

## **TS-755 Metadata Extensibility**

In the area of “metadata extensibility” there are 2 similar questions that are often incorrectly interchanged:

1. Is the metadata extensible?
2. Is the metadata model extensible?

Often times when the user asks question 2, what they really want to know is 1., and have assumed an implementation (or have had their minds biased by a repository vendor).

Our approach has been to build “well defined extensibility constructs” into the model as opposed to being solely reliant on, or allowing, the model to be extended.

The following lists these extensibility constructs in terms of the questions that a user might ask.

### **How do I define extensions to the metadata objects with my own, unique attributes?**

There are 2 features of the metadata model that deal with adding additional attributes.

1. Extended Attributes
2. Keywords

**We do not support the concept of adding attributes to the “core” model.**

**Extended Attributes** (called Extensions in the model), are typically a simple “name-value” pair that the customer wants to define for themselves. The attribute name is very specific to the customer terminology and the meaning only is useful for them (and thus there would be no way for us to provide this functionality out of the box).

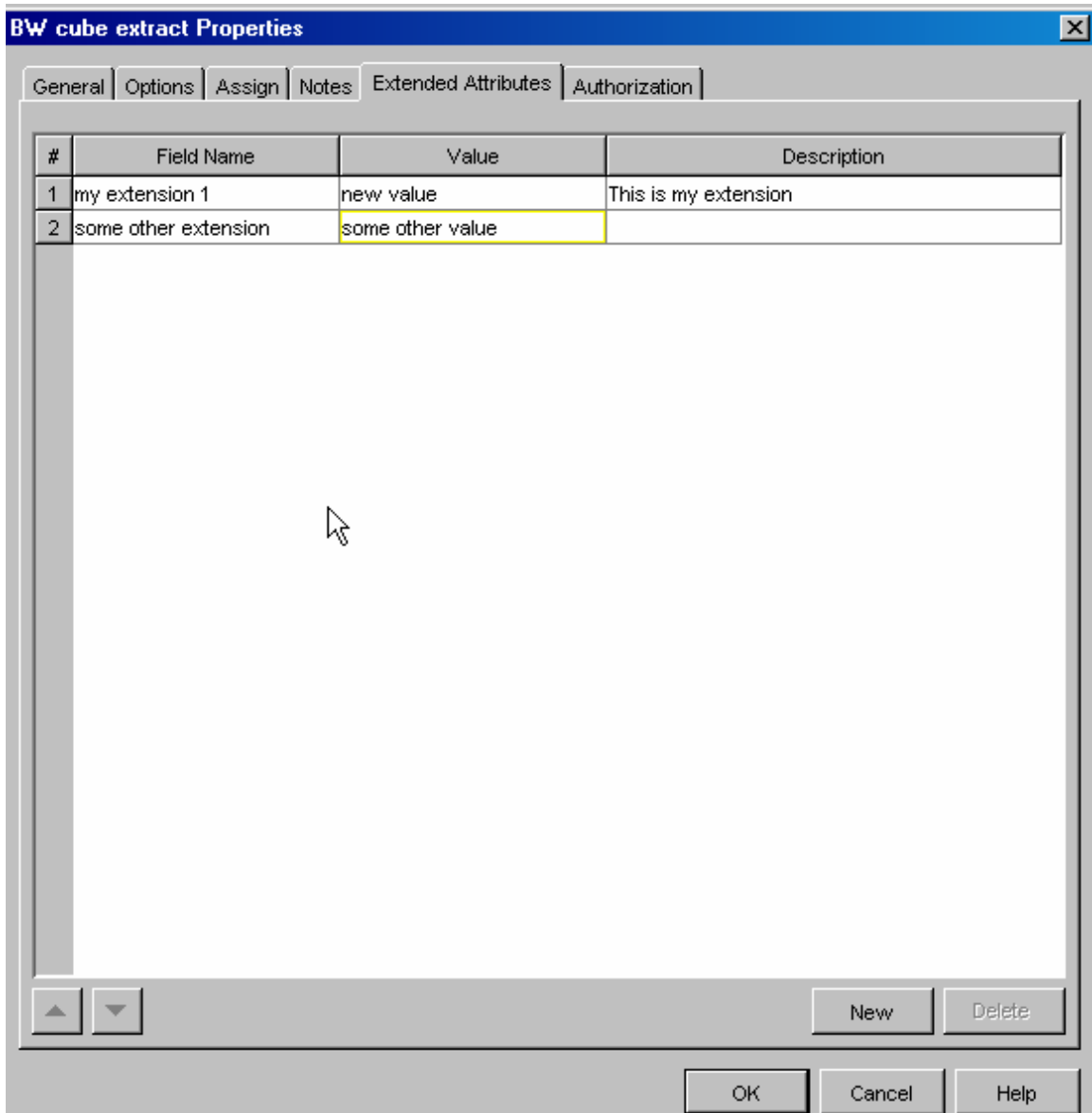


Figure 1 SAS Management Console screen for entering user Extended Attributes

These extended attributes are mostly used for documentation purposes or to access via apis for their custom built applications that extend our product functionality.

