



SESUG 2018

St. Pete Beach, FL
October 14-17

Paper 171

Taking XML's Measure: Using SAS® to Read In and Create XML for Analytic Use and Websites

Louise S. Hadden

Abt Associates Inc.

Presenter

Louise S. Hadden

Lead Programmer Analyst, Abt Associates Inc.

Louise Hadden has been using and loving SAS since the days of punch cards and computers the size of a tiny house. She spends most of her time in support of health policy analytics at Abt Associates Inc. and loves a good SAS reporting challenge. She is an ardent life long learner and reads voraciously, loves photography and volunteers at the MSPCA Boston Adoption Center walking, training and photographing dogs.



Introduction

XML has become a standard over the years for populating websites and transferring information. This presentation demonstrates how to parse mystery XML files, read in XML files that you can't right-click on, read into Microsoft Excel using SAS[®], how to use maps and schemas to input and output various XML representations, and how to construct and output “measure code” data sets from input data to maximize the flexibility of XML data representation and usage.



Introduction

XML

- HTML Markup Language used to build static web pages
- HTML5 Latest version of HTML with support for multimedia
- XML Extensible markup language
- XHTML XML that mirrors HTML in syntax and adds hypertext capability
- DHTML Dynamic HTML
- KML ML to display geographic data



Introduction

Markup Languages

Wikipedia tells us “Markup Language is a system for annotating a document in a way that is syntactically distinguishable from the text.”



SESUG 2018



St. Pete Beach, FL
October 14-17

Introduction

XMLtipsheet.pdf - Adobe Acrobat Reader DC

File Edit View Window Help

Home Tools

XMLtipsheet.pdf x



2 / 2



66.7%



SAS

http

eFA

SAS

also

bot

nplat

SAS[®]9 SAS[®]9.3 XMLV2 LIBNAME Engine Tip Sheet

LIBNAME Statement Syntax

LIBNAME *libref* XMLV2 < '*SAS-library* | *XML-document-path*' > <*options*>;

Required Arguments

libref

a valid SAS name to associate with the XML document. A libref cannot exceed eight characters.

XMLV2

the LIBNAME engine name. (Alias: XML92)

'*SAS-library* | *XML-document-path*'

the physical location of the XML document.

Options

FORMATACTIVE=NO|YES

INDENT=*integer*

XMLV2 Engine

XMLV2 accesses enhancements since SAS 9.1.3, which includes the ability to assign a libref to a SAS library in a directory-based environment, and enhanced XMLMap support. XML92 is supported as an alias.

XMLMap File

An XMLMap file is a separate XML document that is used by the XMLV2 engine to provide enhanced support for XML processing. You can manually code an XMLMap file with specific XMLMap element syntax, or you can generate an XMLMap file with the SAS XML Mapper application, which is the recommended method.

Why Do I Get Errors When Importing an XML Document? The engine imports only XML documents that conform to the markup types

XMLMap Syntax Version 2.1

Tip: Rather than manually code XMLMap syntax, it is recommended that you generate XMLMap syntax by using the SAS XML Mapper application.

SXLEMAP version=*number* name=*XMLMap*
description=*description*

NAMESPACES count=*number*
NS id=*number* <prefix=*name*>

OUTPUT

HEADING

ATTRIBUTE name=*name* value=*value*

TABLEREF name=*name*

TABLE name=*data-set-name*

TABLE-PATH syntax=*type*

TABLE-END-PATH syntax=*type*

beginend='BEGIN|END'

TABLE-DESCRIPTION

SESUG 2018



St. Pete Beach, FL
October 14-17

Introduction

Basic XML Concepts

- XML documents
 - Nodes
 - Relationships
- DTD
- XML maps or schemas
- XSL style sheets

