

How to read a data model

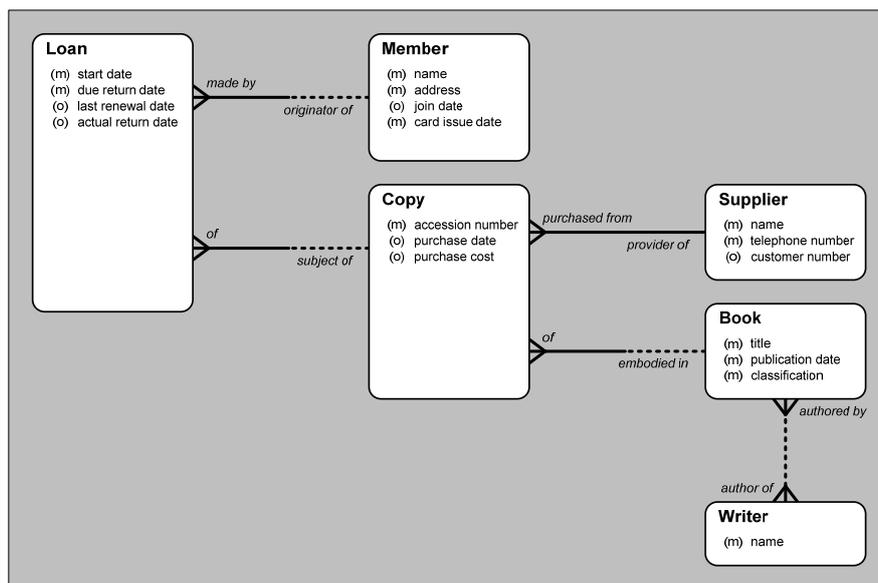
(A guide for candidates for the BCS Requirements Engineering examination)

This may be a contentious statement but it is our view that all business analysts need to be able to model the information requirements of the business that they are analysing. Normally this is done using a 'data model' using either one of the entity-relationship notations or the Class Diagram notation that is part of the Unified Modeling Language (UML).

Data modelling is normally seen as a technical activity that is all about database design, but it also provides a sensible way to provide a complete, concise and unambiguous statement of information requirements. The trick for the business analyst is to pick a notation that will be easy for a business person to understand. For this reason we recommend the use of the notation developed by Harry Ellis and Richard Barker many years ago when they were working on a complex project and were looking for accuracy in analysis whilst minimising interactions with users. The beauty of this notation is that it provides clear diagrammatic representations of information requirements using a very small number of notational artefacts. The resulting models are not cluttered with the technical detail that is typical of models drawn using other notations.

The syllabus for the oral examination for BCS Diploma in Business Analysis identifies data modelling as a key business analysis skill and also identifies a data model as an essential component of a requirements document. However, in the syllabus for the BCS Practitioner Module in Requirements Engineering the only requirement is for the candidate to be able to read a data model.

Shown below is the model of the information requirements for the management of the book stock and the book loans for a fictitious library.



Our data models (or models that express information requirements) show three concepts diagrammatically:

- entity types,
- attributes and
- relationships.

Entity types represent things or concepts of significance to the business about which information needs to be recorded. Entity types are shown diagrammatically using a box named with a singular noun. In our model we have six entity types: "Member", "Loan", "Copy", "Book", "Writer" and "Supplier". The "Member" entity type represents all of the members of the library. Each individual member is known as an entity, ie an instance of the type.

Attributes represent some of the information that is to be recorded about the entities. An attribute is used to qualify, identify, classify, quantify or in some other way express the state of an entity. For the "Member" entity type the attributes are "name", "address", "join date" and "card issue date". In our model each attribute is annotated with either "(m)" or "(o)" to show whether the attribute is mandatory or optional. The attributes "name", "address" and "card issue date" are mandatory so each instance of the "Member" entity type must have a name, an address and a card issue date. The attribute "join date" is optional so each instance of the "Member" entity type may have a join date. Not every instance of the "Member" entity type needs to have a join date.

Relationships represent business associations between any two instances of entity types. They, therefore, represent additional information requirements. Relationships are shown diagrammatically using a line. Using the notation of the lines and the names of the relationships it is possible to 'read' each relationship as two sentences.