**Keywords** are just “values”, which help the user recognize the objects in their own terms.

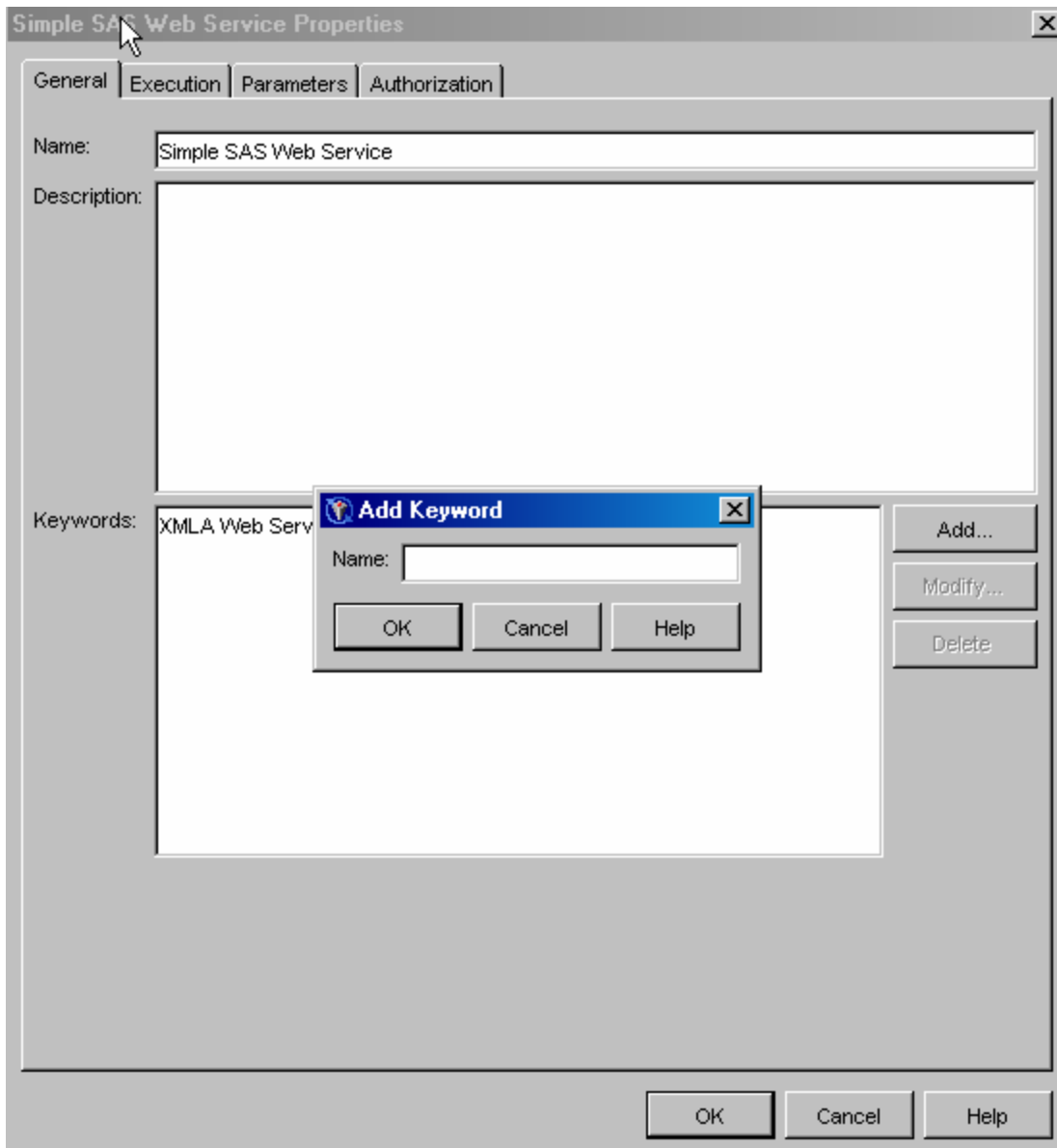


Figure 2 Stored Process Manager of SAS Management Console Screenshot of entering Keywords

### **How do I extend the descriptive information about an object?**

Typically, the user will say “we already have this documentation about this topic and we want to just link to it. Can I extend the metadata with this documentation?”

We currently provide 2 answers for this:

1. We provide a free form, unlimited length Note concept. An object can have unlimited set of notes associated with it and the same note can be shared across multiple objects.
2. We provide the ability to link existing documents, accessible via a URL, to objects.

This option is used extensively to tie information to their existing infrastructure (for example, they may already have a corporate standard for content management and they just want to link documents there to this context).

