

SAS® GLOBAL FORUM 2015

The Journey Is Yours

Automating Your Metadata Reporting

Darryl Prebble, Prebble Consulting
Rupinder Dhillon, Bell Canada

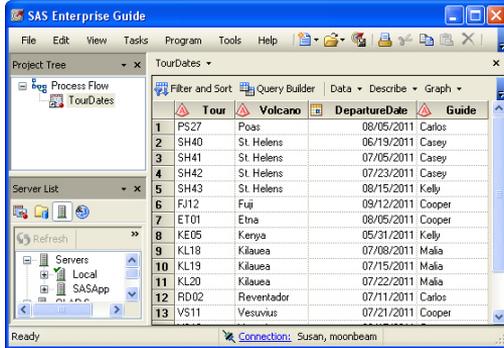


Agenda

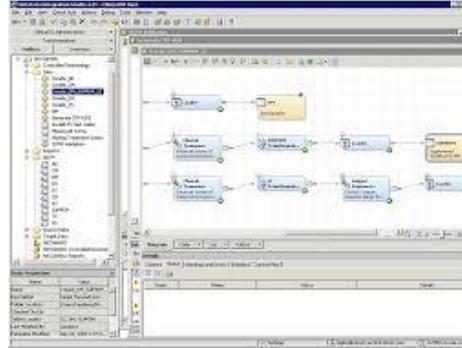
- What is metadata and why is it important
- Need for automated metadata reporting
- Pulling and Interpreting Metadata
- Making sense of XML
- Useful applications and challenges

What is Metadata?

Think about your current SAS deployment



	Tour	Volcano	DepartureDate	Guide
1	PS27	Posas	08/05/2011	Carlos
2	SH40	St Helens	06/19/2011	Casey
3	SH41	St Helens	07/05/2011	Casey
4	SH42	St Helens	07/23/2011	Casey
5	SH43	St Helens	08/15/2011	Kelly
6	FJ12	Fuji	09/12/2011	Cooper
7	ET01	Etna	08/05/2011	Cooper
8	KE05	Kerya	05/31/2011	Kelly
9	KL18	Kilauea	07/08/2011	Maka
10	KL19	Kilauea	07/15/2011	Maka
11	KL20	Kilauea	07/22/2011	Maka
12	RD02	Reventador	07/11/2011	Carlos
13	VS11	Vesuvius	07/21/2011	Cooper



Credentials Required

SAS Metadata Repository
sas92cloud

User:

Password:

OK Cancel

Almost everything generates some form of Metadata which is then stored in the metadata server

SAS Metadata 101

- Tells us
 - Who are your users and what can they access?
 - Where is your data and how can you connect?
 - What SAS processes exist and what data do they use?
 - Who created my SAS processes and how can I run them?

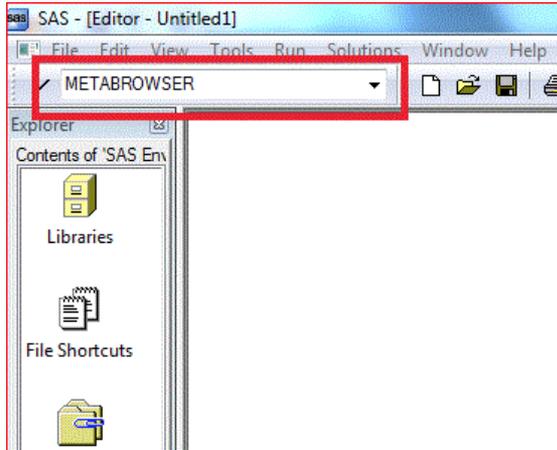
This “data about data” is stored in repositories within the SAS Metadata server. There is a wealth of information about your SAS system here; the challenge is how to get that information out.

Initial Challenge: What We wanted to know...

- Inventory of SAS processes and assets to help get us organized for our upcoming migration to 9.4
 - What jobs we had and what flows they belonged to?
 - Who last updated a job?
 - What job updated which table?

Before we start: Using the Metadata Browser

- Unfortunately, metadata is not stored in a simple table format. To browse and narrow down what you're looking for, you need to use the metadata browser.



