a many-many relationship in this way is called an *intersection entity*.

6

Figure 8 – Students and courses in many-many relationship represented using an intersection entity 'enrollment'.


The avoidance or elimination of many-many-relationships from models is important to database designers who wish to represent or communicate the facts of a situation in tabular form. After elimination of many-many relationships the tables that are required are just one table per entity type in the model. This topic is covered in chapter 5.

Instead of using a pair of reference relationships in the elimination of a many-many, one composition and one reference can be used instead as illustrated in figure 9.

Figure 9 – Eliminating of a many-many relationship using composition.

Elimination by Party Within Agreement - This is the model that feels most appropriate conceptually that 'parties' are parts of agreements.
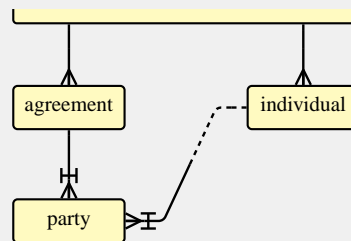

When a many-many recursive relationship is eliminated using a pair of reference relationships then the resulting model has the shape shown in figure 10.

Figure 10 – The shape after elimination of a many-many recursive relationship.
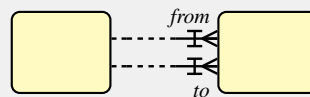
In regard to any model shaped as in figure 10, we can think of entities of the left hand type as being points and those of the right hand type as connections between these points; in this way we can imagine the instances of a many-many relationship as forming a network. For example because social interactions between people are many to many we can, and do, speak of social networks and in like manner we speak of business networks, supply networks, computer networks and so on; these are all usages based on many-many relationships between various types of entities as for instance this one:
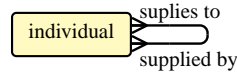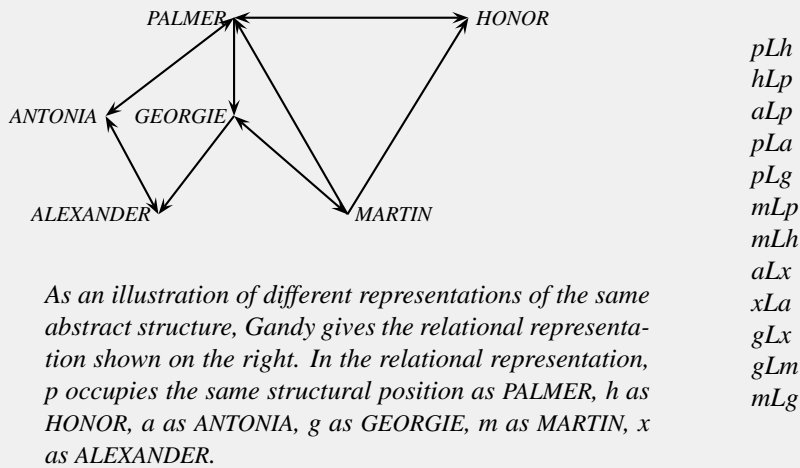
Of course we don't need entity modelling to enable us to think of networks of this, that or the other; entity modelling simply gives shape (that of figure 10) to this practice - as we say elsewhere entity modelling gives shape to metaphor. In the next section we discuss the network concept and variations of it in the forms that they have been given by mathematicians. At the same time we look at what would be meant by replacing either or both of the two reference relationships in figure 10 by composition relationships.

5 Graphs

5.1 Directed Graphs continued

There is a reliance here on the use of the work 'abstract' and on an understanding of what the mathematician means by this. The mathematical idea of an abstract structure is that of a set of individuals and relations independent of any particular representation of the individuals. In the case of a graph the individuals are vertices and edges and the relations are the incidence relations between them. With regard to any particular abstract structure, to describe it we need *some* kind of representation. Mathematican Robin Gandy describes this to a non-mathematical audience in lecture 'Structure in Mathematics'[4]; he draws the love relationships to be found in Iris Murdoch's novel Severed Head as a directed graph and I have drawn an equivalent graph in figure 11. If we abstract a directed graph from this then there are no labels, no lines, no points on paper just the facts of a set of individuals points and relationships[5].