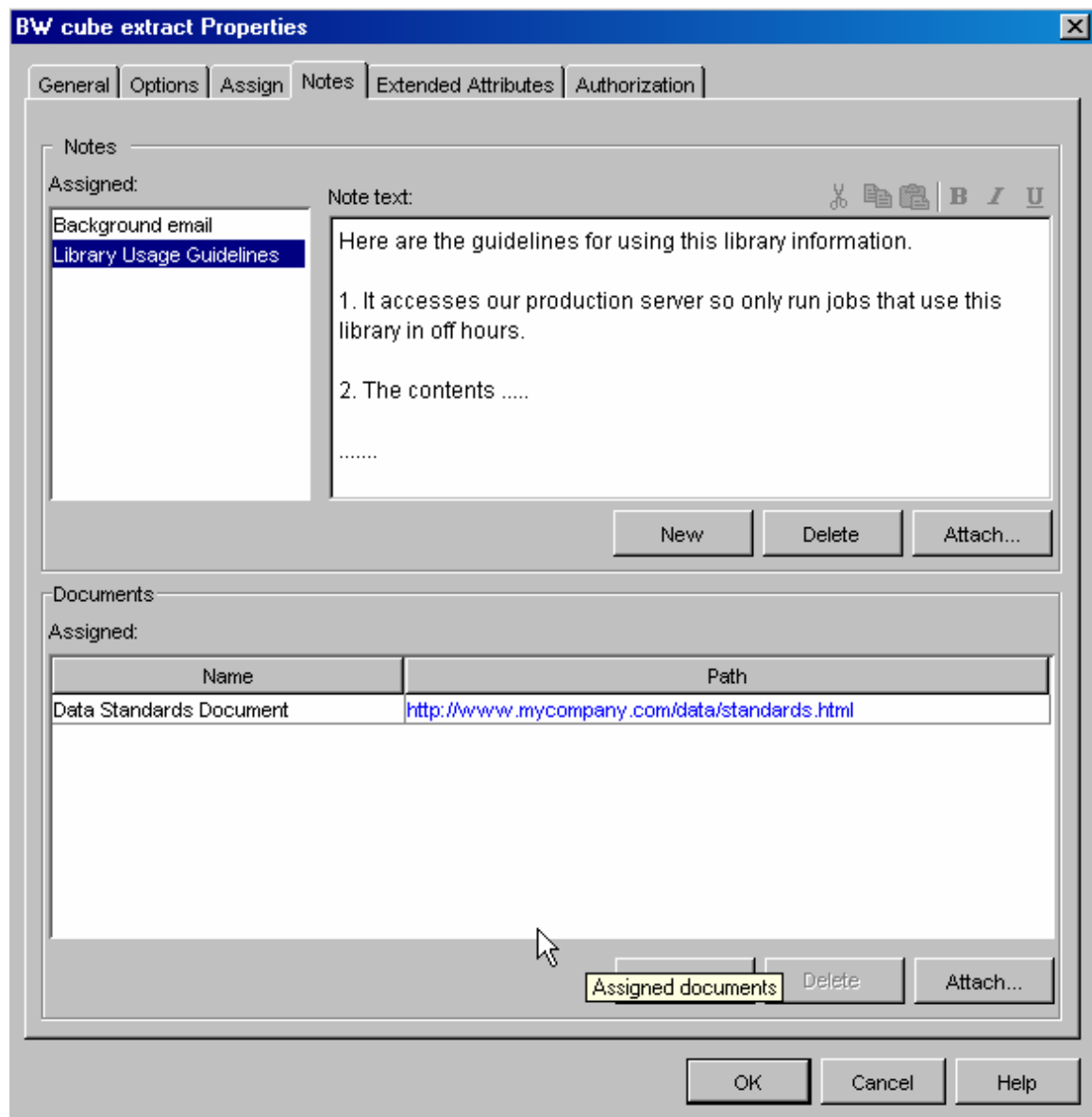


Figure 3 SAS Management Console screen for extending the descriptive information

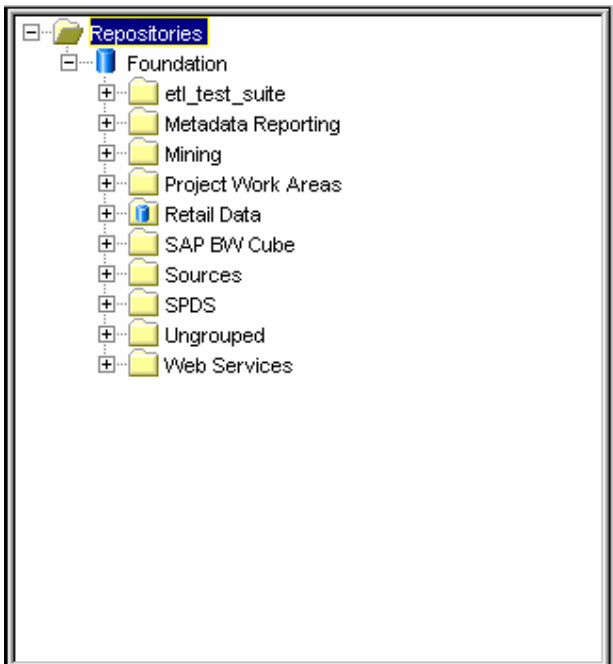
## How do I group the metadata object in different ways/views?

One of the very common requests is to extend the “view” on top of the metadata by allowing multiple groupings of metadata based on user requirements.

We provide 2 mechanisms to do this: 1) a Tree based structure and a 2) Group based structure.

The tree based structure allows you to present a concept of a hierarchical layout to the users. Because different users might want to different hierarchical views, an object can be added into multiple tree structures. Typically, the SAS Client applications organize the information in a tree structure that they think is most relevant to the user (by default). Some, allow you to define multiple structures.