- Open up PC-SAS and type "METABROWSER" into the command line
- Provide server info and credentials

Before we start: Using the Metadata Browser

- Navigate to the metadata objects that you're interested in. In our example, we want to look at Job and corresponding Jobflow information. We expand it until we see what we're looking for...



Before we start: Using the Metadata Browser

The screenshot displays the SAS Metadata Browser interface. The main window is titled "SAS - [Metadata Browser]" and features a menu bar (File, Edit, View, Tools, Solutions, Window, Help) and a toolbar. The interface is divided into three main sections:

- Explorer:** Shows the "Contents of 'SAS'" with categories like Libraries, File Shortcuts, Favorite Folders, and Computer.
- SAS Environment:** A tree view showing the hierarchy of metadata objects. The selected object is "JF_TD_DLY_GM_6000_STG_LOAD" under "Jobs".
- Attributes and Associations of "JF_TD_DLY_GM_6000_STG_LOAD" JFJob:** A table listing various attributes and their values. The "MetadataCreated(GMT)" and "MetadataUpdated(GMT)" rows are highlighted with an orange box.

Name	Value
UsageVersion	2000000
Id	A5FPY8BO.CK000AFP
IsActive	1
IsHidden	0
IsUserDefined	0
MetadataCreated(GMT)	26Feb2015:17:39:10
MetadataUpdated(GMT)	17Apr2015:18:33:56
Name	JF_TD_DLY_GM_6000_STG_LOAD
PublicType	DeployedFlow
TransformRole	SCHEDULER_FLOW
ComputeLocations	Association
JobActivities	Association
Properties	Association
ResponsibleParties	Association
Trees	Association

Building an XML Pull of the Metadata

Location of the resulting XML

```
filename outxml "/data/mi/data/outbox/cntl_dis_meta/job details report.xml";
```

```
proc metadata
```

```
in='
```

```
<GetMetadataObjects>
```

```
<Reposid>$METAREPOSITORY</Reposid>
```

```
<Type>Job</Type>
```

```
<Objects/>
```

```
<NS>SAS</NS>
```

```
<Flags>388</Flags>
```

```
<Options>
```

```
<Templates>
```

```
<<<<<< The GUTS >>>>>>>
```

The good stuff

```
</Templates>
```

```
</Options>
```

```
</GetMetadataObjects>'
```

```
out=outxml
```