Figure 11 – Example from 'Structure in Mathematics' of a directed graph showing the love relationships to be found in Iris Murdoch's novel 'Severed Head'. The labels on the graph are simply to show the correspondence with the novel.


pLh
hLp
aLp
pLa
pLg
mLp
mLh
aLx
xLa
gLx
gLm
mLg

As an illustration of different representations of the same abstract structure, Gandy gives the relational representation shown on the right. In the relational representation, p occupies the same structural position as PALMER, h as HONOR, a as ANTONIA, g as GEORGIE, m as MARTIN, x as ALEXANDER.

[4]In *Structuralism, edited by D. Robey, Wolfson College Lectures, 1972*.

[5]According to Gandy: This indifference to form of *representation* is what Bertrand Russell referred to when he said that in mathematics we are not interested in what we are talking about. We return to the subject abstraction versus representation in Chapter xxx.
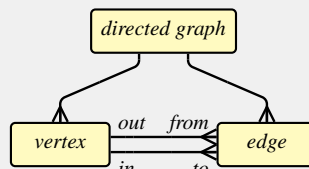
Gandy's depicted love relationship is a many-many relationship and we can represent in an entity model like this:
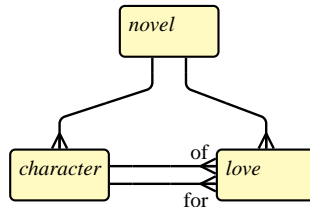
Equivalently in the technical vocabulary but with the same shape we have the entity model of a directed graph shown in figure 12


Figure 12 – The model of a directed graph.

There is of course nothing special about this example - 'loves' is a recursive many-many relationship and any such could be used to illustrate that a recursive many-many relationship is structurally a directed graph.

Gandy's relational representation shown in figure 11 could be rearranged in two ways- from the point of view of the lover or the loved; these are shown in the left and right hand sides of figure 13.

Figure 13 – Different versions of the relational representation of figure 11. From the left hand representation you may quickly find who is loved by *a*, who is loved by *g* and so on, from the representation on the right you may quickly determine who loves *a*, who loves *g* and so on.
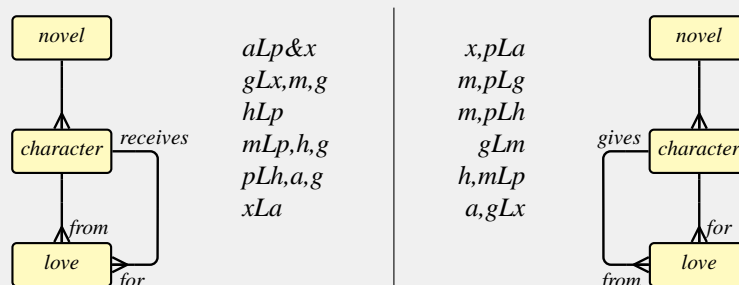

5.2   ER Models of Directed Graph

Viewing figure 13 and following on from figure 12 we see that there are two further ways of modelling directed graphs. These are shown in figure 14.

**Figure 14** – Further two ways of modelling directed graphs.
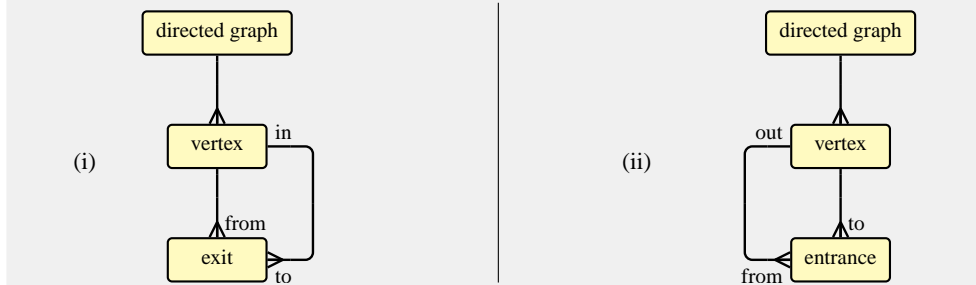
Though we have given three ways of modelling directed graphs note that mathematically there is only a single concept. The different models represent different ways of localising and communicating the information content of a graph. The models in figure 14 vary the incidence relationships between edges and vertices between categories *reference* and *composition*. If both are made composition then we get a further variation:
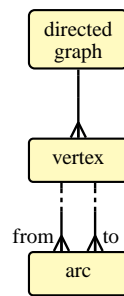
But what communication structure do we associate with this? Well, this way of modelling directed graphs is akin to the matrix structure modelled in chapter one. Applied to the representation of the Severed Head love relationships it implies a communication in which it is apparent both (i) who loves each subject *a,g,h*, etc and (ii) who each subject *a,g,h*, etc is loved by. This is made apparent by a matrix representation:

