PharmaSUG 2012 - Paper AD10

Experience of Generating an XML file and Uploading Serious and Frequent Adverse Events to ClinicalTrials.gov

Juan Wu, Medtronic Inc., Santa Rosa, CA
Min Lai, Medtronic Inc., Santa Rosa, CA

ABSTRACT

The Food and Drug Administration Amendments Act (FDAAA) requires that basic results for applicable clinical trials for drugs, devices, and biologics be posted to ClinicalTrials.gov after September 27, 2008, and serious adverse events (SAE) and frequent adverse events (FAE) be posted to ClinicalTrials.gov after Sep 27, 2009. To comply with the requirements, a cross functional team was established at Medtronic to implement the necessary infrastructure. While regulatory or clinical staff performs the data entry for basic results posting using a Web-based Results Registration System at Medtronic, our statistical programming team produced validated SAS® macros and generated an XML file for posting serious and frequent adverse events to ClinicalTrials.gov. The posting process also ensures the SAE/FAE results are published to ClinicalTrials.gov accurately and efficiently. This paper will discuss the experience learned in generating and uploading an XML file for SAE and FAE posting to ClinicalTrials.gov, will describe the flow chart to illustrate the process and will include explanation of the macros.

INTRODUCTION

U.S. Public Law 110-85[1](Food and Drug Administration Amendments Act of 2007 or FDAAA), Title VIII, Section 801 requires basic results be posted to ClinicalTrials.gov after September 27, 2008, serious adverse events (SAE) and frequent adverse events (FAE) be posted to ClinicalTrials.gov after September 27, 2009, for all "applicable" trials for Drugs, Biologics and Devices.

The law requires the following two types of events must be posted to ClinicalTrials.gov.

• Serious Adverse Events:

A table of all anticipated and unanticipated serious adverse events, grouped by system organ class, with number and frequency of such events, in each arm of the clinical trial.

• Other (Not Including Serious) Adverse Events:

A table of anticipated and unanticipated events (not included in the serious adverse event table) that exceed a frequency threshold of not more than 5% within any arm of the clinical trial, grouped by system organ class, with number and frequency of such events in each arm of the clinical trial.

In order to comply with the requirements, a cross functional team was established at Medtronic to implement the necessary infrastructure. While regulatory or clinical staff performs data entry into the Protocol Registration System (PRS) and basic results into the Results Registration System (RRS) at ClinicalTrials.gov using a Web-based Results Registration System at Medtronic., the SAS programming team develops SAS® macros based on Daniel B., Andy I., Troy R.,"ClinicalTrials.gov: Creating an XML upload from SAS®"[2] in NESUG 2010, uses existing AE analysis data set, and generates an XML file with results of serious and frequent adverse events for uploading purposes. This method avoids data entry error and ensures the SAE/FAE results loaded to ClinicalTrials.gov more efficiently.

MACRO PROCESS

The flow chart of the PRS SAE/FAE result uploading process is displayed in Figure 1.