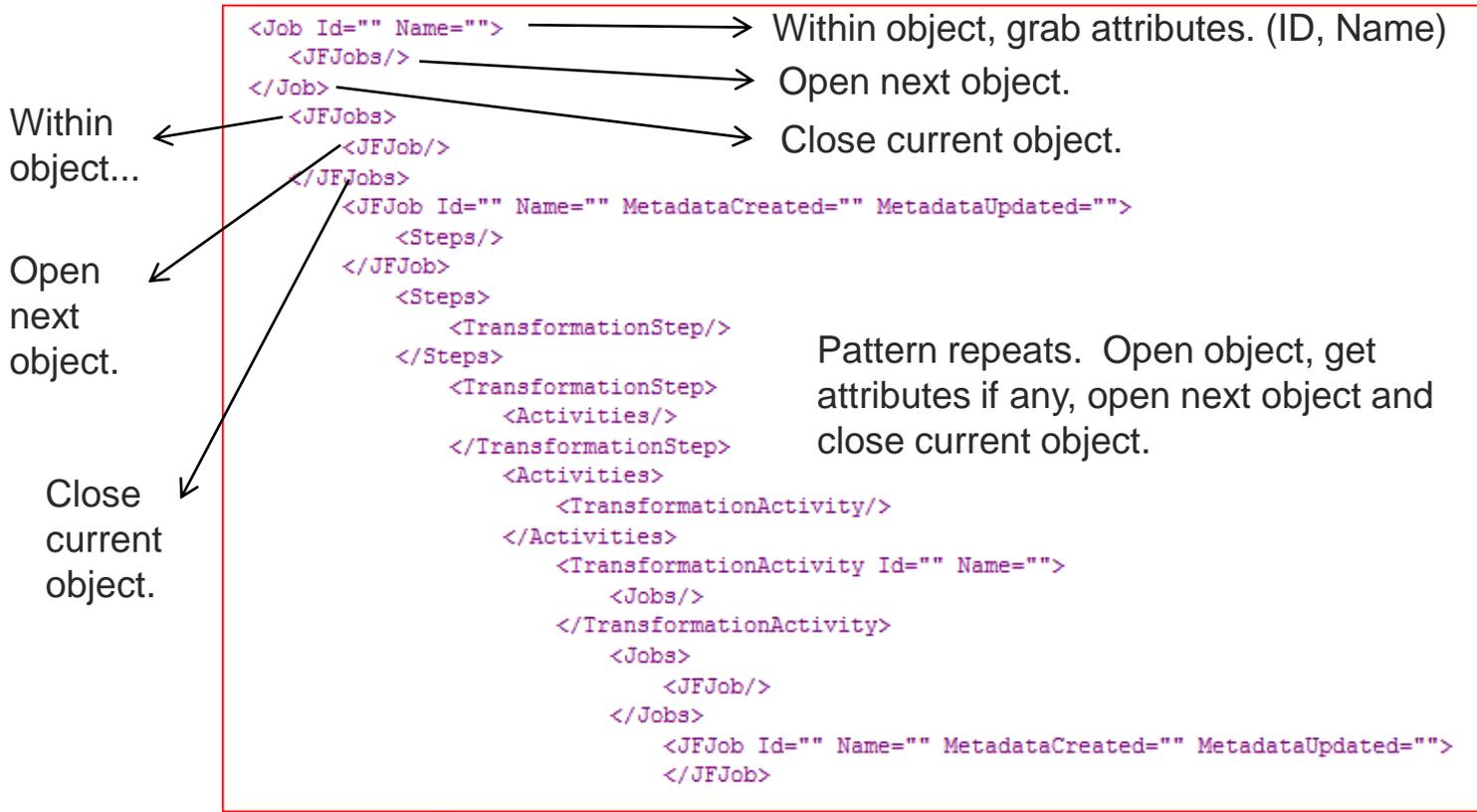
```
;
```

```
run;
```

Generic

Generic

Building an XML Pull of the Metadata



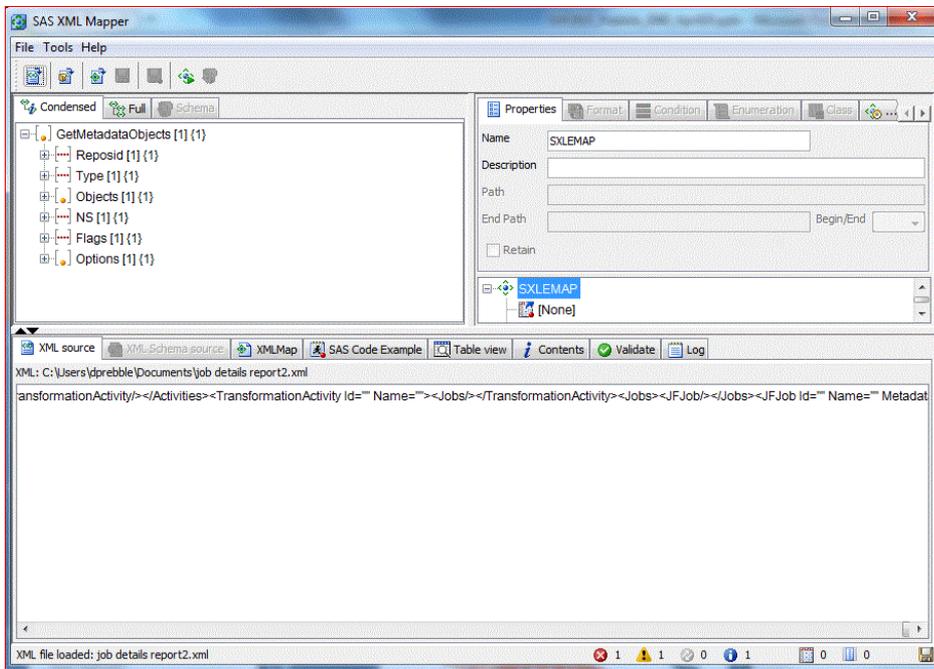
Let's see what we pulled...

```
/users/dprebble/test.xml - QAT
Encoding
<GetMetadataObjects><Reposid>A0000001.A5FPY8BO/<Reposid><Type>Job</Type><Objects><Job Id="A5FPY8B
_OpenExtract_MIDATA_BATCH"><JFJobs><JFJob Id="A5FPY8BO.CK0003G6" Name="J_TD_BART_DLY_1000_OpenExt.
Id="A5FPY8BO.CK0003GJ" Name="J_TD_BART_DLY_1003_LOAD_LND_BART_SUB_STATUS_TL" MetadataCreated="10J
MetadataCreated="10Jan2015:13:37:33" MetadataUpdated="02Feb2015:15:21:48"/></JFJobs></Job><Job Id
Name="J_HEAT_5025_Load_ODS_STATES_I" MetadataCreated="10Jan2015:13:37:49" MetadataUpdated="02Feb2
Name="J_Prem_SMS_Aggregator_Mthly"><JFJobs><JFJob Id="A5FPY8BO.CK00030A" Name="J_Prem_SMS_Aggrega
Name="J_TD_MLY_SMS_USE_2000_SMS_USAGE_OPEN_BATCH" MetadataCreated="11Jan2015:14:11:23" MetadataUp
MetadataCreated="11Jan2015:12:40:45" MetadataUpdated="13Jan2015:08:08:54"/></JFJobs></Job><Job Id
MetadataUpdated="11Jan2015:12:45:03"/></JFJobs></Job><Job Id="A5FPY8BO.C00005SH" Name="J_BEX_1110
Id="A5FPY8BO.C00005SZ" Name="J_BEX_1009_Load_ODS_BEX_RECOVERY_TIX"><JFJobs><JFJob Id="A5FPY8BO.CK
MetadataCreated="11Jan2015:12:19:26" MetadataUpdated="19Jan2015:04:26:21"/></JFJobs></Job><Job Id