|	a	g	h	m	p	x
a					×	×
g		×		×	×	
h				×	×	
m		×				
p	×			×	×	
x	×	×				

We mentioned in chapter one that information is usually hierarchically communicated whereas with double composition relationships there is not one hierarchy but two hierarchies. Information must be double communicated once in each hierarchical form. For linear, hierarchical communication the above model is therefore replaced by one in which each of the two hierarchies is represented:
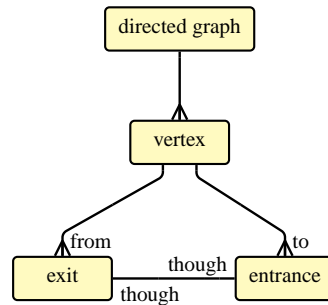
5.3 Modelling Graphs and Symmetric Relationships

We use the unqualified term graph to mean a set of vertices and a set of undirected connections or 'edges's between them. To the mathematician the definition is straightforward but to the entity modeller it is less so and is somewhat unsatisfying. There is a difficulty and this difficulty lies at the heart of entity modelling. An understanding of the difficulty reveals simultaneously a gap in the entity relational approach by comparison with the mathematical approach but also an affinity between it and the means of language and communication.

To model a bidirectional graph is to add a reverse relationship to any of the models of a directed graph. If we start from the left hand side of figure 14 then we get this model:
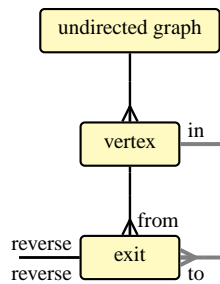
In this model we have greyed out the 'to' relationship to show that it has become redundant. This is because the vertex that an exit leads to is always the same vertex that its reverse leads from.

6 Context and Relationship Scope

Some errors arise through confusions of context and some such, in programming parlance, are called scope violations or scope errors. These are different to, but can be related with, the category mistakes identified by Gilbert Ryle. Whereas a category mistake is an attempt to relate entities whose types are inconsistent with the defined end types of a relationship, a scope violation occurs if the types are correct but the contexts of the respective entities, or, equivalently, the enclosing composite entities (see wider perspective on context), are inconsistent with the defined scope of the relationship. Scope violations are attempts to relate across contexts improperly.

Knowledge of a relationship's scope is a very significant part of understanding how a relationship is used and failure to respect this aspect of proper usage is what constitutes a scope

violation.

For example when talking about the cast members of a performance of King Lear it would be a scope error to suggest that a member of cast play the character Desdemona for Desdemona is a character within the scope of different play namely Othello. The type of entity is correct for 'Desdemona' is indeed a 'character' of a play, but the context is not. It is part of our knowledge of the 'plays part of'/'part played by' relationship (see figure 15) that this is a relationship whose scope is local to the enclosing 'play' context. We can say that it is intra-play rather than inter-play.

It would be a scope error in a conversation about antipodeans to assert that the New Zealand born physicist Ernest Rutherford could have been a native of Nelson in Lancashire, a place just 20 miles away from where he was Professor of Physics. Not only would it be a factual error - it would be non-sensical and this is because of the grasp we have of the meta-relationship between relationships 'country of birth' and 'place of birth', that the latter is a more detailed version of the former. The assertion would violate the scope of the 'place of birth' relationship (see figure 16).