Consider the relationship between the "Loan" entity type and the "Member" entity type. Reading this relationship from right to left we have the sentence:

"Each **MEMBER** may be *originator of* one or more **LOANS**"

We have used text formatting, bold small capitals, underlining and italics, to indicate the different elements of the sentence, which is constructed using the following rules:

- The word 'Each' is used because the box with the word "Member" inside it is an entity type (that is, it represents all instances of the type - all the members) but we want to refer to a single instance of the type.
- 'Each' is followed by the name of the entity at the end from which we are starting the sentence - in this case this is "**MEMBER**".
- The term 'may be' is used because not every member has to have taken out loans - this is represented on the diagram by a dotted line at the **MEMBER** end of the relationship and a dotted line is always read as 'may be'.
- '*originator of*' comes from the name of the relationship at the **MEMBER** end of the relationship.
- The term 'one or more' is used because there is a "crow's foot" symbol (the inverted arrow head) at the far end, the **LOAN** end, of the relationship - a "crow's foot" is always read as 'one or more'.
- '**LOANS**' comes from the name of the left hand entity type, but we make it plural so that the sentence reads easily.

But each relationship also has to be read in the opposite direction. Reading the relationship from left to right - from **LOAN** to **MEMBER** - we have the sentence:

"Each **LOAN** must be *made by* one and only one **MEMBER**"

This sentence is constructed as follows:

- 'Each' because we want to refer to a single instance of the type.
- '**LOAN**' from the name of the left hand entity.
- 'must be' because every loan has to be associated with a member (the member who took out the loan) - this is represented on the diagram by the solid line at the **LOAN** end of the relationship and a solid line is always read as 'must be'.
- '*made by*' comes from the name of the relationship at the **LOAN** end of the relationship.

- The term 'one and only one' is used because there is no "crow's foot" at the far end, the **MEMBER** end, of the relationship - the absence of a "crow's foot" is always read as 'one and only one'.
- '**MEMBER**' comes from the name of the right hand entity type.

In data modelling parlance, this relationship is known as a one-to-many relationship. A member can be associated with many loans (Each **MEMBER** may be originator of one or more LOANS) but a loan can only be associated one member (Each **LOAN** must be made by one and only one MEMBER) through this relationship. It is also possible for a data model to include one-to-one and many-to-many relationships.

So, our model represents the following information requirements:

For every member the following information must be recorded:

- their name ("name" is a mandatory attribute),
- their address ("address" is a mandatory attribute),
- the date that their membership card was issued ("card issue date" is a mandatory attribute).

Additionally, each member may have the following information recorded:

- the date they joined the library ("join date" is an optional attribute),
- any loans they have taken out ("Each **MEMBER** may be originator of one or more LOANS" - we sometimes say that a member may originate zero, one or more loans).

For every loan the following information must be recorded:

- the start date of the loan ("start date" is a mandatory attribute),
- the date the loan is due to be returned ("due return date" is a mandatory attribute),
- the member who took out the loan (Each **LOAN** must be made by one and only one MEMBER),
- the copy of the book that is the subject of the loan (Each **LOAN** must be of one and only one COPY).

Additionally, each loan may have the following information recorded:

- the date that the loan was last renewed ("last renewal date" is an optional attribute),
- the date that the loan was finally returned ("actual return date" is an optional attribute).

For every copy of a book the following information must be recorded:

- the accession number of the copy assigned by the library ("accession number" is a mandatory attribute),
- the supplier who supplied the copy (Each **COPY** must be purchased from one and only one SUPPLIER),
- the book of which this is a copy (Each **COPY** must be of one and only one BOOK).

Additionally, each copy of a book may have the following information recorded:

- the date that the copy of the book was purchased ("purchase date" is an optional attribute),
- the purchase cost of the copy of the book ("purchase cost" is an optional attribute).
- any loans of the copy of the book ("Each **COPY** may be subject of one or more LOANS").

For every book the following information must be recorded:

- the title of the book ("title" is a mandatory attribute),
- the publication date of the book ("publication date" is a mandatory attribute),
- the classification of the book ("classification" is a mandatory attribute).

Additionally, each book may have the following information recorded:

- the copies of the book that are held by the library (Each **BOOK** may be embodied in one or more COPIES),
- the authors, if any, of the book (Each **BOOK** may be authored by one or more WRITERS).

For every writer the following information must be recorded:

- the name of the writer ("name" is a mandatory attribute).

Additionally, each writer may have the following information recorded:

- the books, if any, that have authored (Each **WRITER** may be author of one or more BOOKS).

For every supplier the following information must be recorded:

- the name of the supplier ("name" is a mandatory attribute),
- the telephone number of the supplier ("telephone number" is a mandatory attribute),
- the copies supplied by the supplier (Each **SUPPLIER** must be provider of one or more COPIES - the implication is that only details of suppliers who have actually supplied copies to the library are recorded).

Additionally, each supplier may have the following information recorded:

- the customer number of the library assigned by the supplier ("customer number" is an optional attribute).

*Keith Gordon
January 2013*