

# BUILDING QUALITY DATA MODELS

David C. Hay  
Essential Strategies, Inc.

## INTRODUCTION

Some people view data models as necessary evils — a way to document data base structures. For such people, the primary objective in producing a data model is to get it done and get on with the real work of developing systems.

Now, if data modeling is truly an extra nuisance task, and it isn't important to do well, then it probably shouldn't be done at all. Indeed, if that is the case, we should skip that step and get on with building systems.

On the other hand, if data models are important parts of the development process, both to improve communications with users, and to provide a sound logical basis for data base design, then it is important to do the best possible job of producing them.

This paper is intended to provide some guidelines to help you with that task. Most of the paper is about specific rules and guidelines. This is in hopes that specific guidance to people just learning data modeling will be helpful. Prior to considering the specific guidelines, however, there are three important points to be made about a modeler's *attitude* toward h' craft.

First, there is no substitute for clear thinking. For a model to be effective, it must describe that which is *fundamental* to a business, not simply that which appears on the surface. Entities should be things that really are of significance to the business, not just the things that are causing your interviewees the most trouble today.

Second, aesthetics are important. If you are to present this drawing to management and win them over, it must be attractive. This means that it must be well balanced, interesting, and uncluttered. Take a walk outside the computer section of the bookstore and wander over to the graphic arts (or the art) section. These are the books we should be reading in our industry now (and not only so we can produce better data models, but so we can do better graphical user interfaces and report layouts as well).<sup>1</sup>

The third important point is that, if you are to be successful in data modeling (as in any other field) you have to be *passionate* about it. You have to have a mind set that cannot turn in a drawing that "doesn't look right". If something is wrong with a model, you can't rest until you have fixed it.

---

<sup>1</sup> A particularly interesting (and attractive) book that offers aesthetic guidance is by Edward Tufte, and is called *Envisioning Information* (Cheshire, CT:Graphics Press, 1990).

It is true that data models encompass a tremendous number of details, both in the logic of the model and in its representation on paper. Building them is very hard work. It is very easy to wave our hands over less than perfect work, and say that the model is “good enough”.

To do so, however, brings to mind Gerald Weinberg’s story of Corporal McAndrew’s Arkansas Stewed Possum Company, Ltd.<sup>2</sup> Over the years a hundred or so small changes were made to the recipe for its previously very successful possum-patties. Each change cut costs and made the company more profitable. In each case, the “money saving modification made *absolutely no difference* to customer perception and reception of the possum-patty. As far as anyone can tell, the new formula is *indistinguishable* from our current recipe.” Of course, over the years the possum-patties have become so inedible that even the company managers won’t eat them.

This paper is about how not to make small compromises in drawing data models that, cumulatively, undermine the usefulness of the model both as a communication tool and as a basis for logical data base design.\*

## STANDARDS FOR THE UNDERLYING MODEL

### **Entities**

1. Use a single, unambiguous name for each entity, describing the thing of significance it represents.
  - Use common English (or French, or German, or whatever) names.
  - If possible, use names familiar to your audience, but if the only available terms are ambiguous or do not have the same meaning to all concerned, create new terms as precisely as possible, and define them clearly.
  - Do not use abbreviations, acronyms, or table names. (No FIN LOC MASTER, for example. Use FINANCIAL LOCATION.)
  - Use the singular name of an occurrence of the thing. (No INSTRUCTIONS. Use INSTRUCTION. No PRODUCT STRUCTURE. Use PRODUCT STRUCTURE ELEMENT.
  - Accompany each entity with a clear, concise definition in the dictionary. Use correct grammar and don’t use jargon, acronyms, or abbreviations. Don’t use the entity to define itself.

---

<sup>2</sup> Weinberg , Gerald, *The Secrets of Consulting*. New York:Dorset House Publishing Co., 1985, pp 129-130.

\* These rules are derived from work done with the USDA Forest Service. The author appreciates the efforts of Regina Kidd, John Butler, Dale Lowery and others who contributed to that effort.

2. Share entities across applications wherever possible.
3. When an entity is identified, pursue its analysis until all possible relationships have been identified. If you know that you have not had time to analyze it completely, indicate that in the dictionary, and come back to it later.
4. A sub-type is a subset of the occurrences of an entity. An occurrence of an entity may appear in one and only one sub-type.