Figure 15 – Model of the Performance of a Play
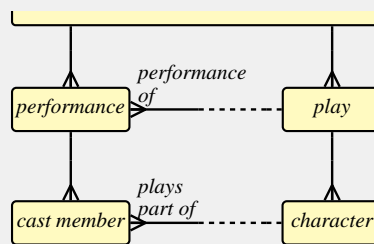
Likewise it would be a scope error to think that a local telephone call could be made between different countries or that the hydrogen of a water molecule could be covalently bonded to the oxygen of a *different* molecule or that the captain of one team in a cricket match might be scheduled to bat for the opposing team. All of these errors are characterised as failures to respect the scopes of relationships.

Figure 16 – The relationship 'place of birth' is scope constrained to 'country of birth'.
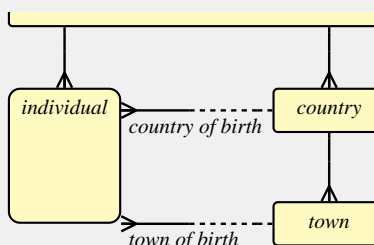
In accord with the most fundamental principles of information theory the more constrained a relationship is in its scope then the less the information needed to express its individual
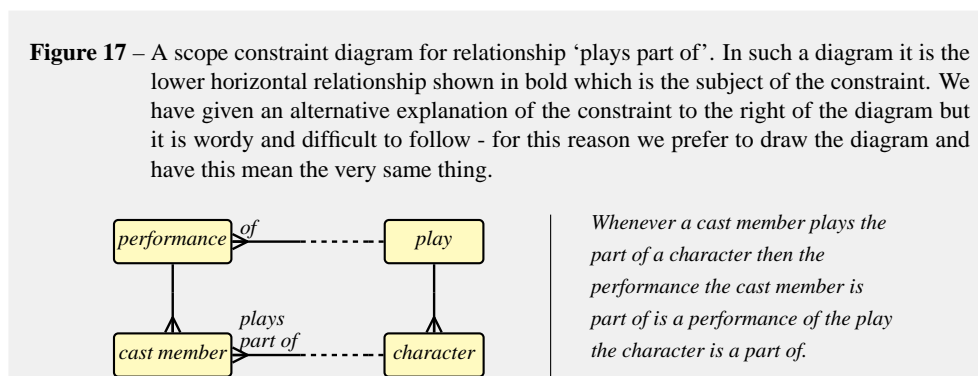
instances or facts of which it comprises. So the definition of relationship scopes is intimately connected to specifying information requirements for representing or communicating relationship instances.

For example if we wish to propose that, in the context of a performance of King Lear, we give someone the role of 'the fool' then we do not have to say 'the fool in King Lear' we simply have to say 'the fool' for our shared knowledge of the scope of the relationship 'plays the part of'/'part played by' implies the rest. Contrast this to the description often given of previous parts played by the actors as an enumeration of characters within plays.

7   Diagrams Expressing Scopes

To summarise, the 'plays the part of' relationship is limited in scope. In the specific context of a cast member, the part they play is in the same play as the performance is a performance of.

The relationship 'plays the part of' has a type constraint : in its proper usage it has to relate cast members with characters but it also has a scope constraint: that the character is a character within the context of the play being performed.

Figure 17 – A scope constraint diagram for relationship 'plays part of'. In such a diagram it is the lower horizontal relationship shown in bold which is the subject of the constraint. We have given an alternative explanation of the constraint to the right of the diagram but it is wordy and difficult to follow - for this reason we prefer to draw the diagram and have this mean the very same thing.

Whenever a cast member plays the part of a character then the performance the cast member is part of is a performance of the play the character is a part of.

In mathematical notation it is possible to include the scope constraint as a more general kind of type constraint than can be expressed in an entity model, namely a 'dependent type constraint'. In entity modelling this is not possible and every relationship defined in a model should have a scope constraint specified for it. There is no standard way of doing this but an accompanying diagram, one per associative relationship is a satisfactory way of doing this. Figure 17 is the scope constraint diagram for relationship 'plays part of'.

