Paper 013-2008

Automated Bulk Loading of Documentum® Using XML Control files Created with Base SAS®

John Booterbaugh, PharmaLogic, LLC, Sanatoga, PA
Terek Peterson, Shire, Wayne, PA
Kate Wilber, Shire, Basingstoke, Hampshire, UK

ABSTRACT

This presentation addresses the use of Base SAS to write XML code to automate bulk file loading into Documentum, a document management system (DMS). Documentum with FirstDoc® manages document attributes and has features such as check-in, check-out, version control, ownership, and access control. Even when using the bulk loading utility of Documentum with FirstDoc, the user must manually enter ample document metadata which can be monotonous and time consuming. Automating repetitive manual tasks are best handled with programming languages like SAS to eliminate tedious tasks. Benefits to the business include substantial timesaving and error reduction. Documentum with FirstDoc allows the use of XML control files to bulk load document objects. Base SAS can effectively be used to generate code in other languages like XML, especially when the other language has a consistent structure. This presentation will describe XML syntax requirements, outline the XML structure, and show the required document attributes to be processed via SAS code. Step by step instructions and example SAS code show how to control the mandatory keyword parameters and loop control in order to read a desired directory of objects. Concluding remarks will discuss implementation considerations and the advantages and disadvantages of this technique.

INTRODUCTION

According to *Guidance for Industry Providing Regulatory Submissions in Electronic Format — Human Pharmaceutical Product Applications and Related Submissions Using the eCTD Specifications* [1], "submissions are a collection of documents. A document is a collection of information that includes forms, reports, and datasets. When making an electronic submission, each document should be provided as a separate file. The documents, whether for a marketing application, an investigational application, or a related submission, should be organized based on the five modules in the CTD." Regulatory agencies, in particular FDA, require documents and data in the form of programs, patient profiles, SAS transport files, CRFs, signed informed consent documents, and tables/figures/listings, be included in an electronic submission. It is best to manage the numerous documents via a document management system like Documentum. This allows the sponsor the ability to respond to FDA requests promptly.

During a training session for our document management system, an individual described how it took one month to load 1200 PDF objects even using the bulk loading utility provided by Documentum with FirstDoc. It was mentioned that this task took about two hours a day due to the tedious nature of the assignment, which equaled about 45 total hours. In a separate, unrelated task over 2500 file objects were loaded thanks to bulk loading via XML control files in approximately the same amount of time. This was a fantastic way to load large quantity of files into Documentum because it works in the background and we were able to continue to do other work too. Calculated savings equate to approximately one third to one half the amount of time with a guaranteed level of quality and potential increase in timeliness. As this process is standardized, calculated savings could be substantially more.

In order to understand this concept, this paper will first show how files are loaded into the DMS via the bulk loader and what important fields need to be populated. Next, we show how an individual would launch a bulk loading session if a XML control file is created. Then we will show how to create a XML file with Base SAS. Finally, an example of the XML control file is presented and conclusions are made.

ABBREVIATIONS / DEFINITIONS

XML (eXtensible Markup Language) – A Language which supports multiple applications which can use the structured characteristics found in various documents. A document contains basic structured information such as words, and may contain other structure information such as graphs, pictures, tables, etc. XML identifies the structures in a document to be used by other applications.

DMS – Document Management Systems provide well-defined framework interfaces for storing and accessing data and documents.

eCTD – Electronic Common Technical Document

XML Control File – a XML file which contains a document’s structured information within tags that describe the document’s values and attributes.

Bulk Import File – a XML file used by an application to add similar documents or file objects to a Document database.

OBJECTIVE

The primary objective was to automate the bulk loading process into the DMS. Early pilots of the XML bulk loader demonstrated benefits to the business like substantial timesaving and error reduction. As part of Shire’s configured Documentum with FirstDoc system, a bulk import tool is provided. This tool has two options, to populate the metadata for each file to be imported, with limited capabilities of copying across documents, or to use an XML file to specify the metadata for a group of documents. Unfortunately, the system does not provide a mechanism for generating the XML file for bulk importing. Therefore, the use of Base SAS and the Macro language capabilities offered a quick, effective solution for creating the XML control files.