SESUG 2018



St. Pete Beach, FL
October 14-17

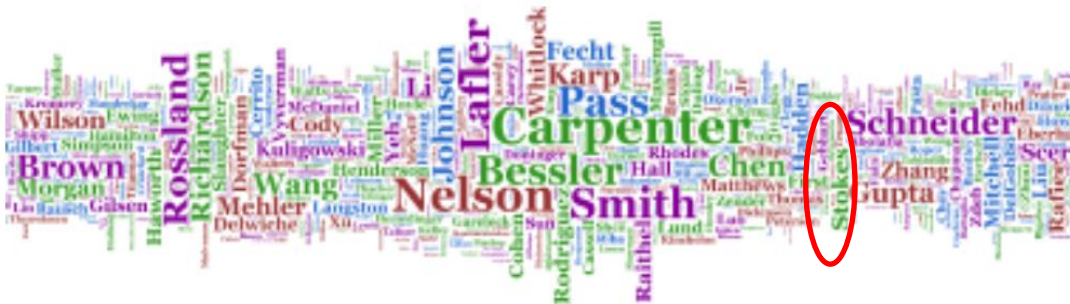
Methods for Reading XML in

IRL Example

← → ↻ Secure https://www.lexjansen.com/cgi-bin/saspapers_query.php ☆ a

Home
Contact
SAS Paper Search

SAS:MWSUG papers
SAS:NESUG papers
SAS:PharmaSUG papers
SAS:PharmaSUG China papers
SAS:PhUSE papers
FDA/PhUSE CSS posters
SAS:PNWSUG papers



Search **30815** SAS proceedings from SAS Global Forum, SUGI, PharmaSUG, NESUG, SESUG, WUSS, MWSUG, PNWSUG and SCSUG.
This search is based on title and author. You need to specify at least 3 characters in one field.
Searches are case insensitive and can be done in one of 3 ways:

- A literal string within double quotes. Examples: "dynamic data exchange" or "dde "
- Multiple words separated by a plus (+). The results must contain all words. Example: cl metadata
- Multiple words separated by blanks. The results must contain at least one of the words. report tabulate

TITLE contains

AND **AUTHOR** contains

SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML in

IRL Example

UG

G papers

G papers

papers

F/SUGI

papers

papers

papers

ords

TITLE contains

AND **AUTHOR** contains Louise+Hadden

Search Sort by: ☒ Date ☐ Title

Maximum number of results return: **600**.

Returned **106** result(s) sorted by month DESC, title.

[Experimental new feature] Save all search results as: [XML](#) [JSON](#) ([Suggest other formats](#)).

Filter

[1] **Document and Enhance Your SAS® Code, Data Sets, and Catalogs with SAS Functions, Macros, and SAS Metadata**