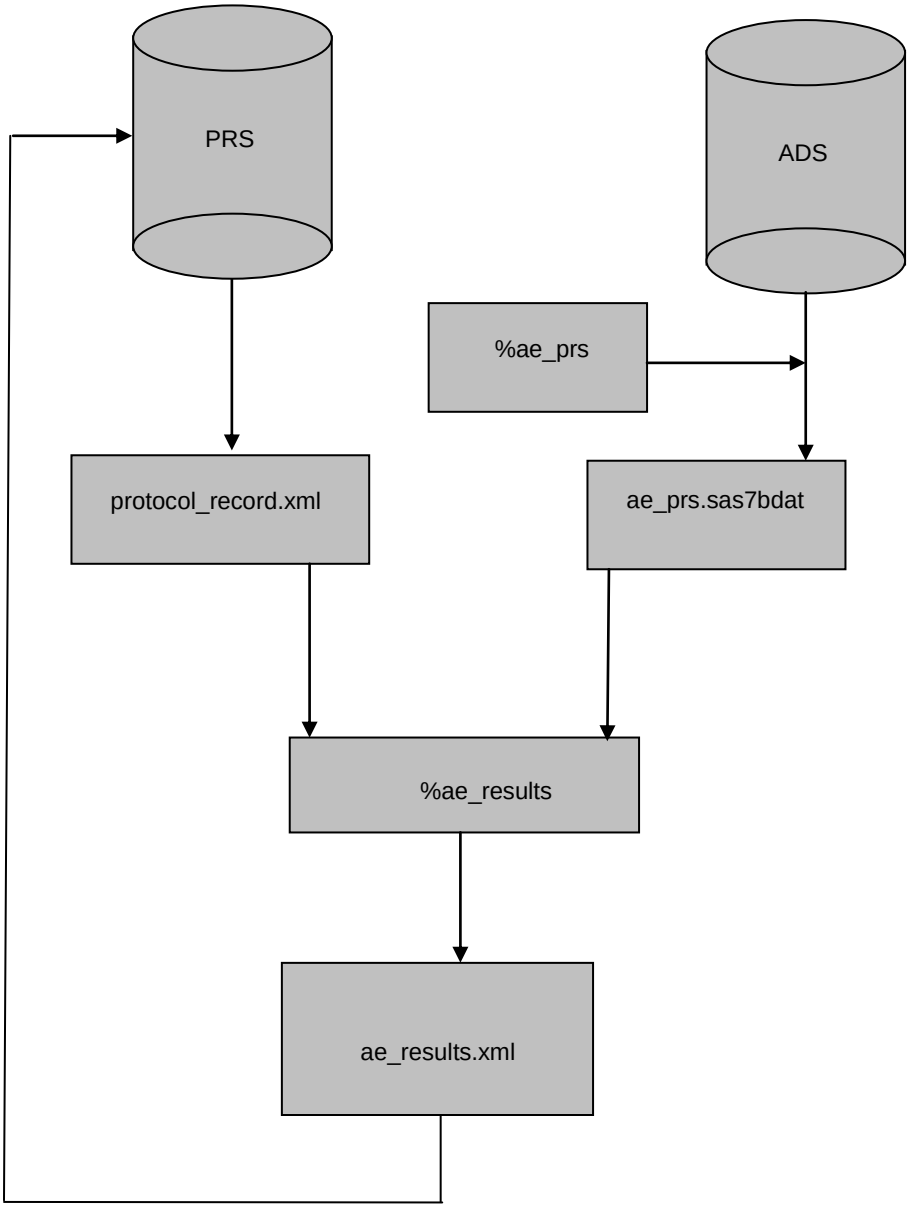

Figure 1: Flow Chart of PRS AE Result Uploading Process

UPLOADING INSTRUCTION

In order to upload the SAE/FAE results to PRS, the programmers at Medtronic took a two-step approach.

1) Use the first macro (%ae_prs) to generate an SAE/FAE result data set (ae_prs.sas7bdat). It matches the requirement of ClinicalTrials.gov by using two existing analysis data sets (ADS) that we already generated for regular reports such as annual progress report (APR), clinical study report (CSR).

2) Second macro (%ae_results) is used to convert the result data set to an XML file by using a pre-defined XML schema downloaded from ClinicalTrials.gov. We learned this method from the paper of "ClinicalTrials.gov: Creating an XML upload from SAS®"[2]. This XML file is ready for uploading by utilizing the PRS XML upload facility.

Since these macros were developed, study teams at Medtronic have successfully posted several study trial results to ClinicalTrial.gov. We have established following steps to ensure an accurate, successful uploading of SAE/FAE.

1) First, the regulatory or clinical staff enters basic trial information including basic results into PRS, leaving section of Adverse Events untouched.

2) Download Results XML at the bottom of the section "Results".

3) Send Results XML - "protocol_records.xml" to the study SAS programmer.