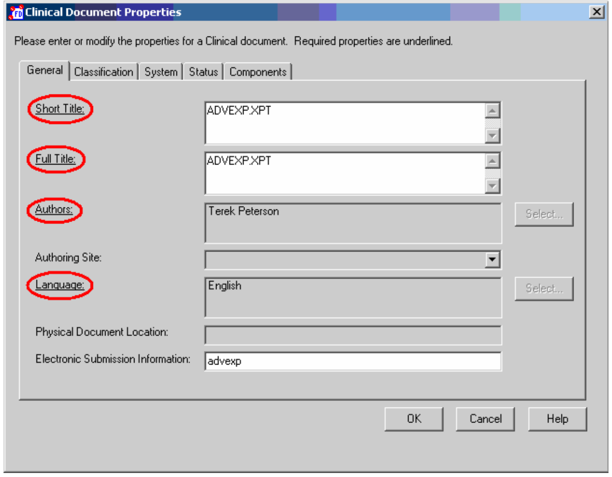
DOCUMENTUM REQUIREMENTS

Documentum is a pharmaceutical industry standard electronic document management system. Shire implemented this system in 2002 to facilitate global regulatory dossier document management and compilation. The system that was implemented included a Documentum with FirstDoc configuration of document types and subtypes, properties and controlled folder structure. Shire has adopted this model to minimize customizations to the system, but still allowing for some augmentation to meet the business needs. Part of Shire’s Documentum system configuration is a controlled folder hierarchy to ensure consistent document indexing across products. This folder hierarchy is determined by the properties assigned to the documents. For this reason, each document has several required fields to be populated on import.

The requirements for regulatory dossiers vary by region. One of the FDA requirements is SAS datasets from clinical trials; the SAS programs are also frequently requested. Shire’s standard is for all regulatory dossier components to be created in or imported to the repository prior to submission to the regulatory authority. This offers version control and tracking capabilities that are not available outside of a controlled system. Depending on the trial, the number of required files can quickly become a burden when manually importing into the Documentum repository.

When populating Documentum manually, as can be seen in Figures 1 and 2 below, the user enters document or file metadata including the title and type of document or file. Some fields are populated automatically by pulling network user information. Each field, like Short Title, has a related object name which will be the variable’s name used to populate the document’s metadata. Under each figure below, the name of each object in our Documentum system and the associated variable name are illustrated. Some objects will be carried over from the Bulk Import kick off screen, see Figure 5 on the next page. Your system may be different, so contact your system administrator for details.

Figure 1

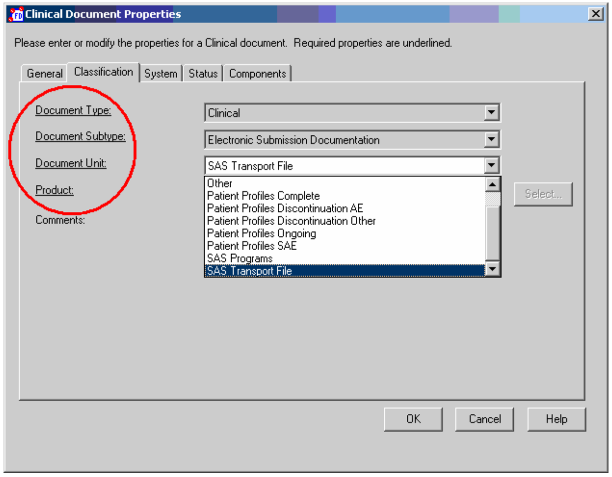


Please enter or modify the properties for a Clinical document. Required properties are underlined.