Louise Hadden; Roberta Glass

SESUG 2017

Keywords: macro,

<http://www.lexjansen.com/sesug/2017/PSA-73.pdf>

[2] **Methods for Creating Sparklines using SAS®**

Rick Andrews · Louise Hadden · Robert Allison

SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML in

IRL Example

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <searchresults CreationDate="2018-04-02T15:03:20Z" SearchTitle="" SearchAuthors="Louise+Hadden"
3 <paper paperid="sesug2017.psa-73">
4   <title>Document and Enhance Your SAS® Code, Data Sets, and Catalogs with SAS Functions, Mac
5   <authors>Louise Hadden; Roberta Glass</authors>
6   <keywords>macro,</keywords>
7   <link>http://www.lexjansen.com/sesug/2017/PSA-73.pdf</link>
8   <conference>SESUG</conference>
9   <month>2017-11</month>
10 </paper>
11 <paper paperid="sesug2017.riv-42">
12   <title>Methods for Creating Sparklines using SAS®</title>
13   <authors>Rick Andrews; Louise Hadden; Robert Allison</authors>
14   <link>http://www.lexjansen.com/sesug/2017/RIV-42.pdf</link>
15   <conference>SESUG</conference>
16   <month>2017-11</month>
17 </paper>
```

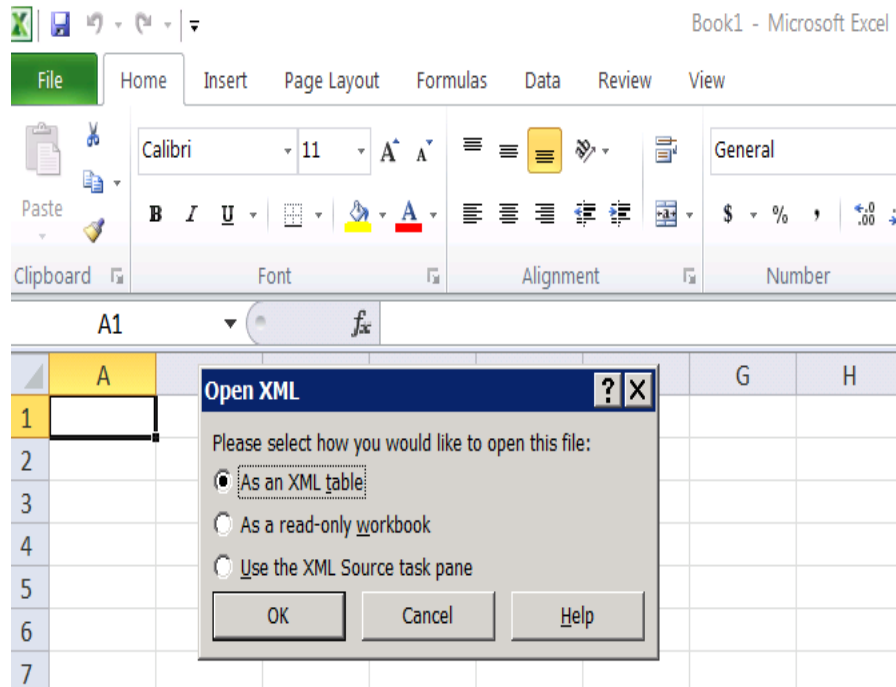
SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

Open Microsoft Excel and Open XML



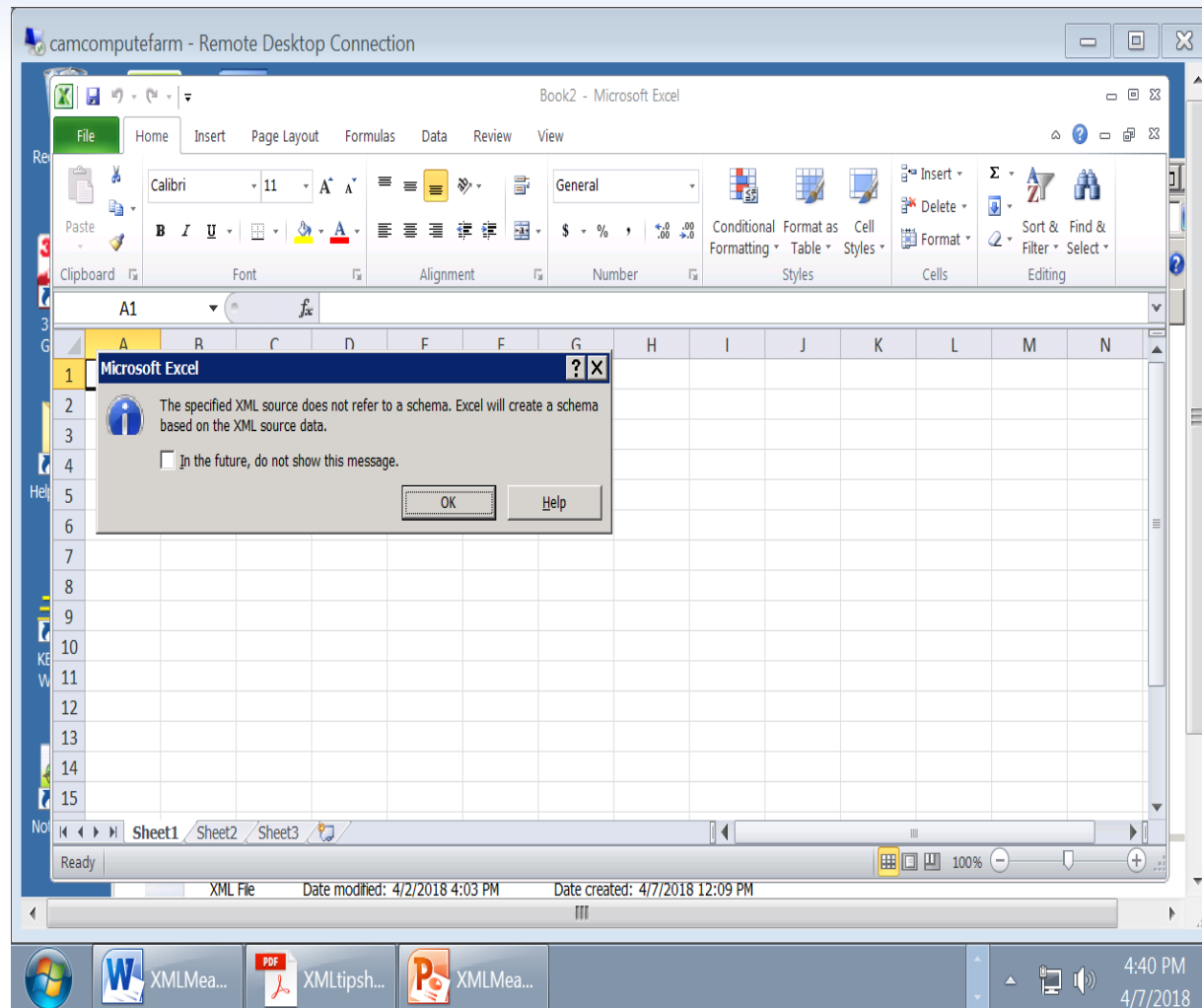
SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

Open Microsoft Excel and Open XML



SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

Open Microsoft Excel and Open XML

	A	B	C	D	E	F	G	H	I	J
1	paper_ORDI	paper_paperid	title	Comment	TopicArea	Keywords	authors	link	conferenc	month
2	1	sesug2017.psa-73	Document and Enhance Your SAS® Code, Data Sets, and Catalog macro,				Louise Hac	http://ww	SESUG	2017-11
3	2	sesug2017.riv-42	Methods for Creating Sparklines using SAS®				Rick Andre	http://ww	SESUG	2017-11
4	3	sesug2017.epo-91	Purrfectly Fabulous Feline Functions				Louise Hac	http://ww	SESUG	2017-11
5	4	sesug2017.riv-92	SAS/GRAPH® and GfK Maps: a Subject Matter Expert Winning Combination				Louise Hac	http://ww	SESUG	2017-11
6	5	mwsug2017.tt08	Get Smart! Eliminate Kaos and Stay in Control - Creating a Complex Directo				Louise Hac	http://ww	MWSUG	2017-10
7	6	mwsug2017.rf07	PROC DOC III: Self-generating Codebooks Using SAS®				Louise Hac	http://ww	MWSUG	2017-10
8	7	mwsug2017.po01	Red Rover, Red Rover, Send Data Right Over: Exploring External Geographic				Louise Hac	http://ww	MWSUG	2017-10
9	8	scsug2017.lh05	Red Rover, Red Rover, Send Data Right Over: Exploring External Geographic				Louise S. F	http://ww	SCSUG	2017-10
10	9	mwsug2017.po02	SAS/GRAPH® and GfK Maps: a Subject Matter Expert Winning Combination				Louise Hac	http://ww	MWSUG	2017-10
11	10	wuss2017.043.dm	Get Smart! Eliminate Kaos and Stay in Control Creating a Complex Director				Louise Hac	http://ww	WUSS	2017-09
12	11	wuss2017.042.dp	PROC DOC III: Self-Generating Codebooks Using SAS®				Louise Hac	http://ww	WUSS	2017-09
13	12	wuss2017.051.dp	Red Rover, Red Rover, Send Data Right Over: Exploring External Geographic				Louise Hac	http://ww	WUSS	2017-09
14	13	wuss2017.050.dp	SAS/GRAPH® and GfK Maps: a Subject Matter Expert Winning Co GfK, SAS/G				Louise Hac	http://ww	WUSS	2017-09
15	14	pharmasug2017.qi	PROC DOC III: Self-generating Codebooks Using SAS®				Louise Hac	http://ww	PharmaSU	2017-05

SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

Read In Using Automap

