

Using SAS® to Audit Programming Code Versioning

Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

ABSTRACT

The solution presented will walk through the problem encountered with programming code versioning and how it was solved with an automated Base SAS® programming solution. Some featured techniques include SYSTASK Commands, Macro Processing, and Text Mining using SAS Functions.

INTRODUCTION

This is a technical paper about how to perform a “line-by-line” code comparison between the programming code in a “Production” library and the corresponding code in a “Versioning” library. The problem is that production environments may not be able to accommodate the ability to run program code from a “Version” library. This creates a situation where 2 copies of the programming code must be maintained. One copy of the code is run in production and the other is a “back-up version” of the production code in a versioning library. If these copies get out of sync, it poses a risk to the organization, whereby, the most recent changes and/or the entire program may be lost due to a system failure, accidental user event, or malicious user activity.

METHODOLOGY & SPECIFICATIONS

At PREMIER Bankcard TortoiseSVN (SVN), which is a Subversion Library used in an OpenSSL Toolkit, is a method to protect against loss of intellectual property in the form of SQL Script and/or SAS® Programming Code. This tool maintains historical copies of programming code/script that has to be checked out in order to be revised. There are several nice tools for editing and revising code that make it easy to identify changes by comparing multiple versions of a module using SVN highlighting features.

Execution is invoked directly against the most recent version in the library provided the “SVN Update” process is executed consistently. This is due to the fact that the SVN Library image the user interacts with is maintained on the resident “Hard Drive” of their computer. Thus, regular updating of the image is a significantly important part of the process.

Here is an example of DOS Command Script that will execute the SVN Update command for the most recent version of SAS Code and then execute the program module in batch:

```
TortoiseProc.exe /command:update /path:"C:\Subversion Portfolio
Review\SAS\Production Code\SAS_FS_FRM_Build.sas" /closeonend:2

"C:\Program Files\SAS\SASFoundation\9.2\sas.exe" -SYSIN "C:\Subversion
Portfolio Review\SAS\Production Code\SAS_FS_FRM_Build.sas"
```

This works great for SAS Code since it is stored and executed in test file scripts. Unfortunately, there are other script languages that are not stored as text files and subsequently parsed and/or compiled during execution.