Short Title: ADVEXP.XPT
Full Title: ADVEXP.XPT
Authors: Terek Peterson
Authoring Site:
Language: English
Physical Document Location:
Electronic Submission Information: advexp

Displayed Text	= Variable Name
Short Title	= OBJECT_NAME
Full Title	= TITLE
Authors	= AUTHORS
Language	= LANGUAGE

Figure 2



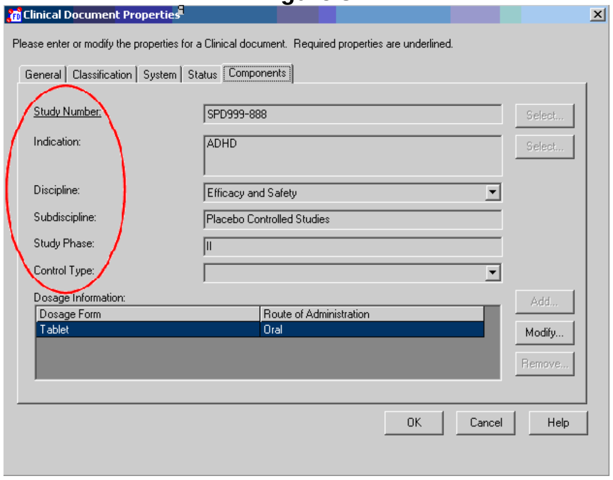
Please enter or modify the properties for a Clinical document. Required properties are underlined.

Document Type: Clinical
Document Subtype: Electronic Submission Documentation
Document Unit: SAS Transport File
Product:
Comments:

Other
Patient Profiles Complete
Patient Profiles Discontinuation AE
Patient Profiles Discontinuation Other
Patient Profiles Ongoing
Patient Profiles SAE
SAS Programs
SAS Transport File

Displayed Text	= Variable Name
Type	= DOCUMENT_TYPE
Subtype	= DOCUMENT_SUBTYPE
Document Unit	= DOCUMENT_UNIT

Figure 3



Clinical Document Properties

Please enter or modify the properties for a Clinical document. Required properties are underlined.

Study Number: SPD999-888

Indication: ADHD

Discipline: Efficacy and Safety

Subdiscipline: Placebo Controlled Studies

Study Phase: II

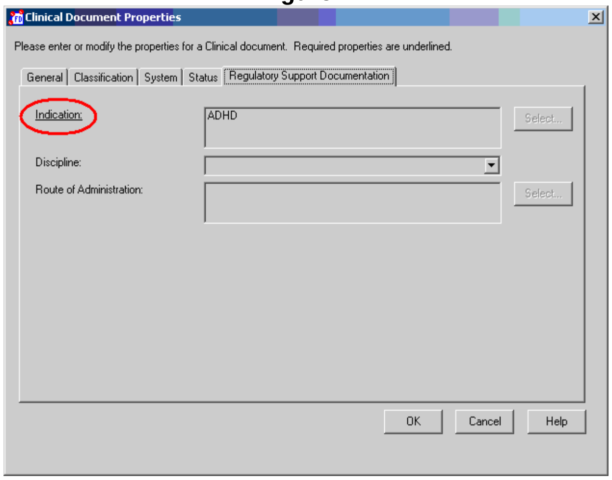
Control Type:

Dosage Information:

Dosage Form	Route of Administration
Tablet	Oral

For Clinical Trial documents, all of this information comes from the study number in the XML file.

Figure 4



Clinical Document Properties

Please enter or modify the properties for a Clinical document. Required properties are underlined.

