

# SAS Metadatenmanagement

## Reporting und Analyse

Melanie Hinz  
mayato GmbH  
Am Borsigturm 9  
Berlin  
melanie.hinz@mayato.com

### **Zusammenfassung**

Metadaten sind seit Version 9 ein wichtiger Bestandteil von SAS. Neben der Management Console oder dem Metadatenbrowser im Window Environment kann man Metadaten unter anderem durch SAS Base Code abfragen oder verändern.

Einige Wege der Zugriffe auf Metadaten werden aufgezeigt. Es werden Beispiele vorgestellt, in denen Metadaten nur mittels SAS Base Code bearbeitet werden.

**Schlüsselwörter:** Metadaten, Reporting, XML, PROC METADATA

## **1 Einleitung**

Metadaten oder Metainformationen sind Daten, die Informationen über Merkmale anderer Daten enthalten, aber nicht diese Daten selbst.

„Bei den durch Metadaten beschriebenen Daten handelt es sich oft um größere Datensammlungen wie Dokumente, Bücher, Datenbanken oder Dateien. So werden auch Angaben von Eigenschaften eines einzelnen Objektes (beispielsweise Personennamen) als dessen Metadaten bezeichnet.“ Quelle: Wikipedia

Metadaten sind das Gerüst in SAS, sie beschreiben Daten und auch die Zusammenhänge verschiedener Metadatenobjekte. Allerdings kann man nur diejenigen Metadaten sehen und auslesen, für die man die Berechtigung hat. Ein vorheriges Backup ist sinnvoll, wenn man die Metadaten nicht nur auslesen, sondern auch modifizieren möchte. Einen guten Einstieg in das Thema bieten [1] und [2]. In [3] erklärt Paul Homes Metadatenreporting mit PROC METADATA.

Metadaten sind strukturierte Daten, die den Ort und die Struktur von Daten beschreiben. Beispiele sind:

- Die Konfiguration verschiedener SAS Server (Workspace Server, Stored Process Server)
- Die Konfiguration verschiedener SAS Produkte (Enterprise Guide, Enterprise Miner)
- User, Gruppen und Rollen
- Zeiger auf von SAS verwendbare Daten (Excel-, CSV-Dateien)
- Spaltenattribute von SAS-Tabellen (Formate, Längen)

## 2 Wege, Metadaten zu verwalten

Ein erster Weg befindet sich in der SAS Management Console, mit dem Plugin Metadata Manager.