```
filename ajw '.\AlanWhite.xml';  
filename map '.\Map\AJW.map';  
libname ajw xmlv2  
automap=replace xmlmap=map;  
  
proc contents data=ajw._all_ ;  
run;
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

Read In Using Automap

```
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
combined4XMLOut.xml AlanWhite.xml AJW.map
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- ##### -->
3 <!-- 2018-04-07T12:21:39 -->
4 <!-- SAS XML Libname Engine Map -->
5 <!-- Generated by XML Mapper, 904000.0.0.20130522190000_v940 -->
6 <!-- ##### -->
7 <!-- ### Validation report ### -->
8 <!-- ##### -->
9 <!-- XMLMap validation completed successfully. -->
10 <!-- ##### -->
11 <SXLEMAP name="AUTO_GEN" version="2.1">
12
13   <NAMESPACES count="0"/>
14
15   <!-- ##### -->
16   <TABLE description="searchresults" name="searchresults">
17     <TABLE-PATH syntax="XPath">/searchresults</TABLE-PATH>
18
19     <COLUMN class="ORDINAL" name="searchresults_ORDINAL">
20       <INCREMENT-PATH beginend="BEGIN" syntax="XPath">/searchresults</INCREMENT-PATH>
21       <TYPE>numeric</TYPE>
22       <DATATYPE>integer</DATATYPE>
23     </COLUMN>
```