Indication: ADHD

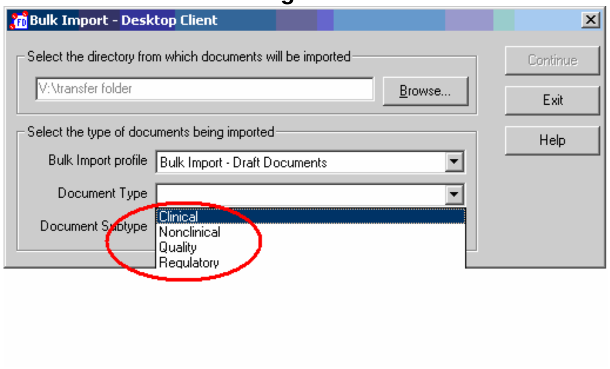
Discipline:

Route of Administration:

For Regulatory Support Documentation, no study report is required, but Indication is.

To kick off the bulk import, the following screen will launch, see Figure 5. The user needs to specify a folder, the profile (we always import to draft), and the document type and document unit. The XML file has to be in the folder specified.

Figure 5



Bulk Import - Desktop Client

Select the directory from which documents will be imported

V:\transfer folder

Select the type of documents being imported

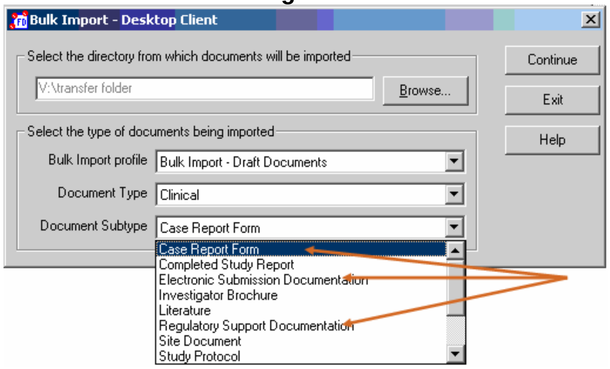
Bulk Import profile: Bulk Import - Draft Documents

Document Type:

Document Subtype

Bulk import screen – choose document type

Figure 6



Bulk Import - Desktop Client

Select the directory from which documents will be imported

V:\transfer folder

Select the type of documents being imported

Bulk Import profile: Bulk Import - Draft Documents

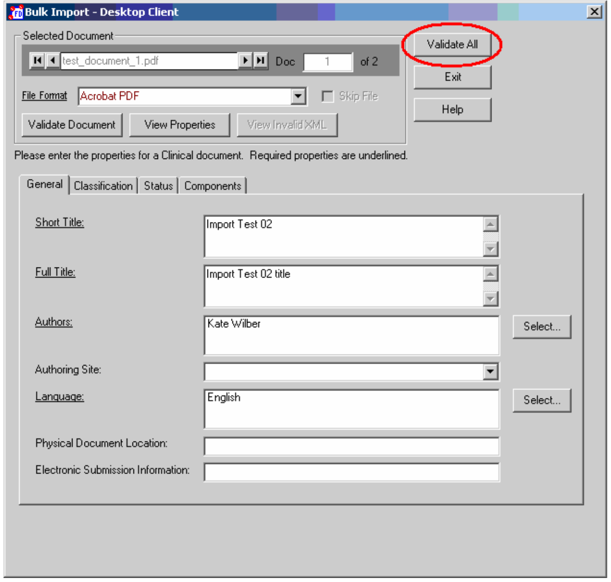
Document Type: Clinical

Document Subtype: Case Report Form

Then Document Subtype – subtype depends on type, Document unit depends on subtype, some documents can have a sub-unit which is dependent on the document unit.

As seen in Figure 6, the three most common documents that we bulk import are Case Report Forms (CRFs), Electronic Submission Documentation (SAS programs and SAS transport files for each trial), and Regulatory Support Documentation (SAS programs and SAS transport files for integrated analyses). Once the document subtype is specified the following screen launches, see Figure 7 on the next page.

Figure 7



Selected Document

test_document_1.pdf Doc 1 of 2

File Format: Acrobat PDF ☐ Skip File

Please enter the properties for a Clinical document. Required properties are underlined.