4) The study SAS programmer generates the ae_results.xml by using %ae_prs and %ae_results

5) After receiving the file "ae_results.xml" from the SAS programmer, the regulatory or clinical staff uploads the file into the PRS.

a) Back to Main Menu

b) Click Upload protocol records

c) File Name: Provide the full pathname of the XML file to be uploaded - Encoding: Unicode (UTF-8)

d) Upload

6) Check the results for the data integrity.

a) Click preview

b) Roll down till "Reported Adverse Events"

SAS® MACROS

MACRO %AE_PRS

The first macro (%ae_prs) is to generate an SAE/FAE result data set (ae_prs.sas7bdat) which matches the requirement of ClinicalTrials.gov prior to generating an XML file. The two existing analysis data sets (ADS) are needed.

1. ADS.AE, which includes the following six variables - subject ID (subjid), treatment group (treat), system organ class (soc), preferred term (pt), serious adverse events flag variable (SAEYN), and time frame variable (days_ae).

2. ADS.DM, which includes two required variables - subject ID (subjid) and treatment group (treat) for getting analysis population for each group.

The table content of the data set ae_prs is provided in Table 1. It contains one record per treatment group per system organ class per preferred term. The output of the data set ae_prs is partially displayed in the Table 2.

Variable	Label
XMLORDER	Order for AETYPE
AETYPE	Grouping variable including FAE, GROUPS and SAE
SOC	System Organ Class
PT	Preferred Term
GRPID	Treatment group values in a sequential order starting with 1 Used for intervention Group id or reporting Group id in XML
GRPDESC	Description of treatment group
GRPTITLE	Short title of treatment group
N	Count of unique subjects who experienced SAE/FAE
EVENTS	Count of events who experienced SAE/FAE
ANYFAE	Total number of all unique subjects who had an FAE per treatment group
ANYSAE	Total number of all unique subjects who had an SAE per treatment group
DENOM	Overall number of subjects included in the assessment of SAE/FAE

Table 1. The table content of the data set ae_prs

XMLORDER	AETYPE	SOC	PT	TRT	GRPID	GRPDESC	GRPTITLE	N	EVENTS	ANYFAE	ANYSAE	DENOM
1	FAE	General disorders and administration site conditions	Chest Pain	1	1	Treat 1	1. Group 1	0	0			100
1	FAE	General disorders and administration site conditions	Chest Pain	2	2	Treat 2	2. Group 2	6	7			100
2	GROUPS			1	1	Treat 1	1. Group 1			0	30	100
3	GROUPS			2	2	Treat 2	2. Group 2			6	25	100
9	SAE	Blood and lymphatic system disorders	Anaemia	1	1	Treat 1	1. Group 1	1	2			100
9	SAE	Blood and lymphatic system disorders	Anaemia	2	2	Treat 2	2. Group 2	0	1			100
9	SAE	Blood and lymphatic system disorders	Iron Deficiency Anaemia	1	1	Treat 1	1. Group 1	1	0			100
9	SAE	Blood and lymphatic system disorders	Iron Deficiency Anaemia	2	2	Treat 2	2. Group 2	0	0			100
9	SAE	Blood and lymphatic system disorders	Pancytopenia	1	1	Treat 1	1. Group 1	1	0			100
9	SAE	Blood and lymphatic system disorders	Pancytopenia	2	2	Treat 2	2. Group 2	0	0			100

Table 2. The output of the data set ae_prs

There are several steps in the %ae_prs.

Step 1: The overall number of subjects included in the assessment of SAE/FAE (i.e., the denominator for calculating frequency of SAE/FAE) per treatment group is obtained by using the following code.

```
proc sql noprint;
    create table _denom as select &trtVar, count(*) as denom from &dsDM
    where &trtvar in (&useValues) group by &trtVar;
quit;
```