The diagram in figure **??** can be interpreted as the scope constraints for relationship 'assigned to' within the context of the entity models of figure 5. The text on the right of the figure explains the constraint expressed by the scope diagram - it seems obvious but this is so only if we *know* this model and this relationship i.e. providing we understand its proper usage. In this example there is a single entity type at the top of the diagram - therefore we call the diagram a scope triangle rather than a scope square. If we allow of the use of identity relationship in a scope square and allow it to be drawn horizontally then any scope triangle can be re-expressed as a scope square as illustrated by figure 19.

Figure 18 – A scope constraint diagram for relationship 'assigned to'. This is a scope triangle because rooted at a single entity type.
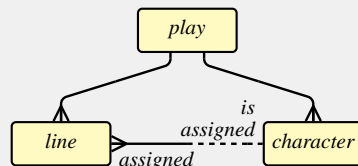
Whenever a line of a play is assigned to a character then the play the character is part of is the same play as the line is part of.

Some relationships may be unconstrained in their scope in the sense that they are global in their reach. We have given an example of such a relationship, 'translation of', in figure 6. The scope of this relationship, the fact that it is unconstrained, is expressed by the relationship scope diagram in figure 20. By way of explanation - what this diagram says is :

> The absolute of the language of a translated book is the absolute of the language of the native book it is a translation of.

In other words it says that the relationship 'translation of' is such that two absolutes are equal - which is to say nothing at all about the relationship because *a priori* all absolutes are equal as absolute is unique of its type.

Figure 19 – The identity relationship 'same as' used to express the scope triangle of figure **??** as a square.
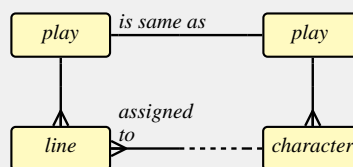
Figure 20 – An example of a scope diagram for a relationship which is unconstrained in scope - the scope diagram is rooted at absolute.

14
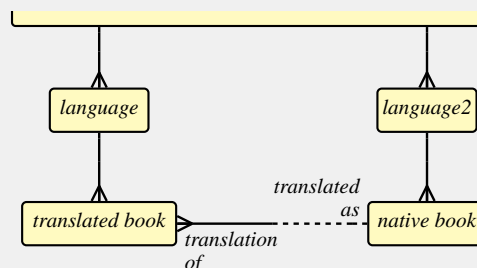
8 Further examples

Ryle's second example is of a child witnessing the march-past of a division of soldiers. After having had battalions, batteries, squadrons, etc. pointed out, the child asks when is the division going to appear. 'The march-past was not a parade of battalions, batteries, squadrons and a division; it was a parade of the battalions, batteries and squadrons of a division.'.

The child might have asked further in discussion of a battalion and its sergeant major, to what battalion does the sergeant major belong? This would have been to fail to understand the scope of the 'sergeant major' or 'leader' relationship (see figure 21. Contrast this to asking in a discussion of a university examination of what institution an external examiner belongs.

It is part of our understanding of the 'sergeant major' relationship that a sergeant major is a soldier within the company, battalion or squadron of which he is the leader.

Figure 21 – Based on Ryle's second example of category mistakes
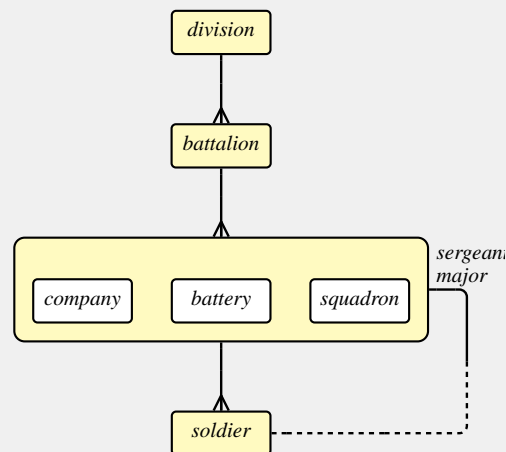
9 Disambiguation

We are not alone should we dispair at the difficult of expression of concepts. Poet's have expressed this difficulty and writer H.G.wells:
The examples in this section illustrate disambiguation of concepts via relationship modelling.