PROBLEM

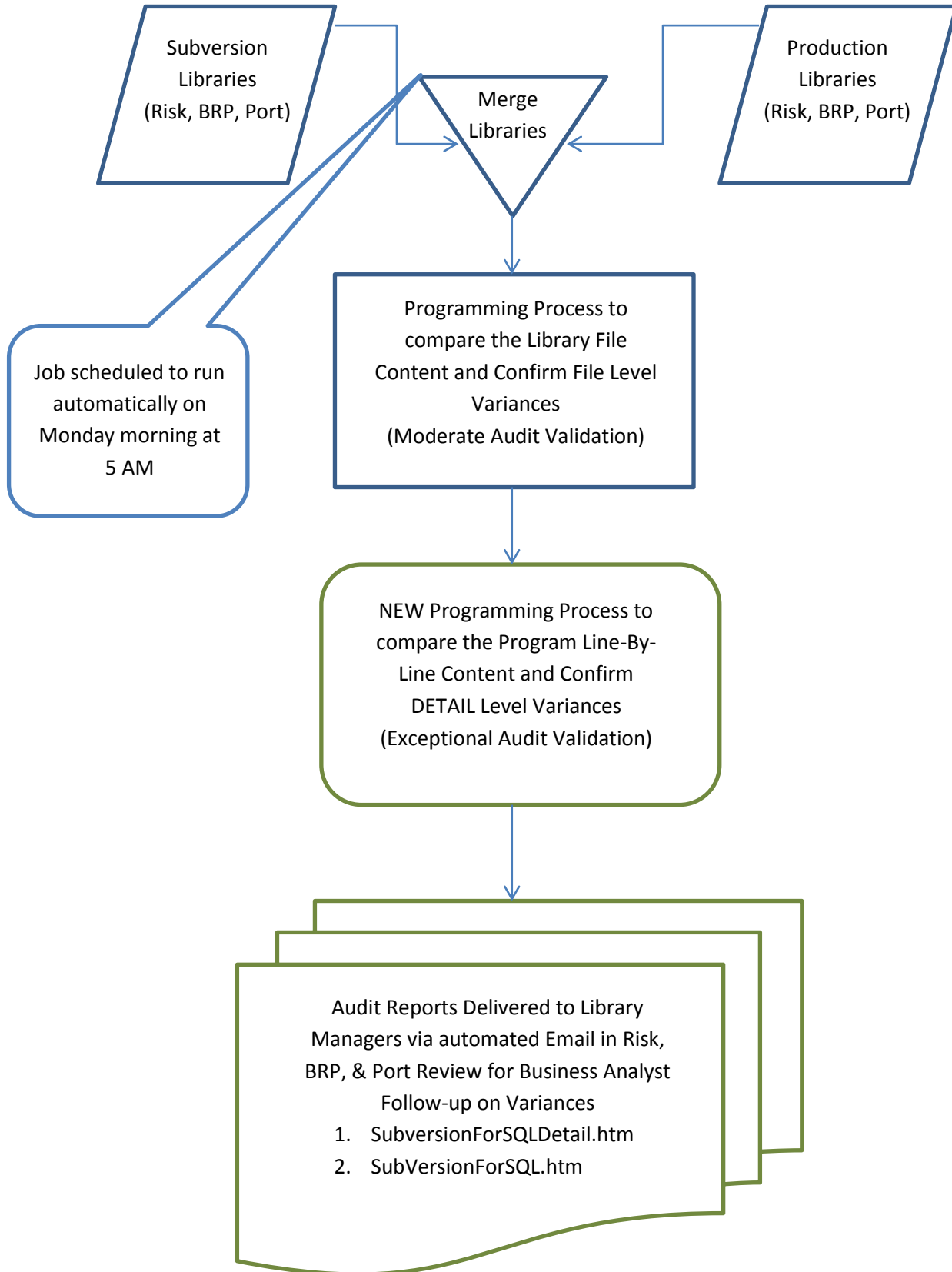
In PREMIER’s environment users create thousands of SQL executable stored procedure programs that are executed directly in the Microsoft® SQL Server® 2008 environment. These stored procedures are embedded in actual SQL tables instead of text files. Therefore, SVN cannot capture the versioning history for change management. This creates a risk of intellectual property loss to PREMIER through a system failure, accidental user event, or malicious user activity.

This became the subject of an audit finding during a third party audit of our technology environment. There were no known remedies to this issue and it continued to be reported accordingly for almost 3 years. In the spring of 2011, a SAS solution was recommended to the VP of MIS so this could be removed from the audit findings list. The solution is depicted in the following section using a process flow diagram.

Using SAS® to Audit Programming Code Versioning

Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

SOLUTION



Using SAS® to Audit Programming Code Versioning

Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

AUDIT PROGRAM – READ IN THE SCRIPT FILES

```

/*****
/*BEGIN - Audit the code line-by-line.*/
/*****
Options Center;
Options symbolgen mprint mlogic mrecall;
%let n=&MatchCount; /*set to number of files being processes*/
%macro readfiles;
/*****
/*BEGIN - Process all files in the Subversion Library (Repeated for Production Library)*/
/*****
%do i=1 %to &n;
Data _Null_;
    Set Validate
        (Where= (
            MatchCode='Matched Record'
            and
            MatchCount=&i
            ))
        ;
        Call Symput("FileNM&i",FileNm_Subver);
run;
filename in&i "&SubversionLibrary\&&FileNM&i";
data a&i;
    length
        CodeLine $1000.
        ;
    infile in&i Trunccover lrecl=1000 recfm=v;
    input
        @1 CodeLine $ &
        ;
    LineNo+1;
    FileNm="&&FileNM&i";
    CodeLine=Trim(Left(CodeLine));
    FileNm=Trim(Left(Uppcase(FileNm)));
    FileNm_Subver=Trim(Left(Uppcase(FileNm)));
    StartPos=Index(FileNm, '.STOREDPROCEDURE');
    If StartPos>0
        Then FileNmTrim=Substr(FileNm,1,StartPos-1)||'.SQL';
        Else FileNmTrim=FileNm;
    RUN;
%end;
Data a;
    set
        %do x=1 %to &n;
            a&x
        %end;
        ;
run;
proc datasets library=work;
    Delete
        %do y=1 %to &n;
            a&y
        %end;
        ;
run;
/*****
/*END - Process all files in the Subversion Library (Repeated for Production Library)*/
/*****