Short Title: Import Test 02

Full Title: Import Test 02 title

Authors: Kate Wilber

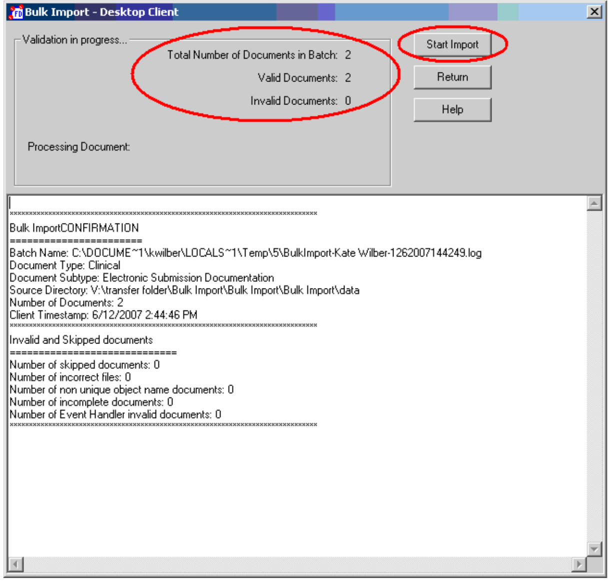
Authoring Site:

Language: English

Physical Document Location:

Electronic Submission Information:

Figure 8



Validation in progress...

Total Number of Documents in Batch: 2
Valid Documents: 2
Invalid Documents: 0

Processing Document:

```
********************************************************************************
Bulk ImportCONFIRMATION
========================
Batch Name: C:\DOCUME~1\kwilber\LOCALS~1\Temp\5\BulkImport-Kate Wilber-1262007144249.log
Document Type: Clinical
Document Subtype: Electronic Submission Documentation
Source Directory: V:\transfer folder\Bulk Import\Bulk Import\Bulk Import\data
Number of Documents: 2
Client Timestamp: 6/12/2007 2:44:46 PM
********************************************************************************
Invalid and Skipped documents
==============================
Number of skipped documents: 0
Number of incorrect files: 0
Number of non unique object name documents: 0
Number of incomplete documents: 0
Number of Event Handler invalid documents: 0
********************************************************************************
```

Typically all attributes are blank which the user needs to specify before documents can be imported. Manual bulk loading takes a considerable amount of time, especially when there are a large number of objects to load. Using the bulk import process, the user selects the bulk import XML control file, the bulk import begins immediately, and the DMS system validates the import process for each file in the XML control file. It is recommended to verify that the XML control file has populated the fields correctly before clicking on the "Validate All" button.

As can be seen in Figure 8, the validation confirmation for the bulk import is displayed. This document should be reviewed for messages indicating invalid document attributes. A decision must be made to fix invalid attributes manually or to recreate the XML control file and start the whole process over. However, if no problems exist then the import can begin by clicking on the “Start Import” button.

BULK IMPORT VIA XML CONTROL FILE

Bulk import enables users to import multiple documents of various types using a XML control file. Basically the XML control file will contain the name of a document to import and the document’s mandatory attributes required by the Documentum system.

XML INSTRUCTIONS

The Bulk Import Utility uses the attribute values specified in an XML control file. Collectively the XML tags and document attributes function to control and label imported documents for General and System Classifications with attributes identifying the type of file being imported, titles, authors, language, supportive type and subtype information (Clinical, Regulatory), compound/product information, status of the document (approved, draft), and regulatory support information such as indication, efficacy and/or safety, and route of drug administration. The processing of the XML file via the Bulk Import Utility configures and automates the import with these attributes.

RESEARCH / CODING REQUIREMENTS

The prerequisites necessary to create an XML Bulk Import file are outlined below:

1. Contact the Documentum Administrator to obtain the mandatory attributes for the specific file type(s) to import.

2. Outline how the required attribute names will be linked within the XML encapsulated tags. The required attributes will be identified as a “value” tag for each external and internal mandatory attribute to be processed from the XML code during the bulk import.