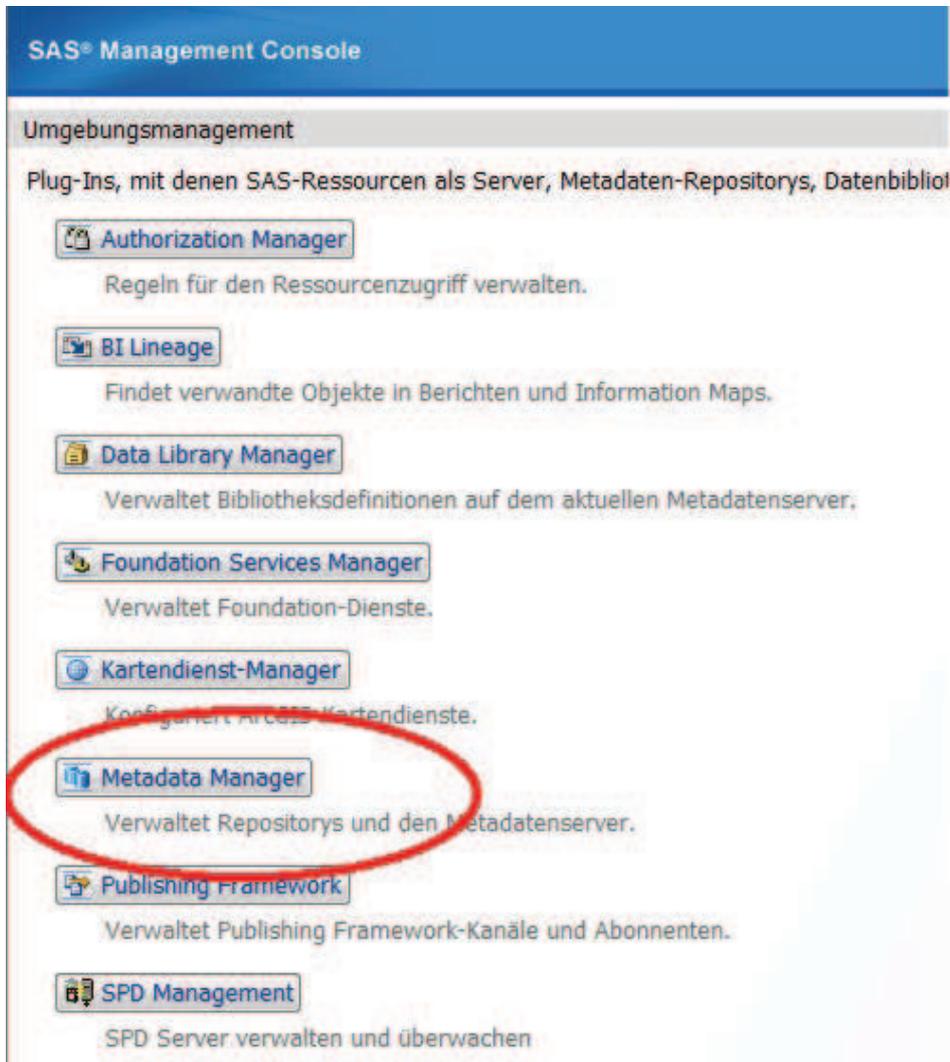
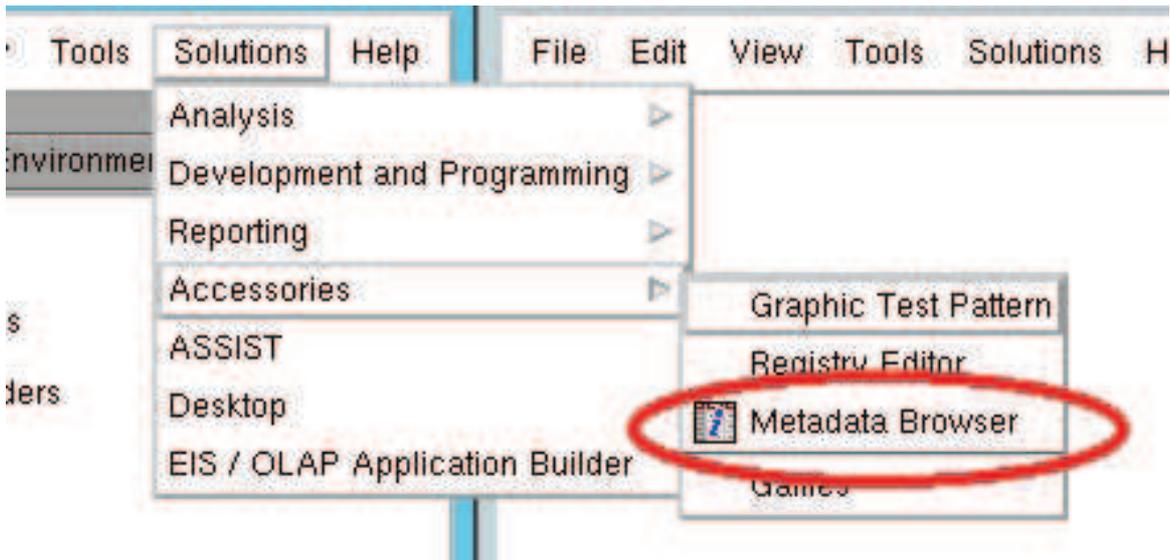


Abbildung 1: Metadata Manager in SAS Management Console

Um sich mit dem Metadatenserver zu verbinden, benötigt man einige Informationen, wie den Namen des Metadaten-servers, die Logindaten, sowie den Namen des Repositorys.

```
options metaserver='metadata-server-hostname'  
        metaport=8561  
        metauser='userid'  
        metapass='password'  
        metarepository='Foundation';
```

Diese Daten gibt man unter anderem beim Metadata Browser im SAS Window Environment ein.