```

Paper BI-17
Using SAS® to Audit Programming Code Versioning
Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

AUDIT PROGRAM – LINE BY LINE COMPARE

```
/******  
/*BEGIN - Compare line-by-line the Subversion Library and Production Library)      */  
/******  
Proc Sort  
    Data=a  
    NoDupKey  
    ;  
    By FileNmTrim CodeLine;  
    run;  
  
Proc Sort  
    Data=b  
    NoDupKey  
    ;  
    By FileNmTrim CodeLine;  
    run;  
  
Data ValidateCodeLine;  
    Length  
        MatchCode  
            $20.  
        ;  
  
    Merge  
        a (in=ina)  
        b (in=inb)  
        ;  
        By FileNmTrim CodeLine;  
        If Ina & Inb Then MatchCode='Matched Record';  
        If Ina=1 & Inb=0 Then MatchCode='SubVersion Only';  
        If Ina=0 & Inb=1 Then MatchCode='Production Only';  
  
        If Index(Uppcase(CodeLine),'CREATE')^=0 Then Delete;  
        If Index(Uppcase(CodeLine),'OBJECT:')^=0 Then Delete;  
        If Index(Uppcase(CodeLine),'ALTER')^=0 Then Delete;  
        If Index(Uppcase(CodeLine),'USE ')^=0 Then Delete;  
        If Index(Uppcase(CodeLine),'GRANT')^=0 Then Delete;  
        If Ina & Inb Then Delete;  
        If Findc(Codeline,"a","i")=0 Then Delete;  
        run;  
/******  
/*END - Compare line-by-line the Subversion Library and Production Library)      */  
/******
```

Paper BI-17
Using SAS® to Audit Programming Code Versioning
Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

AUDIT PROGRAM – CREATE THE AUDIT REPORT

```
/******  
/*BEGIN - Create the Audit Report */  
/******  
ods listing close;  
ods html  
    path="&OutputLibrary"  
    file="SubversionForSQLDetail.htm"  
    style=PremierLogoRex  
    ;  
Title1 'Subversion SQL Code Line-By-Line Audit';  
Title2 "As of &TitleDt";  
Title4 'NOTE: These programs matched at the name level, but changes have been made ONLY in  
Subversion OR Production';  
Options Nocenter;  
  
Proc Print data=ValidateCodeLine  
    (Where=(  
        MatchCode^='Matched Record'  
    ))  
    noobs Label u n  
    ;  
By FileNmTrim;  
var  
    MatchCode  
    LineNo  
    CodeLine  
    FileNm_Subver  
    FileNm_Prod  
    ;  
Label  
    FileNmTrim='Matched Program File Name'  
    LineNo='Code Line #'  
    CodeLine='Line of SQL Code'  
    FileNm_Subver='Subversion Program File Name'  
    FileNm_Prod='Production Program File Name'  
    ;  
  
run;  
  
ods html close;  
ods listing;  
/******  
/*END - Create the Audit Report */  
/******
```

Paper BI-17
Using SAS® to Audit Programming Code Versioning
 Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

AUDIT REPORT OUTPUT

Auditors love evidential material. Therefore, two reports were developed to capture any variances that are evident during the systemic audit process. The first identifies any file that exists in the Subversion vs. the Production SQL libraries. The second identifies any lines of code that do not match between any file that is in both libraries.

The audit reports are delivered to library managers of each operational analytics support area (Risk, BRP, & Port Review) via automated email in for Business Analyst follow-up on variances.