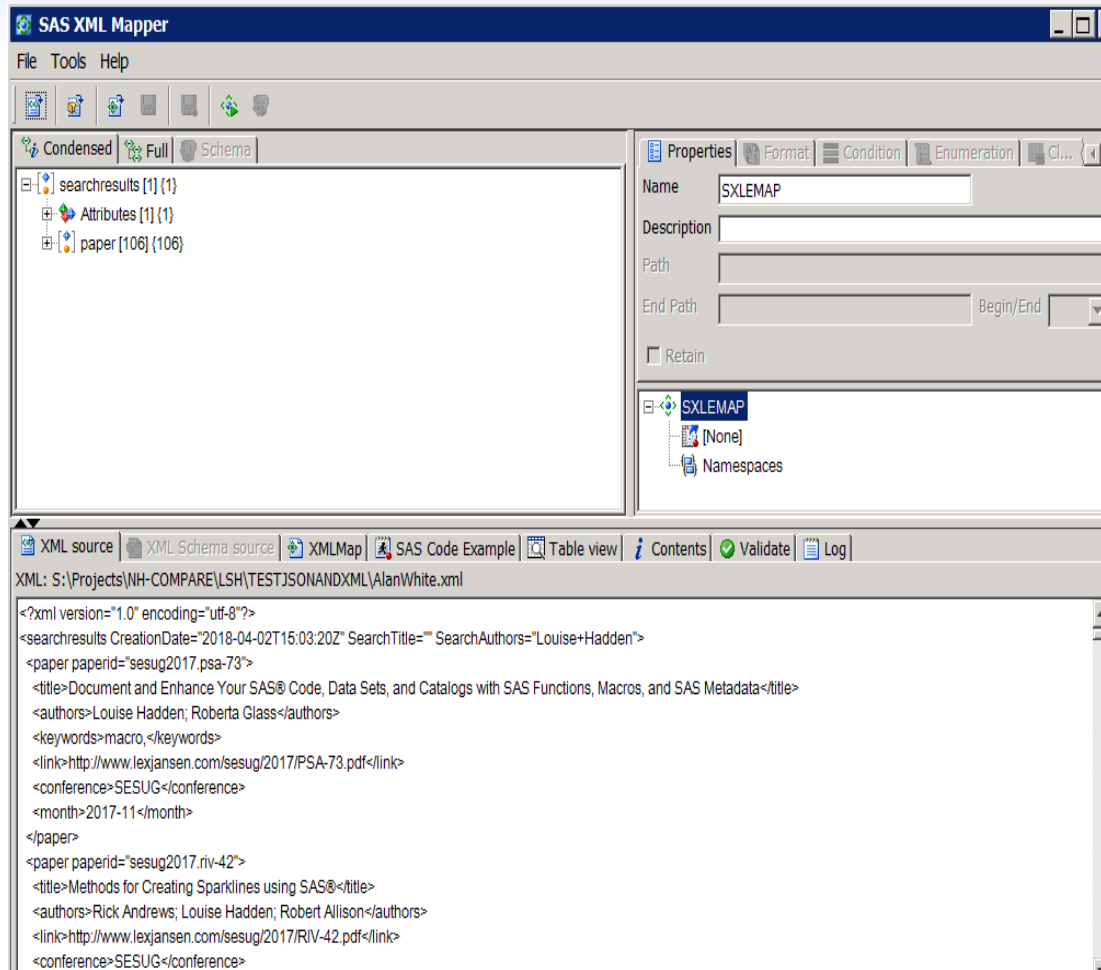
SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