3. Create a flowchart showing how the XML file will be structured to incorporate the XML tags as required by the Documentum system such that all tag names are balanced with an identical ending tag name.

4. Structure the SAS code such that each file to be processed will contain the mandatory attributes encapsulated (or balanced) by the appropriate XML tags.

5. List the attributes required by your Documentum system to facilitate your programming plan. See Table 1 for an example of how attributes can be setup in a Documentum system.

Table 1

Mandatory Attribute Name	Function	Programming Considerations
R_OBJECT_ID	Object ID Internal to Documentum *(The value from the XML file is actually replaced by a Documentum assigned object ID, but is required for the import to run successfully)*	Use a counter, incremented for each file processed.
OBJECT_NAME	Document Name (*This is the name as displayed to users when the document goes into the system)*	Document Name as read from Processed Directory
DOCUMENT_UNIT	Indication of Document Type *(Document unit is level of classification. It is possible to import documents of more than one document unit at a single time)*	Example : SAS Programs
TITLE	Full Document Title (*This is the document's full title in the System, typically make the title equal the object name)*	SAS Program name, for example AE5.SAS
AUTHORS	Person responsible for Importing Documents	Parameter. Note: Expand code section for more than one author (if required).
DOCUMENT_LANGUAGE	English	Default
DOCUMENT_STATUS	Draft	Default
PRODUCT	Name of drug or compound as specified by the business unit. *(Can be internal development number, generic name or trade name)*	Brand, Chemical Name, etc.
STUDY_NUMBER	Actual Study Number	Per Organizational Requirements
INDICATION	Condition under investigation *(This is required for Regulatory Support Documentation, but not for trial related documents, as it is inherited from the study number for trial documents)*	Parameter control for Drug Indication

The user interface attributes shown in Table 2 must be populated through the user interface at the start of the bulk load. An example of the user interface was presented in Figures 5 and 6 earlier in the paper.

Table 2

Mandatory Attribute Name	Function	Programming Considerations
DOCUMENT_TYPE	Clinical (*Other types are Quality, Regulatory, Non-clincal)*	Specified by the Business Unit
DOCUMENT_SUBTYPE	Regulatory Support Documentation *(A number of subtypes are available depending on the document type – Datasets and Programs fall under Electronic Submission Documentation, unless they are for the ISS or ISE, in which case they go into Regulatory Support Documentation. This field is likely to evolve as we upgrade the system)*	Specified by the Business Unit

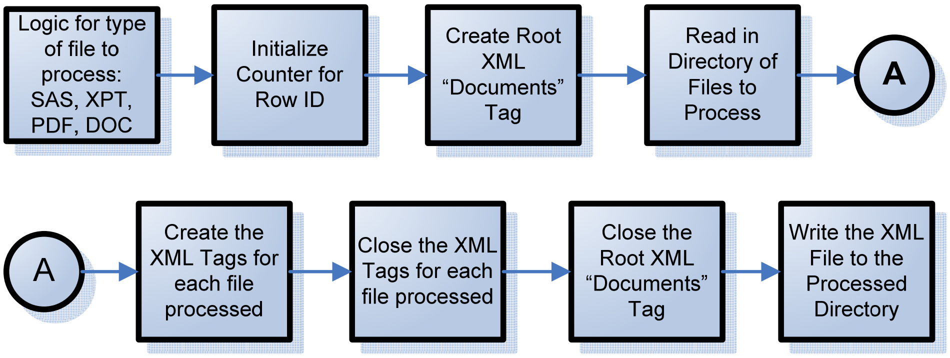
6. Draft a plan to read in the documents to be loaded into Documentum from a specific directory, for a specific file type to be processed.

7. Draft a plan to process all of the files in that directory and apply the mandatory attribute requirements for the Documentum System.

8. Process each file's information one at a time.

9. Once all of the files have been processed, complete the scheme to process the XML closing "root" tags and initiate the code (data _null_) necessary to write out the XML file in the prescribed directory.