1. SubversionForSQL.htm Report Sample

SAS Output

Page 1 of 1

*HTML Listing of Subversion SQL Procedures
 As of August 27, 2012 - 7:21:36
 Detail File Matching Information of Programs in PRODUCTION
 Matched on Write Date and Byte Count*

MatchCode=Production Only

| File Name | Write Date | SV Write Date | Write Date Match? | Byte Count | SV Byte Count | Byte Count Match? |
|---|------------|---------------|-------------------|---------------|---------------|-------------------|
| DBO.SPPR_RAGHU_SASTEST.SQL | 2012-08-26 | . | No | 2,400 | . | No |
| DBO.SPPR_RFA11808_PCPCANCELEDIN30DAYS.SQL | 2012-08-26 | . | No | 8,522 | . | No |
| DBO.SPSCHEDULE6AM.SQL | 2012-08-26 | . | No | 920 | . | No |
| MatchCode | | | | 11,842 | 0 | |
| N = 3 | | | | | | |

MatchCode=SubVersion Only

| File Name | Write Date | SV Write Date | Write Date Match? | Byte Count | SV Byte Count | Byte Count Match? |
|---|------------|---------------|-------------------|---------------|---------------|-------------------|
| DBO.SPPR_RFA11808_CANCELEDPCPIN30DAYS.SQL | . | 2012-08-21 | No | . | 4,286 | No |
| | | | | 11,842 | 4,286 | |
| N = 1 | | | | | | |
| Total N = 4 | | | | | | |

Paper BI-17
Using SAS® to Audit Programming Code Versioning
 Rex Pruitt, PREMIER Bankcard, LLC, Sioux Falls, SD

2. SubVersionForSQLDetail.htm Report Sample

SAS Output

Page 1 of 2

*Subversion SQL Code Line-By-Line Audit
 As of August 27, 2012*

NOTE: These programs matched at the name level, but changes have been made ONLY in Subversion OR Production

Matched Program File Name=DBO.HA_LPLOSSAVOIDANCEREP.SQL

| MatchCode | Code Line # | Line of SQL Code | Subversion Program File Name | Production Program File Name |
|-----------------|-------------|---|-------------------------------|------------------------------|
| SubVersion Only | 2639 | @MaxRptDt SMALLDATETIME, | DBO.HA_LPLOSSAVOIDANCEREP.SQL | |
| SubVersion Only | 2645 | @YrOldDay1 SMALLDATETIME -- 1st day of 12 Months ago | DBO.HA_LPLOSSAVOIDANCEREP.SQL | |
| SubVersion Only | 2647 | SET @MaxRptDt = (SELECT MAX(RptDate) FROM dbo.HA_LossAvoid_ExcelRollUp) | DBO.HA_LPLOSSAVOIDANCEREP.SQL | |
| SubVersion Only | 2648 | SET @Month = DATEPART(Month,@MaxRptDt) | DBO.HA_LPLOSSAVOIDANCEREP.SQL | |
| SubVersion Only | 2649 | SET @Year = DATEPART(YEAR,@MaxRptDt) | DBO.HA_LPLOSSAVOIDANCEREP.SQL | |
| SubVersion Only | 2650 | SET @Day = 01 | DBO.HA_LPLOSSAVOIDANCEREP.SQL | |
| N = 6 | | | | |

Matched Program File Name=DBO.KO_SCRA_NEW.SQL

| MatchCode | Code Line # | Line of SQL Code | Subversion Program File Name | Production Program File Name |
|-----------------|-------------|---|------------------------------|-------------------------------------|
| Production Only | 37 | if object_id ('dbo.Temp_SCRACurrentAccounts01') is not null drop table dbo.Temp_SCRACurrentAccounts01 | | DBO.KO_SCRA_NEW.STOREDPROCEDURE.SQL |
| N = 1 | | | | |

CONCLUSION

Implementation of this SVN Audit of SQL Stored Procedure programming code occurred during the summer of 2011 and was presented to the executive management team for official adoption into our best practice procedures. As a result of the third party audit in spring of 2012, the audit finding issue was determined to be resolved and subsequently removed from our exit report.

Having a method of protecting programming code is critical. Automating the audit of our SQL Stored Procedure programming code is protecting our company from a risk exposure to the potential loss of important corporate intellectual property.

Contact Information

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

Rex Pruitt
 PREMIER Bankcard
 3820 N. Louise Ave.
 Sioux Falls, SD 57107
 (605) 575-9810 - Office
rpruitt@premierbankcard.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.