9.1 book(1) and book(2)

It is often the case that a single word in a single dictionary sense can be used at a number of different levels. We can illustrate this with the word 'book' which in its primary sense seems unambiguous but consider how we can ask the question:

> *How many books were published last year in the UK?*

and contrast with our interpretation of:

> *How many books were sold last year in the UK?*

We expect an answer to the first question to be a count of book titles or ISBN numbers but we expect an answer to the second to be a count of actual physical books sold by publishers. The use of the word 'book' in the first question is at a different level to the use in the second question. The two are related and we classify as a reference relationship:
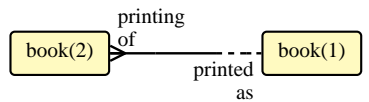

An entity model can help disambiguate these distinct uses. We have the situation shown in figure 22.

Figure 22 – Different uses of 'book' as a type of entity.
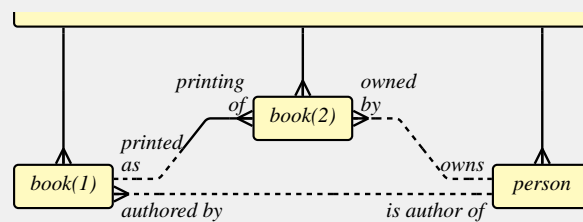
9.2 word(1) and word(2)

The word 'word' itself is also used at different levels. For example if we set about counting how many words in the English language we will be counting a different type of entity than if we count words in an essay or in an advertisement that we pay for by the word. There is a clear type instance relationship between the two which we again classify as a reference relationship:
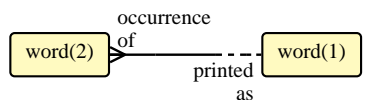
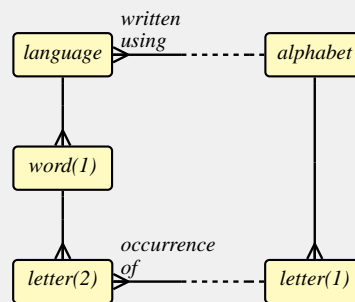**Figure 23** – The occurrence relation between multiple ways uses of the word 'word'.

16
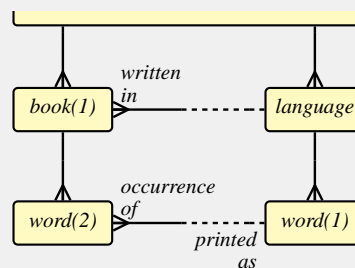
9.3 letter(1) and letter (2)

The word 'letter' also for we can count the letters in a word and each one is an occurrence of a letter from the alphabet as shown in figure 24.

Figure 24 – The scope square of the 'occurrence of' relationship between multiple ways uses of the word 'letter'.

9.4 word(3) and letter(3)

In the model in figure 25 we introduce a third type of 'word' and third type of 'letter':

> *word (1) - the word as defined by the dictionary. This is the level I am using when I say 'What does that word mean?'*

> *word (2) - the words that appear in a book as conceived of by the author. This is the level I have in mind when I write, for example 'In the words of T.S.Elliott' or, 'According to the word of God'. (This last usage of the word 'word' is given as a different sense in the 'Concise Oxford Dictionary').*

> *word (3) - the actual copies of the words in some copy of a book. This is the level that we use when we say 'That word is illegible'.*

9.5 The Model

This leads to the model shown in figure 26.

Returning for a moment to the question of when to model a relationship as composition (vertical) and when as reference (horizontal) we see that in the model in figure **??** the distinction between composition and reference has been representative of these considerations:

- Describing an alphabet, I describe its letters but I do not describe the vocabulary of every language using the alphabet.

- Describing the vocabulary of a language, I describe the words and how they are spelt but I do not describe the corpus of works.