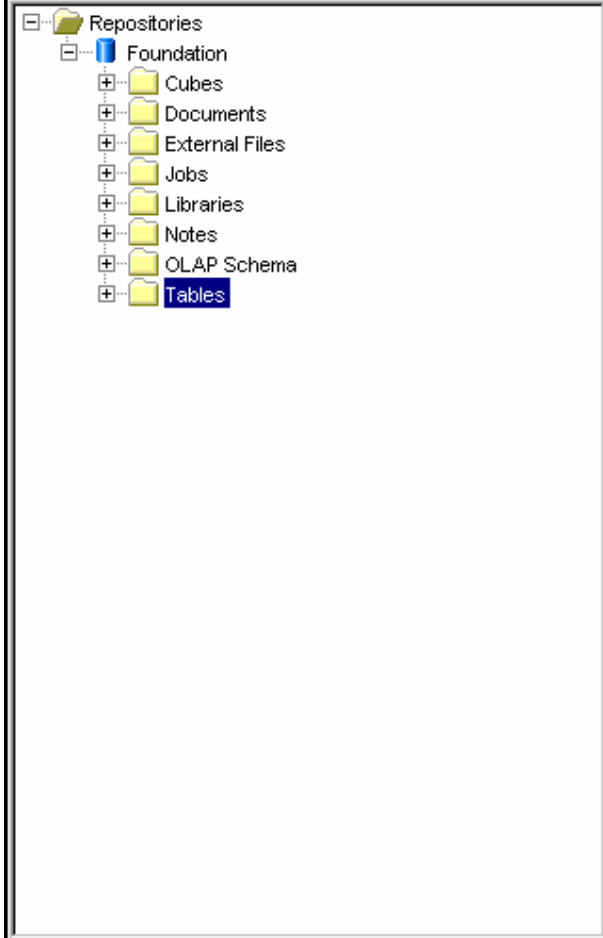
The group based structure allows you to group things in ways that aren't necessarily hierarchical but based on some other criteria the user defines, ex. Audience, likeness, ownership, source, etc. Again, an object can be included in multiple groups.



Custom

Process Library

Comparison Results



Inventory

**Figure 4 ETL Studio screen shot showing multiple grouping mechanisms**

## **How do I extend the model to include my own types and associations?**

As mentioned above, the question of extensibility usually starts with this question “Can I extend the metadata model?”. The short answer to that question is “No. You can’t currently extend the model with your own types/associations and this is not currently on the roadmap to add.”.

However, when you dig in to “What were you thinking about that you needed to extend types?”, the conversation usually turns into one of 4 things (in descending order of occurrence):

1. Wanting to extend in ways that the other mechanisms discussed in other areas of this document.
2. A context of a very “point specific” tool that they have had exposure to that had a very limited metadata model. Our metadata model contains over 160 types and a few hundred different associations. While that number is impressive, it is solidified by the fact that we have a broad gamut of applications across the Intelligence Chain that all utilize the metadata (and this model). Often times, when this is realized, the concept is already covered.
3. Trying to model “data” concepts as “metadata”, for example customer lists, product info, employee info (which is one that really gets tricky because it is both), etc. This just needs to be pointed out and indicated that it needs to be in their data model, not their metadata model.
4. Some site specific concepts. This is the one where we have to reply with “No” and try to understand whether their request is generic enough for future consideration for inclusion or is specific to them. If it is determined that we are in this situation, it’s a good time to ask whether the technology has been positioned incorrectly or the goals aren’t in line with the intent of the software. Almost always, a solution can be derived that uses the other mechanisms discussed here, but it often has an unnatural feel to it.

## **How do I extend the User Interfaces that surface metadata?**

Often times, the customers have a hard time separating what they are seeing in the User Interfaces from what is underneath it (ie. The metadata). Thus, when they ask about extensibility, sometimes they are referring to the ability to extend the user interface, either to show more details, to show the metadata in a different way, to enter some of their “user defined properties”, etc.

Certain applications, like SAS Management Console and ETL Studio, allow the user to write custom “plugins” for this purpose.

## **How do I link metadata in your repository to some other existing metadata repository?**

Sometimes, the user has metadata already in another tool and wants to “document” what the other object is. While the user could use an Extended Attribute to do this, there is an additional concept in the metadata model called and “ExternalIdentity”. This allows you to document the context of the external object (ex. What tool is it valid in) and what the actual identifier used in that tool is.

For more information, see the Foundation Submodel of the SAS Metadata Model.

There are currently no SAS Supplied Clients that allow you to create additional ones of these objects.

## **How do I extend the metadata to integrate with other metadata repositories?**

Sometimes people will ask about extensibility when they refer to the ability to read/write others’ metadata formats. For example “Is your metadata extensible to Erwin?”.