Step 2: The number of events (EVENTS) and the number of unique subjects (N) who experienced FAE per treatment group per system organ class per preferred term are generated. Per the requirement of ClinicalTrials.gov, only frequent adverse events (FAE) – other AE (excluding SAE) with the frequency threshold above 5% are reported, (see Table 2 where AETYPE="FAE"). Similar codes are generated for serious adverse events (SAE). (see Table 2 where AETYPE="SAE").

```
proc sql noprint;
    create table _ae as select treat, soc, pt, subjid,
           count(distinct(subjid)) as n, count(*) as event from aes
    where saeyn = 0 group by treat, soc, pt;
quit;

data _ae2;
    merge _ae _denom;
    by treat;
run;

proc sort; by treat; run;

data _ae3(keep = treat soc pt n event denom  subjid);
    set _ae2;
    if divide(n,denom) > 0.05;
run;

proc sql;
    create table _ae4 as select treat, soc, pt, n, event, denom,
    count(distinct subjid) as anyfae from _ae3 group by treat;
quit;

proc sort data=_ae4 out=_ae5(keep=soc pt) nodupkey; by soc pt; run;

data _ae5;
    set _ae5;
    %do i = 0 %to &nwords - 1;
           treat = &&v&i;
           output;
    %end;
run;

data _ae5;
    merge _ae5 _denom;
    by treat;
run;

data _nonsae;
    length AETYPE $8;
    merge _ae5 _ae4;
    by treat soc pt;
    if event = . then event = 0;
    if n = . then n = 0;
    if anyfae = . then anyfae = 0;
    AETYPE = 'FAE';
    xmlorder = '1';
run;

proc sort data=_nonsae nodupkey; by treat soc pt; run;
```

Step 3: The total number of all unique subjects who had an SAE per treatment group (ANYSAE) is produced. Similar codes are used for the total number of all unique subjects who had an FAE per treatment group (ANYFAE). (see Table 2 where AETYPE=" GROUPS").

```
proc sql;
    create table _anysae as
    select treat, count(distinct(subjid)) as anySAE from aes
    where saeyn = 1 group by treat;
quit;

data _group;
    merge _anysae _nonsae(keep=treat anyfae);
    by treat;
    AETYPE="GROUPS";
    xmlorder= strip(treat+1);
run;
```

Step 4: Set the three parts together to get the final data set for XML generation.

```
data _prs(rename=(grpid_c=grpid));
    set _nonsae _group _sae;
    denom=strip(put(denom,8.));
    grpdesc = put(treat,grpdesc.);
    grptitle = put(treat,grptitle.);
    grpid=treat;
    grpid_c=strip(put(grpid,1.));
    drop grpid;
run;

proc sql noprint;
    create table out.ae_prs as select xmlorder, aetype, soc, pt, treat as trt,
       grpid, grpdesc, grptitle, N, events, anyfae, anysae, denom from _prs
    order by xmlorder, aetype, soc, pt, trt;
quit;
```

MACRO %AE_RESULTS

The macro %AE_RESULTS converts the result data set to an XML file by using a PRS XML schema based on Daniel B., Andy I., Troy R.,"ClinicalTrials.gov: Creating an XML upload from SAS®"[2] in NESUG 2010.

The macro needs two input files: 1) An XML file (protocol_record.xml), which is downloaded from the results section from PRS; 2) The SAS® data set, which is generated from the first macro introduced in the paper per the requirement of PRS XML uploading and the XML schema[3, 4].

The output generated by this macro is a valid AE_RESULTS.xml file. It is ready for uploading by using PRS facility.

The SAS® code of %AE_RESULTS includes: 1) Read XML file (protocol_recode.xml) without SAE and FAE part; 2) Print SAE and FAE counts; 3) Print intervention groups information; and 4) Close Tags. (For detailed macro program, please see Daniel B., Andy I., Troy R., "ClinicalTrials.gov: Creating an XML upload from SAS®"[2] in NESUG 2010. We did some edits per our own experience.)

Table 3 describes the detailed mapping between variables in SAS data set generated by %ae_prs and XML tag.

Variable	Tag in XML	Description
SOC	<organSystemName>	System Organ Class Term
PT	<term>	Preferred Term
GRPID	<reportingGroupId> for AETYPE ne "GROUPS" or <interventionGroup id> for AETYPE = "GROUPS"	Sequential numbering of treatment groups starting with 1
GRPDESC	<description>	Description of the treatment variable
GRPTITLE	<title>	Long Description of the treatment variable
N	<numSubjectsAffected> for AETYPE ne "GROUPS"	Count of unique subjects per preferred term AETYPE="FAE" refers to FAE AETYPE="SAE" refers to SAE
EVENTS	<numEvents> for AETYPE ne "GROUPS"	Count of events per preferred term AETYPE="FAE" refers to FAE AETYPE="SAE" refers to SAE
ANYFAE	<numSubjectsFrequentEvents>	Total number of all unique subjects who had an FAE per treatment group
ANYSAE	<numSubjectsSeriousEvents>	Total number of all unique subjects who had an SAE per treatment group
DENOM	<partAtRiskFrequentEvents> for AETYPE = "FAE" or <partAtRiskSeriousEvents> for AETYPE = "SAE"	Total number of subjects at risk