Use of the sub-type implies two things: first, the sub-type probably has attributes or relationships which distinguish it from other sub-types; second, the division of an entity into sub-types represents a fundamental and relatively stable organization of the data.

In general, do not use sub-types where examples would do. Do include examples as text on the drawing, where they clarify the meaning of the entity.

### ***Relationships***

5. Put one name at each end of each relationship.
6. Relationship names are prepositional phrases, not verbs. Don't use "has", for example.
7. Each relationship name must fit into the following structure to produce simple, understandable English sentences:

Each <entity 1>  
 {must be|may be}  
 <relationship name>  
 {one and only one|one or more}  
 <entity 2>[s]  
 .

8. Do not use weak relationship names, such as "associated with", or "relate". The fact that we drew the relationship line already asserts that the entities are *associated with* or *related to* each other. Use something concrete enough that it could be wrong — this allows your client to correct it.

If all else fails, you may use a gerund (as in "Each PARTY may be *owning* one or more ASSETS"), but it is weak, so avoid it if possible.

## **STANDARDS FOR ORGANIZING THE DRAWING**

### ***In general***

9. Organize the drawing to be visually attractive.
  - Give equal weight to the left and right side of the diagram.
  - Leave white space judiciously.
  - Don't "scrunch" elements together unless absolutely necessary.

10. If using Oracle's Designer/2000 and a PostScript printer on Windows 95, go to the printer's *properties*, and set *Graphics / Scaling* to 70%. If you do not have a Postscript printer, install a Postscript printer drive anyway, and in *Printer Setup*, pretend that you are using it. When you are about to print, be sure to save your work first. When printing (not in printer setup), specify the real printer to use. Designer/2000 will then reformat your display. Quit the drawing without saving it.

The printer will have been fooled into printing the drawing at 70%, however.<sup>3</sup>

This allows about 10-15 entities on a page. This is the best size for a diagram to be readable. If you need more entities, you probably need two pages.

If you are using Windows NT, scaling is not possible, so you will find yourself using four pages for the drawing. This permits a few more entities, but don't get carried away. For readability stick to 10-15 entities per page.

11. Break the overall model into topics, so that each diagram concerns only one or two subjects. Try to organize the drawings so that each represents a complete "logical horizon" (an entity with all the entities it requires). If the logical horizon is too deep to limit the drawing to 25 entities or so, split it into two diagrams, with enough entities overlapping to make it clear how they all are related.
12. Laying out the diagram in "portrait" format is better for publication, but laying it out in "landscape" form makes more efficient use of the screen. Whichever you choose, use it consistently for all diagrams.

### **Crow's feet**

13. Orient the entities and relationships so that the "crows feet" have their toes pointing to the left or top of the diagram.
  - In general, this rule provides a "shape" to the model, so that it is not simply a random collection of boxes and lines.
  - It clusters tangible, reference entities in the lower right, while the less tangible transactions and intersect entities congregate in the upper left.
  - It allows someone who has not seen the diagram before to quickly identify which are defining entities, and which constitute the organization's transactions.

### **Relative entity sizes**

14. Vary the size of the entity boxes to make the diagram more interesting. In general, make each entity's size proportional to its importance.

---

<sup>3</sup> For more detailed instructions on this process, see  
<http://www.essentialstrategies.com/publications/designer2000/shrinkd2k.htm>.

15. As necessary, stretch entities to accommodate relationship problems (described below).

### **Legend**

16. Have a consistent legend in the same place on every drawing, with a consistent format. Engineering convention calls for it to be in the lower right corner. One you may try is as follows:

```
<COMPANY NAME>
<SYSTEM NAME>
<OPTIONALLY, APPLICATION NAME>

<DRAWING NAME>
<DATE>
<ARTIST>
<ARTIST'S PHONE NUMBER, ADDRESS, ETC.)
```

- The company name is in upper case; the other lines are in upper and lower case.
  - In Oracle's Designer/2000, under *File / Summary Information*, select *Diagram, Title, Author, and Information System*. Under *Title*, enter the date. Under *Author*, enter your name.
17. If additional annotation is required ("This is a model of the physical data base", for example), either add this as another line in the legend, or put it along the bottom of the drawing. It may be appropriate to put a box around that as well.