**Abbildung 2:** Metadata Browser

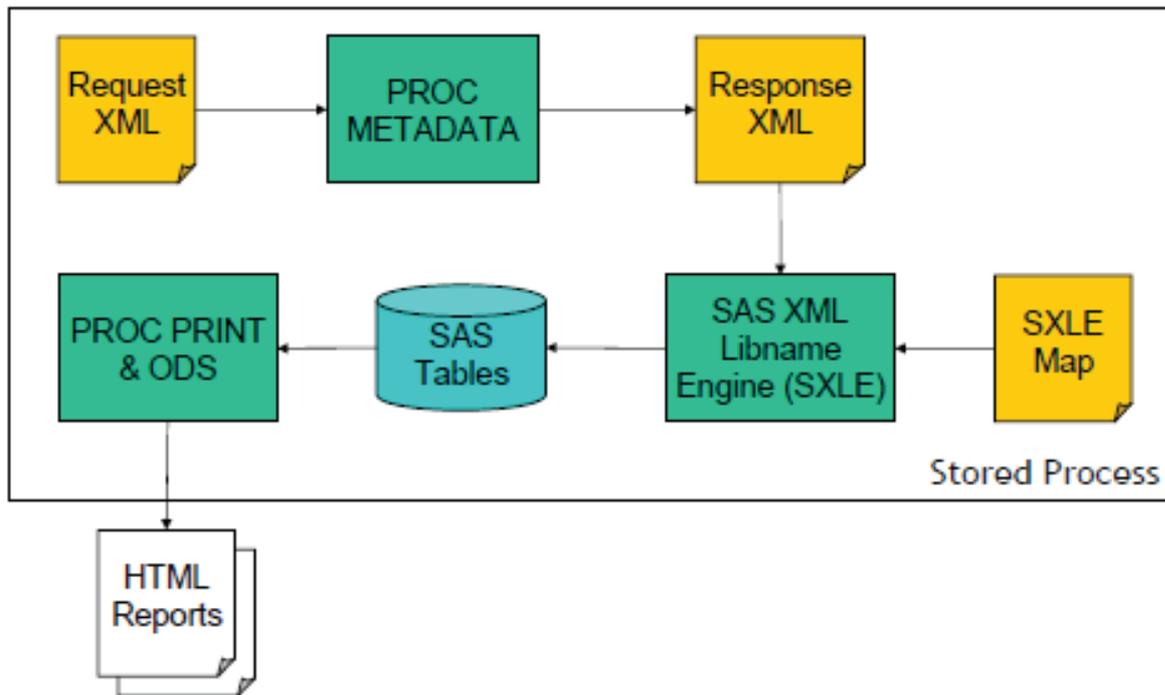
Dort sieht man auch das zentrale Element der Metadaten, den URI (Unique Resource Identifier). Jedes Metadatenobjekt in SAS ist somit einer eindeutigen Ziffernfolge zugeordnet.

 A screenshot of the SAS Metadata Browser window. The window title is '"Job" Metadatenobjekte'. It displays a table with the following columns: Name, Typ, Objekt-ID, Description, and Modified(GMT). The 'Objekt-ID' column contains unique identifiers, with the entry 'A5TUWPB9.C000001Z' circled in red.
 

Name	Typ	Objekt-ID	Description	Modified(GMT)
000_PROC_METALIB_Job		A5TUWPB9.C000000S		18. Dezember 2014 21.48 Uhr
AB_achtestprogramm (1	Job	A5TUWPB9.C000001X	Generated by S.	25. Februar 2015 18.49 Uhr
AB_neuntesprogramm i	Job	A5TUWPB9.C000001Z	Generated by S.	25. Februar 2015 18.49 Uhr
Bestuecken_classcopy	Job	A5TUWPB9.C000000B		18. Dezember 2014 21.48 Uhr
DateCheck	Job	A5TUWPB9.C000000T		18. Dezember 2014 21.48 Uhr
DESTATIS_Bevoelkeru	Job	A5TUWPB9.C000000Z		05. Januar 2015 10.13 Uhr
DIFT Organization Dim	Job	A5TUWPB9.C000002D		19. März 2015 09.02 Uhr
DIFT Populate Order F	Job	A5TUWPB9.C000002C		19. März 2015 15.33 Uhr
DM_STUDIE_2015_BA	Job	A5TUWPB9.C0000007		18. Dezember 2014 21.48 Uhr
DQ_CARRIAGE_RETU	Job	A5TUWPB9.C0000027		10. März 2015 16.58 Uhr
ECC_DATABASE_V2	Job	A5TUWPB9.C000000P		18. Dezember 2014 21.48 Uhr
ECC_DATABASE_V3	Job	A5TUWPB9.C000000Q		18. Dezember 2014 21.48 Uhr
ECC_DATABASE	Job	A5TUWPB9.C0000000		18. Dezember 2014 21.48 Uhr
extrakt_testen	Job	A5TUWPB9.C0000002		06. Februar 2015 15.19 Uhr

**Abbildung 3:** Metadata Browser mit URI

Diese URI kann man nun auch mittels PROC METADATA ansprechen. Dazu erzeugt man zuerst ein XML-Request.