- Describing the corpus of works, I do not describe the dissemination of that corpus i.e the printed copies.
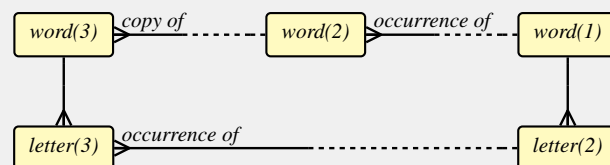
Figure 25 – Occurrence relationships between multiple uses of the word 'word'.
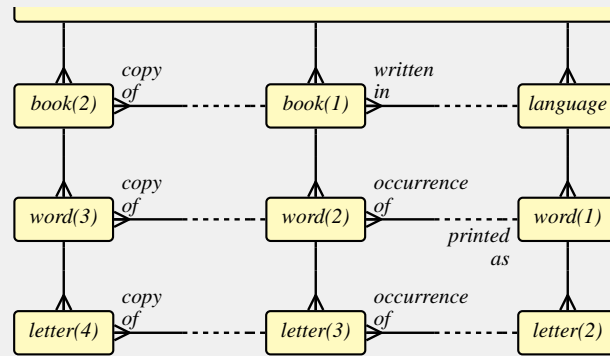
Figure 26 – This example pieces together the fragments described in the previous sections.
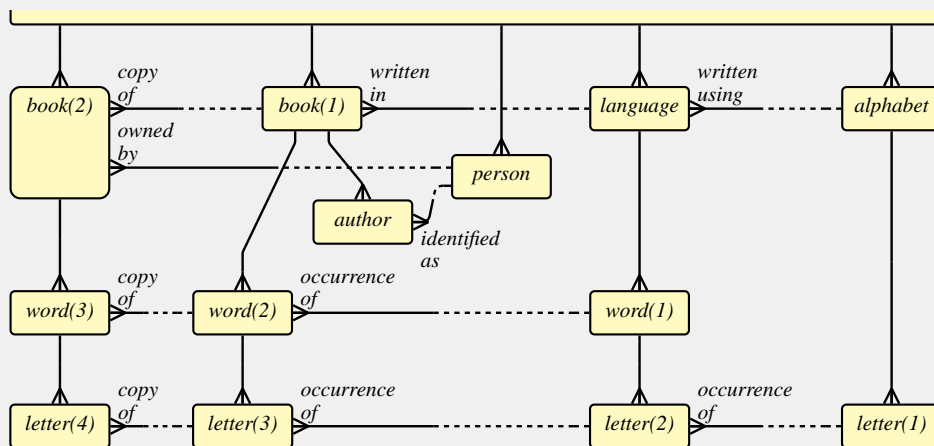

**Figure 27** – A scope square for the occurrence relation between letter(3) and letter(2).

18

10 Scope Squares - A Warning

Not all relationship squares are scope squares. In the model of figure **??** the obvious square is *not* a scope square[6]. Instead the relationship 'stays in' is global in scope i.e unconstrained. As such its scope square is as shown is figure **??**.

Figure 28 – The relationship 'stays-in' is not constrained : a cast member stays in a hotel but this does *not* have to be in the same town that the performance is at.
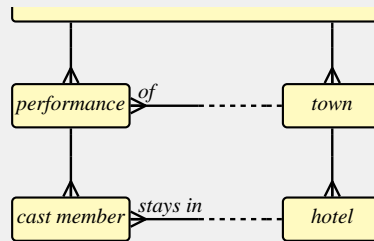
Figure 29 – The scope square for the unconstrained relationship 'stays in'.
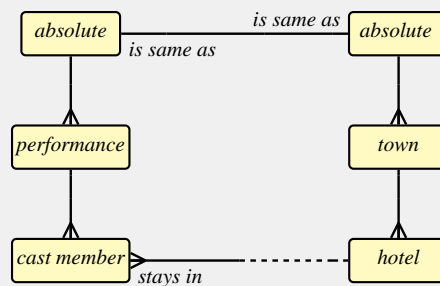
We return to the subject of scope squares in the next chapter after describing notations for constructing relationships.

---

[6]Mathematicians would say that the diagram does not commute