10. Code Flow →


Logic for type of file to process: SAS, XPT, PDF, DOC
Initialize Counter for Row ID
Create Root XML “Documents” Tag
Read in Directory of Files to Process
A
A
Create the XML Tags for each file processed
Close the XML Tags for each file processed
Close the Root XML “Documents” Tag
Write the XML File to the Processed Directory

CODING OVERVIEW

The SAS code will be required to read in directory information containing all files to be processed. Loop through each file applying the mandatory attributes required by the Documentum database (a loop counter will be used to create a dummy Row ID used by the Documentum system). After processing each file, the SAS code will write out a BulkImport.xml file to the processed directory. Create a filename statement for the final xml file output.

```
%let driver= D:\Biostatistics\StudyFiles\P500_S101\code;

filename outfile "&driver\bulkimport.xml”;
```

CODING: HEADER (ROOT TAG SECTION)

Requirements of the XML Header Section:

The XML Header Section identifies the name of the final bulkimport XML file, and will create the “import” XML tag for the XML Version and Encoding (“windows-1252”), and the root XML tag “Documents”.

```
data root(label='ROOT Tags, Balanced later in end_root datastep');
  length tag $300;
  tag="<?xml version=""1.0"" encoding=""windows-1252"" ?>";              output;
  tag="<import>";                                                        output;
  tag="<documents>";                                                     output;
run;
```

CODING: XML BODY

How to Create the XML Body Section loops through and creates XML body for each file you process:

The body will create the Mandatory Document Attribute tags dataset for each file processed mandatory attribute tags as illustrated below with 2 attribute examples:

```
%let loop=%eval(&loop+1);

data body;
  length tag $300;
  tag="<document>";                                                      output;
    tag="<content>";                                                     output;
      tag="<primary format=""&doc_format"">"||"&driver\&document_name";  output;
      tag="</primary>";                                                  output;
    tag="</content>";                                                    output;

    tag="<attributes>";                                                  output;
      tag="<attribute usage=""internal"">";                              output;
        tag="<external_source_name>r_object_id</external_source_name>";  output;
        tag="<values>";                                                  output;
          tag="<value>&loop</value>";                                    output;
        tag="</values>";                                                 output;
      tag="</attribute>";                                                output;
      tag="<attribute>";                                                  output;
        tag="<external_source_name>object_name</external_source_name>";   output;
        tag="<internal_target_name>object_name</internal_target_name>";   output;
        tag="<values>";                                                   output;
          tag="<value>&document_name</value>";                            output;
        tag="</values>";                                                  output;
      tag="</attribute>";                                                 output;

       /* Add blocks of tags for each mandatory attribute*/

    tag="</attributes>";                                                  output;
  tag="</document>";                                                      output;
run;
```

CODING: CLOSING (ROOT TAG SECTION)
Requirements of the XML Closing Section:

The Closing section will apply the balanced closing tag, place the root Header Section, Body, and Closing Sections together, then write the XML file.

```
data end_root(label='End Root, Balances Beginning ROOT Tags');
  length tag $300;
  tag='</documents>';  output;
  tag='</import>';     output;
run;


data final_xml;
  set root
      body
      end_root;
run;


data _null_;
  set final_xml end=eof;
  file outfile recfm=V lrecl=300;
  put @1 tag;
run;
```