SAS XML Mapper



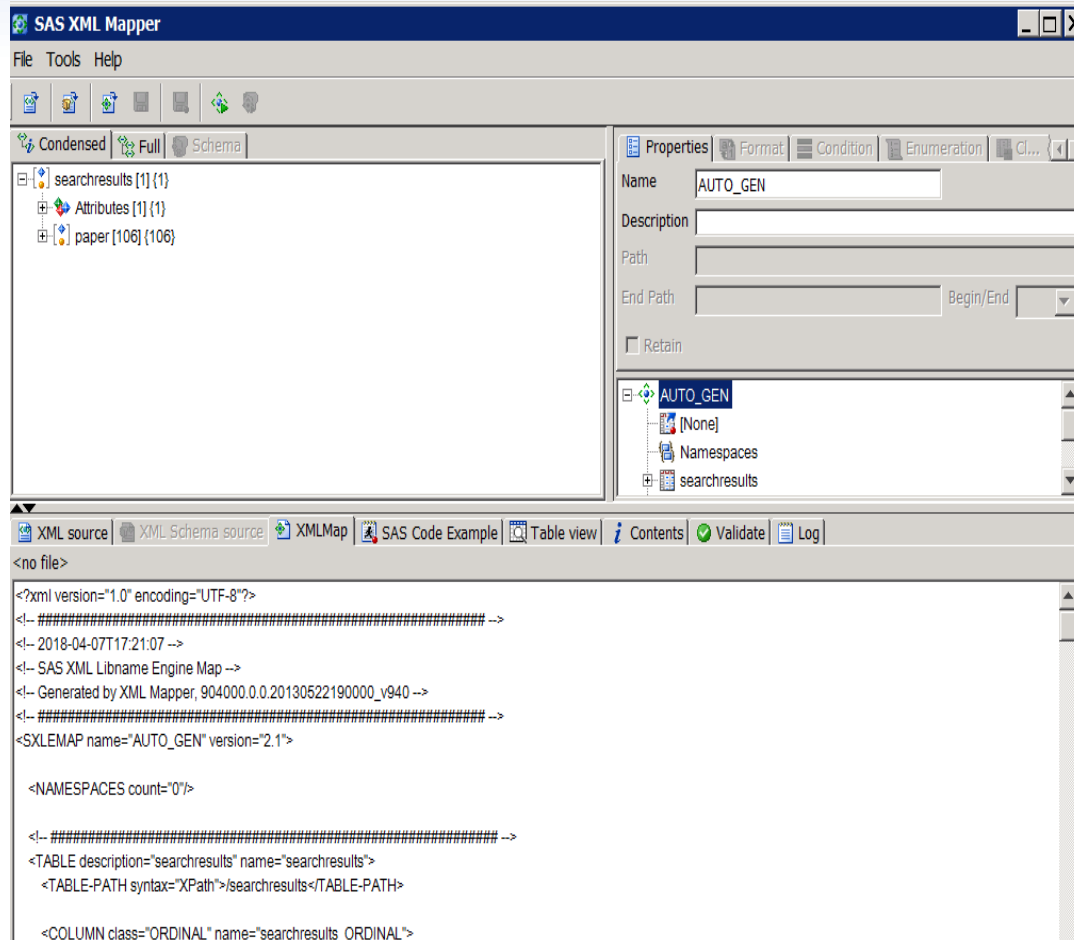
SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

SAS XML Mapper



SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML In

Read In Using an Existing Map

```
filename ajw '.\AlanWhite.xml';  
filename map '.\Map\AJW.map';  
libname ajw xmlv2 xmlmap=map;
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Capitalizing on the “Extensible”

Case Study: Nursing Home Compare

Find a nursing home

Nursing Home Compare has detailed information about every **Medicare** and **Medicaid**-certified nursing home in the country. A nursing home is a place for people who can't be cared for at home and need 24-hour nursing care.

Search below to find nursing homes based on a location and compare the quality of care they provide and their staffing.

A field with an asterisk (*) is required.

*** Location**

Example: 45802 or Lima, OH or Ohio

Nursing home name (optional)

Search



Spotlight

- ◆ Get updated [state website information](#), including the ability to electronically file a complaint against a nursing home in some states
- ◆ Use these helpful resources when looking for a nursing home:
 - ◆ [Guide to choosing a nursing home](#)
 - ◆ [Nursing Home Checklist](#)

Tools and Tips

- ◆ First time here?
 - ◆ Visit [About Nursing Home Compare](#) and the [Resources](#) section to learn more about the site and available resources.
- ◆ [Learn about Medicare coverage of skilled nursing facility care, and swing bed services.](#)
- ◆ [Get help filing a nursing home](#)

Additional Information

- ◆ **Nursing Home Compare data last updated:** August 23, 2017 (Data are updated on or about the fourth Wednesday of the month).
- ◆ [Download the database](#)
- ◆ [Learn how we calculate ratings](#)
- ◆ For nursing homes: [Update your address, phone number and other administrative data.](#)

SESUG 2018



St. Pete Beach, FL
October 14-17

Capitalizing on the “Extensible”

Concepts

- Many files from many different sources go into the Nursing Home XML
- Original files were at different levels, for example nursing home residents versus providers
- Thousands of elements or nodes



Capitalizing on the “Extensible”

XML Output