Table 3. XML tag, description and the corresponding variables in the SAS data set ae_prs

The following code for printing SAE and FAE counts is based on Daniel B., Andy I., Troy R., "ClinicalTrials.gov: Creating an XML upload from SAS®"[2] in NESUG 2010 with our edits.

```
%*Print SAE and FAE counts;
IF left(aetype) ne "GROUPS" THEN DO;
    IF FIRST.aetype THEN DO;
           IF LEFT(aetype)='FAE' THEN PUT " <frequentAdverseEvents>";
           IF LEFT(aetype)="SAE"    THEN PUT " <seriousAdverseEvents>";
    END;

    IF FIRST.pt THEN DO;
           IF LEFT(aetype)="FAE" THEN PUT " <frequentEvent>";
           IF LEFT(aetype)="SAE"    THEN PUT " <seriousEvent>";
           PUT " <adverseEventStats>";
    END;

    PUT " <eventStats>";
    PUT " <reportingGroupId>ReportedEvents-InterventionGroup." grpid +(-
1)"</reportingGroupId>";
    PUT " <numEvents>" events +(-1)"</numEvents>";
    PUT " <numSubjectsAffected>" n +(-1)"</numSubjectsAffected>";
    PUT " <numSubjects>" denom +(-1)"</numSubjects>";
    PUT " </eventStats>";

    IF LAST.pt THEN DO;
           PUT " </adverseEventStats>";
           PUT " <assessmentType>Systematic Assessment</assessmentType>";
           PUT " <organSystemName>" soc +(-1)"</organSystemName>";
           PUT " <sourceVocabulary>MedDRA 10.0</sourceVocabulary>";
           PUT " <term>" pt +(-1)"</term>";
           IF left(aetype)="FAE" THEN PUT " </frequentEvent>";
           IF left(aetype)="SAE"    THEN PUT " </seriousEvent>";
    END;
```

```
    IF LAST.aetype THEN DO;
           IF left(aetype)="FAE" THEN PUT " </frequentAdverseEvents>";
           IF left(aetype)="SAE" THEN PUT " </seriousAdverseEvents>";
    END;
END;
```