**Abbildung 4:** XML mit SAS

Im folgenden Code wird eine Anfrage nach der ID und der Emailadresse eines Users gestellt:

```

data _null_;
  file request;
  put '<GetMetadataObjects>';
  put ' <Reposid>$METAREPOSITORY</Reposid>';
  put ' <Type>Person</Type>';
  put ' <Objects/>';
  put ' <NS>SAS</NS>';
  *Flags: + OMI_GET_METADATA(256) + OMI_XMLSELECT(128) +
  OMI_TEMPLATE(4) = 388;
  put ' <Flags>388</Flags>';
  put ' <Options>';
  put " <XMLSelect search=""Person[@Name =: 'Hinz']""/>";
  put ' <Templates>';
  put ' <Person Id="" Name="">';
  put ' <EmailAddresses />';
  put ' </Person>';
  put ' <Email Address="" />';
  put ' </Templates>';
  put ' </Options>';
  put ' </GetMetadataObjects>';
run;
  
```

Nun wird PROC METADATA ausgeführt:

```

proc metadata in=request out=response;
run;
  
```

Auf die Anfrage nach Namen und Emailadresse bekommt man die Antwort:

```
<GetMetadataObjects>
  <Reposid>A0000001.A5Y8I8TA</Reposid>
  <Type>Person</Type>
  <Objects>
    <Person Id="A5Y8I8TA.AR000006" Name="Melanie Hinz">
      <EmailAddresses>
        <Email Id="A5Y8I8TA.BT000001" Address="melanie.hinz@mayato.com"/>
      </EmailAddresses>
    </Person>
  </Objects>
<!-- original request deleted -->
</GetMetadataObjects>
```

Mit Hilfe einer XML-Map kann man diese nun nach SAS einlesen und in Tabellen und Berichten weiterverarbeiten.

### 3 SAS Data Steps

Um als Einstieg sämtliche Objektarten zu erhalten, kann man folgenden Code ausführen. Mit `metadata_getntyp` bekommt man dort die n-te Objektart ausgegeben.

```
data _null_;
  length type $64;
  do until (rc<0);
    i+1;
    rc=metadata_getntyp(i,type);
    if rc>0 then put type;
  end;
run;
```

Das Ergebnis ist umfangreich, hier nur ein paar Beispiele:

- Directory: Physischer Ordner
- ExternalTable: Externe Tabelle, z.B. Excel oder CSV
- Job: DI-Studio-Job
- PhysicalTable: SAS-Tabelle
- Transformation: Transformation in einem Job, z.B. Extract-Knoten
- Tree: Ordner im DI-Studio

An dem Beispiel einer Jobliste mit Pfaden sieht man nun, dass die Objekte verschiedenen Typs zusammenhängen, sowie, dass wir ihre Eigenschaften auslesen können.

Man gibt den Namen des Repositorys an und initialisiert die Variablen.

```
OPTIONS metarepository="Foundation";
DATA Joblist;
  FORMAT name $64.
         uri_job $32.
         modified $32.
         uri_tree tree_name uri_par path $255.
  ;
```

Mit einer Schleife über alle Objekte in der SAS-Bibliothek, die vom Typ Job sind, erhält man den URI jedes Jobs:

```
DO UNTIL (lrc lt 0);
  n+1;
  lrc = metadata_getnobj("omsobj:Job?@Name ? '", n, uri_job);
```

Zu jedem Job braucht man dessen Namen und wann er zuletzt geändert wurde:

```
IF lrc > 0 THEN DO;
  rc = metadata_getattr(uri_job, 'Name', name);
  rc = metadata_getattr(uri_job, "MetadataUpdated", modified);
```

Jeder Job liegt in einem Ordner, ist also mit ihm assoziiert. Dieser hat natürlich ebenfalls einen Namen:

```
rc = metadata_getnasn(uri_job, "Trees", 1, uri_tree);
rc = metadata_getattr(uri_tree, 'Name', tree_name);
```

Indem man jeweils den übergeordneten Ordner bestimmt, kann man den kompletten Pfad aufbauen.

```
urc = metadata_getnasn(uri_tree, "ParentTree", 1, uri_par);
path="/"||strip(tree_name);
do while (urc>0);
  uri_tree=uri_par;
  rc = metadata_getattr(uri_tree, 'Name', tree_name);
  urc = metadata_getnasn(uri_tree, "ParentTree", 1, uri_par);
  path="/"||strip(tree_name)||path;
end;
OUTPUT;
END; * if lrc...;
END; * do until...;
KEEP name uri_job modified path;
run;
```

Führt man diesen Code aus, zum Beispiel in dem User Written Code-Knoten in einem DI-Studio-Job, sieht das Ergebnis wie folgt aus:

#	name	uri_job	modified	publictype	path
5	test12	OMSOBJ:Job\ASTUWPB9.C0000029	19Mar2015:14:38:51	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
6	DQ_CARRIAGE_RETURN	OMSOBJ:Job\ASTUWPB9.C0000027	10Mar2015:16:58:28	Job	/Projects/mayato/Data Integration/020 Jobs/DQCar
7	vergleichen	OMSOBJ:Job\ASTUWPB9.C0000025	18Mar2015:13:21:42	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
8	_Projects_mayato_CI_Coa...	OMSOBJ:Job\ASTUWPB9.C0000024	05Mar2015:13:21:00		/Projects
9	siebtesprogramm (Generat...	OMSOBJ:Job\ASTUWPB9.C0000023	03Mar2015:13:59:24	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
10	siebtesprogramm (Generat...	OMSOBJ:Job\ASTUWPB9.C0000022	03Mar2015:13:51:49	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
11	AB_neuntesprogramm (Ge...	OMSOBJ:Job\ASTUWPB9.C000001Z	25Feb2015:18:49:52	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
12	AB_achtstesprogramm (Gen...	OMSOBJ:Job\ASTUWPB9.C000001X	25Feb2015:18:49:33	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
13	propagation_job	OMSOBJ:Job\ASTUWPB9.C000001U	17Mar2015:11:47:05	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
14	testneu_mitschleife	OMSOBJ:Job\ASTUWPB9.C000001S	23Feb2015:16:57:59	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
15	scd_ausprobieren	OMSOBJ:Job\ASTUWPB9.C0000016	19Mar2015:16:29:17	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
16	test11	OMSOBJ:Job\ASTUWPB9.C0000015	12Mar2015:18:01:22	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
17	test10	OMSOBJ:Job\ASTUWPB9.C0000014	13Feb2015:17:15:44	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
18	test8	OMSOBJ:Job\ASTUWPB9.C0000012	13Mar2015:14:43:23	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
19	test7	OMSOBJ:Job\ASTUWPB9.C0000011	09Mar2015:18:26:36	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
20	zusammenfassung_schleife...	OMSOBJ:Job\ASTUWPB9.C0000010	16Mar2015:14:22:58	Job	/Projects/mayato/Data Integration/020 Jobs/Kompa
21	DESTATIS_Bevoelkerung...	OMSOBJ:Job\ASTUWPB9.C000000Z	05Jan2015:10:13:44	Job	/Projects/mayato/Data Integration/020 Jobs/VA_Wc
22	TITANIC	OMSOBJ:Job\ASTUWPB9.C000000Y	18Dec2014:21:48:07	Job	/Projects/mayato
23	STAT_UNFAELLE_LAND_K...	OMSOBJ:Job\ASTUWPB9.C000000X	18Dec2014:21:48:07	Job	/Projects/mayato/Data Integration/020 Jobs/Statisti
24	STAT_BEVOELK_LAND_NA...	OMSOBJ:Job\ASTUWPB9.C000000...	18Dec2014:21:48:07	Job	/Projects/mayato/Data Integration/020 Jobs/Statisti
25	STAT_BEVOELK_LAND_GE...	OMSOBJ:Job\ASTUWPB9.C000000V	18Dec2014:21:48:07	Job	/Projects/mayato/Data Integration/020 Jobs/Statisti
26	PROC_METALIB_JOB	OMSOBJ:Job\ASTUWPB9.C000000U	18Dec2014:21:48:07	Job	/Projects/mayato/Data Integration/020 Jobs/Metalib

Abbildung 5: DI-Studio-Job mit Jobliste

## 4 Weitere Anwendungsmöglichkeiten

Den umgekehrten Weg kann man ebenso gehen. Man startet mit allen Ordnern und holt sich die Inhalte. Das können nicht nur Jobs sein, sondern auch andere Objekte, wie Tabellen, Notizen, registrierte Dokumente, usw..

Des Weiteren gibt es Security Macros von SAS, mit denen man Access Control Settings auswerten kann. Ein Beispiel ist, man lässt sich zu einem Ordner die Berechtigungen auf einem bestimmten Ordner anzeigen. Details dazu findet man auf der SAS Homepage.

### Literatur

- [1] E. Muriel, Exploring the Metadata Family Tree, SAS Global Forum 2009, Paper 097--2009.
- [2] E. Muriel, P. Simkin, Metadata for SAS 9 Programmers, SAS Global Forum 2008, Paper 134--2008.
- [3] Paul Homes, Metadata reporting with SAS software, TekEds.com