We have import/export mechanisms to deal with this situation. These primarily use the CWM (Common Warehouse Metamodel) as the intermediate format. There are tools from MetaIntegration that convert native formats to this format that we can then consume.

## **How do I extend the metadata to capture who the appropriate contacts are?**

The users typically have different “contacts” or roles people play in relation to the things that are being managed. Thus, they want to be able to register who these people are in a single place and to use the terms they are comfortable with in defining their role.

The metadata model supports this concept via the ResponsibleParty object and set of associations.

The support of this varies by application. In ETL Studio, for example, they limit the roles to only 2, Owner and Administrator, as those are the common roles in that environment. No SAS supplied client currently allows entry for generic ResponsibleParty information.

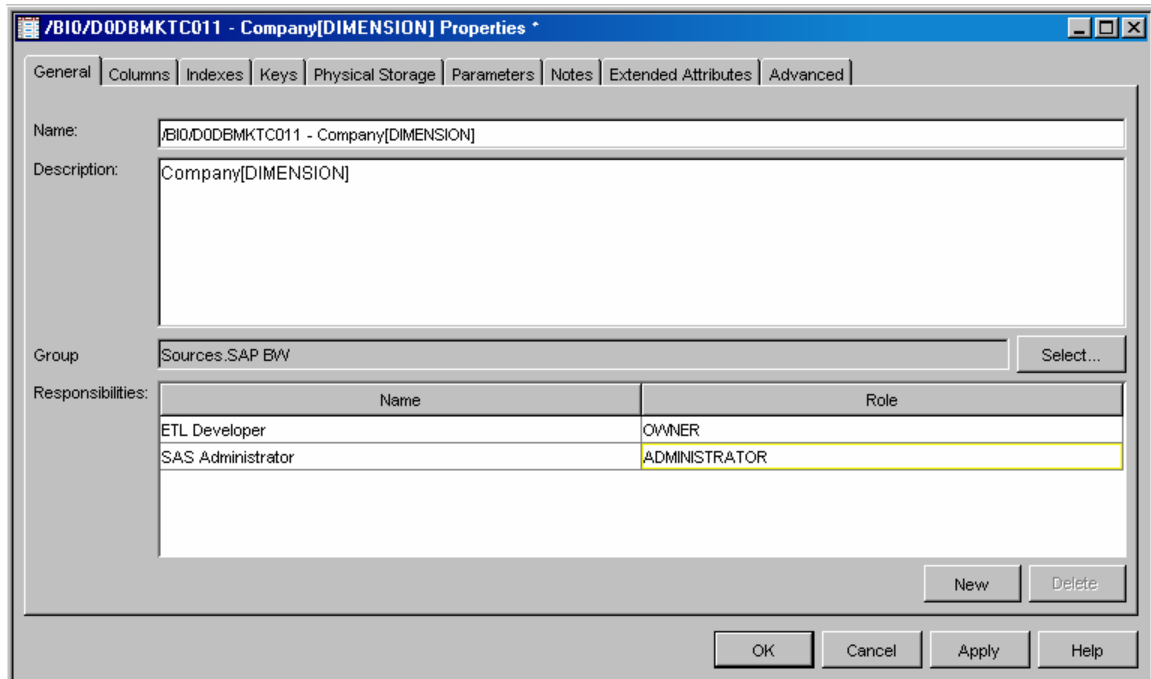


Figure 5 Screen shot of ETL Studio surfacing contacts

## Extensibility=Foreward Compatibility

I have seen situations where the user is asking about extensibility, when really they are referring to “forward compatibility”, ie. Is their investment in the metadata safe moving forward? The obvious answer to this is “Yes, your metadata is safe”.

## Summary

The SAS Metadata architecture is not meant to be a “general purpose metadata repository” for things outside of the Intelligence Architecture space. Thus, “extending the model” is not currently on the priority list to add support for. The key is in understanding not the question they asked “Can you extend the metadata model?”, but instead to understand the intent of the question they are asking. Using the built-in extension points can solve many of the customer questions or needs.