MetadataCreated="11Jan2015:12:27:18" MetadataUpdated="11Jan2015:12:27:18"/></JFJobs></Job><Job Id
_BM_SHR_T_OB_FOLLOW_UP_D" MetadataCreated="11Jan2015:12:16:10" MetadataUpdated="19Jan2015:04:29:4
</JFJobs></Job><Job Id="A5FPY8BO.C00005UT" Name="J_TD_STG_1221_VMC_Subscriber_Contact_D_COPY"><JF
MetadataUpdated="19Jan2015:05:22:12"/></JFJobs></Job><Job Id="A5FPY8BO.C00005VO" Name="J_TD_DATA
Name="J_TD_LND_1201_VMC_Sub_Daily_D"><JFJobs><JFJob Id="A5FPY8BO.CK0004PG" Name="J_TD_LND_1201_VM
MetadataUpdated="11Jan2015:14:03:11"/></JFJobs></Job><Job Id="A5FPY8BO.C00005WK" Name="J_TD_STG_1
Id="A5FPY8BO.C00005WY" Name="J_TD_VMC_1200_OpenExtractBatch"><JFJobs></Job><Job Id="A5FPY8BO.C00
_Load_BRS_I_ACUT_CPE_D"><JFJobs><JFJob Id="A5FPY8BO.CK0003JU" Name="J_TD_ACUT_DLY_4204_Load_BRS_I
MetadataUpdated="11Jan2015:12:45:03"/></JFJobs></Job><Job Id="A5FPY8BO.C00005XS" Name="J_TD_ACUT_I
Id="A5FPY8BO.CK0003LH" Name="J_MI_4107_Load_STG_ACUT_QLOG_HIST_D" MetadataCreated="11Jan2015:12:4
_Load_STG_ACUT_ADMIN_LOV_LISTS_I" MetadataCreated="11Jan2015:12:51:43" MetadataUpdated="01Mar2015
Id="A5FPY8BO.C00005YZ" Name="J_MI_4134_Load_STG_ACUT_ACTPAT_TRK_I"><JFJobs><JFJob Id="A5FPY8BO.CK0
_Load_BY_F_COPYING_ACTPAT_HIST"><JFJobs><JFJob Id="A5FPY8BO.CK0003ZL" Name="J_MI_HYDR000_Load
```