```
ODS MARKUP BODY=TEST.XML;
```

```
PROC PRINT DATA=TEST;
```

```
RUN;
```

```
ODS MARKUP CLOSE;
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML in

Creating Measures

```
data msr_ownership (keep=provnum msr_cd  
value occurrence ftnt filedate);  
    length provnum $ 6 msr_cd $ 20  
    value $ 120 ftnt $ 12 occurrence $ 3  
    filedate $ 8;  
    set dd.owner_ocr;  
    . . .  
    msr_cd = 'ASSOCDATE';  
    value = assoc_date_text;  
    if value ne ' ' then output ;  
Run;
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Methods for Reading XML in

Creating Measures

```
proc sql ;  
create table  
out.msr_Owners_&fileyear.&filedate. as  
    select PROVNUM as PID  
        ,MSR_CD as MCD  
        ,occurrence as OCR  
        ,VALUE as SV  
        ,ftnt as FN  
        ,"Text" as ST  
from msr_ownership  
order by PID, MCD, OCR;  
quit;
```



Methods for Reading XML in

Creating Measures

PID 015009

MCD ASSOCDATE

OCR 1

SV since 09/01/1969

FN

ST Text

SESUG 2018



St. Pete Beach, FL
October 14-17

Capitalizing on the “Extensible”

XML Output

As with reading XML output in, the SAS XML libname engine and a schema or map are employed.

```
filename mapt  
".\map\MapTemplate.map";  
filename map  
".\map\MapModified.map";  
libname temp1 xml92  
xmltype=xmlmap xmlmap= map;
```



Capitalizing on the “Extensible”

XML Output

As with reading XML output in, the SAS XML libname engine and a schema or map are employed.

```
filename map  
".\map\MapModified.map";  
libname temp1 xml92  
xmltype=xmlmap xmlmap= map;  
filename out  
".\XML\&outnm.XMLOut.xml";
```

Capitalizing on the “Extensible”

XML Output

```
combined4XMLOut.xml x AlanWhite.xml x AJW.map x
1 <?xml version="1.0" encoding="windows-1252" ?>
2 <!--
3 SAS XML Libname Engine (SAS92XML)
4 SAS XMLMap Generated Output
5 Version 9.04.01M1P12042013
6 Created 2018-04-06T19:55:49
7 -->
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Capitalizing on the “Extensible”

XML Output

```
<PMIDetails xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  
<HeaderInformation>  
  <DataSource>Abt - Nursing Home Compare - Combined File 4 of 10</DataSource>  
  <DateCreated>04/01/2018</DateCreated>  
  <PMICount>1225</PMICount>  
</HeaderInformation>
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Capitalizing on the “Extensible”

XML Output

```
] <PMI PID="175008">  
[ <M MCD="401">  
[ <S ST="Percentage">  
  <SV >14.7</SV>  
- </S>  
- </M>
```

SESUG 2018



St. Pete Beach, FL
October 14-17

Conclusion

SAS has provided many tools to both read XML into SAS and to output XML from SAS. The use of “measure code” transformation greatly extends the power and flexibility of XML generation from SAS.

“Wit beyond measure is man's greatest treasure.”

J.K. Rowling



Acknowledgements

The author wishes to thank Chevell Parker of SAS, a former colleague Fred Pratter, a current colleague Nancy McGarry, Troy Martin Hughes and Lex Jansen for mentoring and inspiring me.

Thanks also to the Nursing Home Compare project team for providing a welcome challenge to encourage finding new ways to improve our data processing and transfers, especially Christianna Williams of Abt Associates and Zach Sarver of CGI.



Contact Information

Your comments and questions are valued and encouraged. Contact the author at:

Louise S. Hadden

Abt Associates Inc.

617-349-2385

Louise_hadden@abtassoc.com

abtassociates.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

