C H A P T E R

1

Introduction

About This Guide

This guide describes how to perform basic and advanced metadata management tasks using the SAS Open Metadata Interface. It also provides sample usage scenarios that describe how to use the metadata types in the SAS Metadata Model to store metadata.

The SAS Open Metadata Interface uses XML as its transport language. If you are unfamiliar with XML, see the W3C XML Specifications at **www.w3.org/TR/1998/ REC–xml–19980210** before reading this guide.

The examples is this guide show how to issue calls without regard to the programming environment. For details about writing SAS Open Metadata Interface clients, see the *SAS Open Metadata Interface: Reference.*

Introduction to the SAS Open Metadata Interface

The SAS Open Metadata Interface is an application programming interface that provides a set of methods and metadata types for reading and writing metadata about application elements. The SAS Open Metadata Interface can also be used to do such things as register repositories, define dependencies between them, define access controls for repositories and metadata objects, and to perform version control on repositories, but these are secondary tasks. The primary purpose of the SAS Open Metadata Interface is to enable you to read and write metadata.

To read or write a metadata object, you must pass a string of properties that describe that object to the appropriate SAS Open Metadata Interface method. This string identifies the appropriate metadata type to use to represent the application element and supplies or queries specific values for the object's attributes and associations.

This guide is divided into four parts:

☐ *Using the Interface to Create, Update, and Delete Metadata* describes how to use the application programming interface to create and modify metadata. It identifies the correct method to use to add, update, and delete metadata objects and their associations.

☐ *Using the Interface to Query Metadata* describes how to query metadata. The interface provides the GetMetadataObjects method to query all objects of a specific metadata type, and the GetMetadata method to query specific metadata objects.

This section describes how to use method flags and templates to control the metadata properties that you want to return.

□ *Using the Interface to Perform Repository Management Tasks* describes how to perform more advanced metadata management tasks, such as defining relationships between repositories, implementing an object locking scheme, and using change management methods, among other things.

□ *Model Usage* contains usage scenarios to help you identify the correct metadata type to use to create metadata representing the most common application elements.

Prerequisite Information

In order to successfully use SAS Open Metadata Interface methods, you must understand how to use the metadata types documentation in the *SAS Open Metadata Interface: Reference*. Review "Overview of the Metadata Model and Model Documentation" in the reference carefully before attempting to follow the examples in this guide. Metadata types for application elements are described in "Alphabetical Listing of SAS Namespace Metadata Types." This listing is available only in the online version of the *SAS Open Metadata Interface: Reference*.

The SAS Open Metadata Interface enables you to instantiate objects for method parameters and issue method calls directly from the client, or to define all of a method's parameters in an XML string and submit it to the server via a generic DoRequest method. The first approach is referred to as the "standard interface". The second is referred to as the "DoRequest method." The examples in this guide use the DoRequest method. For detailed information about the interfaces, see "Call Interfaces" in the reference. Method calls formatted for the DoRequest method can be passed in Java, Visual Basic, and C++ clients, and in PROC METADATA.

For reference information about the methods used in this book, see "Methods for Reading and Writing Metadata (IOMI Class)" in the reference.