Modeling Standards and Data

Even if a Standards document does exist, which contains guidance\instructions on how to capture and manage data (i.e. not just diagrams), you should check for "problem patterns", e.g.

Incorrect Usage

- Single-line text fields used in such a way that they should be multi-line text fields

- Categories that are not allowed in the standards or where the concept has been clearly misunderstood (e.g. one client the usual Level 1, Level 2 Process, etc, and then had additional categories like "Recruitment Process", "Development Process" - in other words, they were trying to capture in categories what should have been associations to Organizations. It was also not clear what the difference was between say a Level 2 Process and a Recruitment Process. In other words, they had confused subject matter and structure.

- Categories that seem to have been made up "on the fly" (i.e. too many or with no obvious reason for them)

Poor quality of the data

- Multi-line text fields that are done in note form or have no punctuation or contain spelling mistakes

- Duplication in categories (usually where the same concept has two different spellings or there are "Copy of" categories

- Duplication in multi-line text fields: a common problem for the unwary is loading data with Auto Modeler with Append Text switched on. It's surprising how often this goes unnoticed!

- URL\hyperlink properties may be set up using absolute links (the result of using the Browse button to navigate to a document); these will fail when published.

Inconsistent Usage

- A property that is populated in only a few instances - this might suggest that it is not really being used

- Two properties that seem to be capturing the same data (which means users do not know which one to complete and which one should be published)

- Two multi-line text fields where the user is not sure what to put in them. For example, in a Process Object, there often seems to be confusion over the uses for Description and Objective; also in diagrams, should Description be used to describe the diagram or to describe the parent object, which is what it is illustrating? And if the latter, does that mean the description goes in both the Diagram and parent object description or just in the diagram?)

Empty properties

- How many objects have no description? Is this acceptable? If it is acceptable, consider using shape regions to indicate which objects on a diagram have descriptions (or which do not), so a user does not click through to empty pages.

- If an object has no properties at all, is it worth publishing object pages for that object?

Formatting of multi-line text fields

(1) Convert to HTML

What is the client's approach here? Is HTML switched on by default? If so, is the formatting appropriate to published output? Because of the way this feature uses inline styles, it is often not possible to control its look and feel when published. If the text has been uploaded from Word using Auto Modeler, the formatting

inherited will be exactly like the Word document, which is probably NOT what is wanted when published as part of a web page.

(2) Basic, unformatted text

If the client is not using HTML, the formatting is severely constrained and often leads to poor and inconsistent published output. For example,

- Sometimes there will be a space between paragraphs, other times there will not be, making the output look scrappy and unprofessional

- Things like bullet points can be inconsistent: sometimes people use a dash, other times an asterisk, and so. If a bulleted list has been pasted in, you often end up with weird characters that might render as a symbol or question mark in published output.

(3) Casewise Mark-Up Language

Sometimes the properties might have Casewise Mark-up Language tags in it. These can be recognized by the use of curly brackets, e.g. {i) means italics. Older, more established clients may have put these in deliberately; others may not know they are there - if uploading from Word (under some circumstances), Word will insert these tags to reflect any formatting in the source document.

Whether deliberately or accidentally, they are a problem as many of the tags do not work correctly. I would recommend doing a search and replace to strip them out and then configuring Auto Modeler (Advanced button) to not use them.

(4) [HTML ON] tag

Another way of formatting text is to use the [HTML ON] tag; this turns a multi-line text field into a HTML Editor allowing you to put in HTML tags. This works fine for HTML published output, but not for Word output or c4w generated documents.

Are Data Types best for purpose?

A user may set up a Single Line text field when a drop-down box or Date Field may be better, or vice versa. For example, Dates do not come out well in published output so a single line text field maybe better;, but if you want to use comparative shape regions using date ranges, you must use the Date data type.

From a publishing perspective, there are other things to consider regarding data types: If you use Normal layout (as opposed to table-based) for object pages, any property other than multi-line text fields are published using incorrect formatting (i.e. both the Property Name and the property's value are a heading 3). It is not a problem with table-based output. For one client, this was such a show-stopper they converted all Properties that they wanted to populate to multi-line text fields.

Using Shape regions

Another thing to consider - again from a publishing perspective - is whether to use Shape Regions to indicate which objects have descriptions. In this way, the existence of further (useful) information is clearly flagged to the user.

Need for Standards & Guidelines

From the above, it is clear that modeling Standards should include guidelines and rules on what properties to complete and, for multi-line text fields, in what format. Regarding the latter, it may be appropriate to use HTML in some fields (e.g. if you want to embed graphics) and basic test for other fields.

Such a Standards document might also specify if any properties are mandatory; note however that though properties can be made mandatory in CM itself, this is of little use unless you are maintaining properties manually (many clients mass populate with Auto Modeler). A useful check therefore is to check whether a

standards document (if exists) specifies mandatory properties and if so how many instances do not have this property complete?

As a basic generic "pattern", I tend to think that objects should always have at least a Description and Category.