VIEW OF XML FILE CREATED BY BULKIMPORT PROGRAM
Below is an example of the Bulkimport.xml file created from Base SAS. Please note that some tags like the ‘*usage=””internal””*’, ‘*external_source_name*’, and ‘*internal_source_name*’ are necessary for the Documentum system to process the XML file correctly.

```
<?xml version="1.0" encoding="windows -1252" ?>
<import>
   <documents>
      <document>
         <content>
            <primary format="sas">C:\PROJECT\CODE_DIRECTORY\AE5.SAS</primary>
         </content>

         <attributes>
            <attribute usage="internal">
               <external_source_name>r_object_id</external_source_name>
               <values>
                  <value>1</value>
               </values>
            </attribute>

           <attribute>
              <external_source_name>object_name</external_source_name>
              <internal_target_name>object_name</internal_target_name>
               <values>
                  <value>AE5.SAS</value>
               </values>
            </attribute>
            <attribute>
               <external_source_name>title</external_source_name>
               <internal_target_name>title</internal_target_name>
               <values>
                  <value>SAS Program AE5.SAS</value>
               </values>
            </attribute>

            <attribute>
               <external_source_name>authors</external_source_name>
               <internal_target_name>authors</internal_target_name>
               <values>
                  <value>John Booterbaugh</value>
               </values>
            </attribute>

               Repeated for each document and attribute

         </attributes>
      </document>
   </documents>
</import>
```

CONCLUSION

By and large, the Bulk Import process has saved the company time and money, by off-loading the manual time required to individually load documents into the DMS to an automated process. The automated process retains the integrity of the import in at least two ways. Firstly, by loading all of the documents required without the need to cross-check that none were missed (as compared to a manual cross-check necessary with a manual load), and secondly, by correctly applying the required attributes.

One limitation noted with our system was that only about 100 objects could be loaded at a time. Large loads failed and this seems to be a limitation of the memory allocation in Documentum for processing imports. Therefore, the large loads were divided into smaller loads and multiple control files set into separate sub-folders for the import.

Future enhancements may include a report of the directories and files processed by the bulkimport program as supportive documentation; this could be used to document the import. However, since the Bulk Import facility of our DMS provides a validation and confirmation of the files imported, at this time the additional processing for such a report is not a requisite.

Essentially, the Bulk Import process has had a huge positive impact on the corporation by saving time and effort for personnel that formerly imported documents manually, and whose time is better spend on other critical tasks.

REFERENCES

[1] US Health and Human Services - Food and Drug Administration, “Guidance for Industry: Providing Regulatory Submissions in Electronic Format – Human Pharmaceutical Product Applications and Related Submissions Using the eCTD Specifications”, Revision 1 issued April 19, 2006. Document can be found at http://www.fda.gov/cder/guidance/7087rev.pdf.

Electronic Common Technical Document (eCTD) http://www.fda.gov/cder/regulatory/ersr/ectd.htm

Cisternas, Miriam. Cisternas, Ricardo. “Reading and Writing XML files from SAS®”. Paper 119 29.
SUGI 29 Proceedings.

Hoyle, Larry, The University of Kansas. “Reading Microsoft Word XML files with SAS®”. Paper 019 31.
SUGI 31 Proceedings.

Conway, Ted, Chicago, IL. “XSLT Friendly XML: Generating Structured XML from SAS® Data”.
Paper 061 31. SUGI 31 Proceedings.

XML.com. “What is XML?” http://www.xml.com/pub/a/98/10/guide0.html

ACKNOWLEDGMENTS

The authors would like to thank Sandra Harrison for her work story, Sanjay Patel for expertise around the Documentum metadata, and David Izard for his encouragement to try new ideas and review of this paper. Also acknowledged for review of this paper are: Mohamed Hamdani, Jeanette Ilett and Mike Swalina.

CONTACT INFORMATION

John Booterbaugh, RN, BSN
PharmaLogic, LLC
1000 Stuart Drive
Sanatoga, PA 19464 USA
(484) 595-8779
PharmaLogicLLC@comcast.net

Terek J. Peterson, MBA
Shire Pharmaceuticals
725 Chesterbrook Boulevard
Wayne, PA 19087 USA
(484) 595-8947
tpeterson@shire.com

Kate Wilber
Shire Pharmaceuticals
Hampshire International Business Park
Chineham Basingstoke
Hampshire RG24 8EP UK
+44 (0) 1256 894148
kwilber@shire.com

COPYRIGHT INFORMATION