- In order to make sense of this XML, we need to build an XML map to tell SAS how to read it



Building an XML map using SAS XML Mapper



- File → Open XML
- Navigate to file you created in the previous step
- Click import
- Once imported, create and save a map so we can automate the process
 - Tools → Automap using XML
 - File → Save XMLMap

Pulling it all together

- Now we can use Enterprise Guide to read the XML file using our XML map

```
libname outxml xml
  '/data/mi/data/outbox/cntl_dis_meta/job details report.xml'
  xmlmap='/data/mi/data/outbox/cntl_dis_meta/job details report.map'
;
```

```
proc sql;

  * Read in XML data (only 5 connections available at a time otherwise) ;
  create table job                as select * from outxml.job;
  create table jfjobs             as select * from outxml.jfjobs;
  create table jfjob              as select * from outxml.jfjob;
  create table steps              as select * from outxml.steps;
  create table transformationstep as select * from outxml.transformationstep;
  create table activities         as select * from outxml.activities;
  create table transformationactivity as select * from outxml.transformationactivity;
  create table jobs               as select * from outxml.jobs;
  create table jfjob1             as select * from outxml.jfjob1;
```

Tip: Create all tables in WORK first, as only 5 connections allowed at a time.

- Output is created in a format that forces you to join all object tables back together
- Each table has a primary key and foreign key
- Join one table to next until you have a master table

Pulling it all together

```
proc sql;
  create table jobflow_details as
  select jf1.id                label='' as JF_METADATA_ID
        ,jf1.name              label='' as JF_NAME
        ,jf1.metadatacreated format=datetime26.    label='' as JF_METADATACREATED_DTTM
        ,jf1.metadataupdated format=datetime26.    label='' as JF_METADATAUPDATED_DTTM
        ,job.id                label='' as JOB_METADATA_ID

  from job                    job

  left join jfjobs            jfs
    on(
      jfs.job_ordinal          = job.job_ordinal
    )

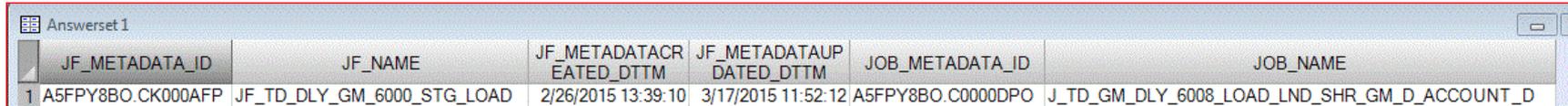
  left join jfjob             jf
    on(
      jf.jfjobs_ordinal        = jfs.jfjobs_ordinal
    )

  left join steps             st
    on(
      st.jfjob_ordinal         = jf.jfjob_ordinal
    )

  left join transformationstep ts
```

As shown here, each object is turned into it's own table which you have to join together one by one...

Pulling it all together



JF_METADATA_ID	JF_NAME	JF_METADATACREATED_DTTM	JF_METADATAUPDATED_DTTM	JOB_METADATA_ID	JOB_NAME	
1	A5FPY8BO.CK000AFP	JF_TD_DLY_GM_6000_STG_LOAD	2/26/2015 13:39:10	3/17/2015 11:52:12	A5FPY8BO.C0000DPO	J_TD_GM_DLY_6008_LOAD_LND_SHR_GM_D_ACCOUNT_D

- We've patched it all together into a single dataset which can then be formatted for reporting and scheduled
- We use this to build a tracking table on Teradata for auditability

Useful Applications

- Job Changes Report
- Job Flow/Job/Table Inventory
- Migration: Master Inventory
- Tracking Users and Groups
- Updating and Creating Metadata in bulk

Other ideas? Start with exploring Metadata Objects types

Our next Challenge

- Tracking active SAS usage using metadata and login information
 - Who are our active SAS users?
 - What are they using and how often?
 - What are our most valuable SAS assets?
 - How far reaching are our SAS assets across the organization?

Stay Tuned....

Questions?

- Contact:
 - Darryl Prebble: dprebble@gmail.com
 - Rupinder Dhillon: rupinder.dhillon@bell.ca

Session ID #2060





April 26-29
Dallas, TX