Output 1 shows the corresponding XML output (partially) for a better understanding.

```
<frequentAdverseEvents>
<frequentEvent>
    <adverseEventStats>
        <eventStats>
            <reportingGroupId>ReportedEvents-InterventionGroup.1</reportingGroupId>
            <numEvents>0</numEvents>
            <numSubjectsAffected>0</numSubjectsAffected>
            <numSubjects>100</numSubjects>
        </eventStats>
        <eventStats>
            <reportingGroupId>ReportedEvents-InterventionGroup.2</reportingGroupId>
            <numEvents>7</numEvents>
            <numSubjectsAffected>6</numSubjectsAffected>
            <numSubjects>100</numSubjects>
        </eventStats>
    </adverseEventStats>
    <assessmentType>Systematic Assessment</assessmentType>
    <organSystemName>General disorders and administration site conditions</organSystemName>
    <sourceVocabulary>MedDRA 10.0</sourceVocabulary>
    <term>Chest Pain</term>
</frequentEvent>
</frequentAdverseEvents>


<interventionGroups>
    <interventionGroup id="ReportedEvents-InterventionGroup.1">
        <description>Treat 1</description>
        <numSubjectsFrequentEvents>0</numSubjectsFrequentEvents>
        <partAtRiskFrequentEvents>100</partAtRiskFrequentEvents>
        <numSubjectsSeriousEvents>30</numSubjectsSeriousEvents>
        <partAtRiskSeriousEvents>100</partAtRiskSeriousEvents>
        <title>1. Group 1</title>
    </interventionGroup>
        <interventionGroup id="ReportedEvents-InterventionGroup.2">
        <description>Treat 2</description>
        <numSubjectsFrequentEvents>6</numSubjectsFrequentEvents>
        <partAtRiskFrequentEvents>100</partAtRiskFrequentEvents>
        <numSubjectsSeriousEvents>25</numSubjectsSeriousEvents>
        <partAtRiskSeriousEvents>100</partAtRiskSeriousEvents>
        <title>2. Group 2</title>
    </interventionGroup>
</interventionGroups>


<seriousAdverseEvents>
<seriousEvent>
    <adverseEventStats>
        <eventStats>
            <reportingGroupId>ReportedEvents-InterventionGroup.1</reportingGroupId>
            <numEvents>2</numEvents>
            <numSubjectsAffected>1</numSubjectsAffected>
            <numSubjects>100</numSubjects>
        </eventStats>
        <eventStats>
            <reportingGroupId>ReportedEvents-InterventionGroup.2</reportingGroupId>
            <numEvents>1</numEvents>
            <numSubjectsAffected>0</numSubjectsAffected>
            <numSubjects>100</numSubjects>
        </eventStats>
```

```
        </adverseEventStats>
        <assessmentType>Systematic Assessment</assessmentType>
        <organSystemName>Blood and lymphatic system disorders</organSystemName>
        <sourceVocabulary>MedDRA 10.0</sourceVocabulary>
        <term>Anaemia</term>
    </seriousEvent>
    </seriousAdverseEvents>
```

Output 1. The corresponding XML output (partially)

CHALLENGES

We have encountered some extra challenges in addition to what has been presented in Boisvert, Daniel and Illidge, Andy and Ruth, Andy's paper[2].

Most clinical trials have APR or CSR submitted before loading SAE/FAE into ClinicalTrials.gov. When comparing SAE/FAE results between APR or CSR and PRS, please note:

1) Count of events per Preferred Term ("numEvents") usually is not required by CSR or APR.

2) Manually entering and editing in the Adverse Events section on Clinicaltrial.gov might leave traces in protocol_records.xml. It will cause a failure when uploading AE_RESULTS.xml. For example, during the testing phase, some SAEs were entered for one study with multiple arms and then were removed from the system, but PRS remembered the old intervention Group ID, which caused sequence numbering errors.

CONCLUSIONS

We have successfully generated xml files and posted serious adverse events and frequent adverse events to ClinicalTrials.gov for several clinical trials. While more sophisticated approaches are developed across the companies, this straightforward approach can be easily adapted from study to study and used to complete the tasks in a short time frame.

REFERENCES

[1] U.S. Public Law 110-85 (also known as Food and Drug Administration Amendments Act of 2007), Title VIII, Section 801. Available at https://register.clinicaltrials.gov/prs/html/fdaaa-info.html

[2] Boisvert, Daniel and Illidge, Andy and Ruth, Andy, 2010. "ClinicalTrials.gov: Creating an XML upload from SAS®" NESUG Annual Conference Proceedings. Available at http://www.nesug.org/Proceedings/nesug10/ph/ph01.pdf

[3] XML Schema for the results portion of clinical trial data to be uploaded into the Protocol Registration System (PRS). Available at http://wiki.hl7.org/images/1/19/Results_xml_schema.pdf

[4] ClinicalTrials.gov Protocol Data Element Definitions (DRAFT). Available at http://prsinfo.clinicaltrials.gov/results_definitions.html

ACKNOWLEDGEMENTS

Many thanks to Boisvert, Daniel and Illidge, Andy and Ruth, Andy to present and publish their paper in North East SAS User Group so that we are inspired to learn and implement xml generating and uploading in our own practice.

CONTACT INFORMATION

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

Name: Juan Wu
Enterprise: Medtronic Inc
E-mail: juan.wu@medtronic.com

Name: Min Lai
Enterprise: Medtronic Inc
E-mail: min.lai@medtronic.com