## **DETAILED STANDARDS FOR THE DRAWING**

### **Entities**

18. Usually center subtypes within super-types, at least horizontally. Leave a border around each subtype. There are legitimate exceptions to the centering rule. There are none to the rule about borders.
19. Center text horizontally within an entity. If the entity is squarish, center the text vertically, as well. If it is vertically elongated, put the text approximately one quarter of the way from the top. If it is horizontally elongated, you may either center the text vertically, or place it along the upper edge.

## **Relationships**

20. A bend in a relationship becomes a symbol on the picture: To minimize clutter on the diagram, don't bend relationships, if possible.
  - Never bend it at a right angle.
  - If necessary, make it an oblique angle.
21. Make most relationships orthogonal (north/south, or east/west). An occasional diagonal is OK. Don't make it look like an afterthought.
22. Where possible, avoid crossing relationship lines, but this is not as important as not bending lines. Do not cross entities, ever.
23. Where possible, arrange relationship names to be read clockwise. That is, for vertical relationships, place them in the upper right and lower left. For horizontal ones, place them in the upper left and lower right. This rule may be waved if there is not room to follow it.
24. If possible, do not let relationship names cross super-type boundaries.
25. If a sub-type has a pig's ear, allow enough room so that the relationship does not extend outside the super-type.
26. If there is room, place any pig's ear on the lower right corner of the entity, with the "many" end on the right side of the entity.

## **Attributes**

27. For clarity in presenting the entities themselves, it is not necessary to display attributes, although as people get more intimately involved in the model it becomes useful to do so.
  - In the early part of the process, don't add more than necessary to illustrate a point.
  - Don't crowd the entity text.

## **AND IF YOU DON'T HAVE A CASE TOOL . . . ?**

Sometimes it is necessary to produce a data model when no CASE tools are available. (Remember, we are only talking to people who are so passionate about data models that they would do them by hand, if necessary.) Numerous graphic packages are available for PC's, but unfortunately, most vendors are so fixed on making it possible to draw space craft and cars in three dimensions, that they aren't very accommodating to the simpler, but highly disciplined requirements of the data modeler.

Your author's favorite package is McDraft, which is finally available for PC's, as well as the MacIntosh. MicroGrafix designer and Aldus Freehand also work well. PC Draw,

PowerPoint, and PageMaker may be used if nothing else is available, but each has limitations.

The good news about using a graphics package is that your models are *much* more attractive. The bad news is that you do have to do the work of keeping relationships connected without stretching the crow's feet. You also are not building an underlying dictionary.

Some of the above rules have variations, if you are doing this from scratch.

28. Use a consistent size for entity names and relationship names. In general, Arial (or helvetica) bold, in 12-14 points is suitable for entity names. Arial italic, in 10-12 points works well for relationship names. Use 10-12 point Times Roman for attributes.
29. Magnify the crow's feet on the screen enough to insure that you can see the lines meet cleanly at the center. If they aren't quite right, it doesn't necessarily show, but the drawing looks somehow less neat.

## HAY'S FAVORITE RELATIONSHIP NAMES:

Your author has frequently been asked for a list of typical relationship names. He is reluctant to do this, because the correct name for a relationship should come from the relationship, and not from some standard list. Still, it is true that a lot of relationship names come up frequently. Here then, is a modest list, with no guarantees that it will make relationship names leap from the page:

### (definitions)

- an example of/embodyed in
- defined by/definer of
- defined by/the definition of
- in (or currently in)/location of (or site of)
- in terms of/reference for
- of/classification for
- part of/composed of
- responsible for/responsibility of
- specified for/specified by
- to carry out/carried out in

### (contracts)

- for/beneficiary of
- for/bought via
- from/buyer in
- to/seller in

### (intersect entities)

- in <or whatever>/object of
- of <or whatever>/subject to

### (product structures)

- the use of/part of
- the use in/assembled from

### (miscellaneous)

- based on/basis for
- played by/player of
- requested by/requester of
- to monitor/monitored by
- triggered by/trigger of

