Paper PP11

A simple tool to catalogue statistical outputs developed for submission by linking two in-house systems – experience from a submission project

Katja Diezel, Novartis Pharma AG, Basel, Switzerland

ABSTRACT

During the submission to health authorities an important quantity of tables, listings, figures and reports were produced. As the project grew the team felt the need for a user-friendly repository for such outputs. Joining information from two in-house systems using SAS, a simple tool was created to allow for the referencing of these outputs and any relevant information, thus increasing the speed, productivity and quality of outputs produced by the programming group.

INTRODUCTION

Background – Project

Novartis licensed a compound from a company for development and potential commercialization. During the submission process a large quantity of tables, listings, figures and reports were produced. Final tables, listings and figures were transferred to a document repository. These single file outputs were further processed by the publishing group and compiled into submission documents. Final reports are then made available in the document repository.

Background – Systems

SAS programming environment - Programmers and Statisticians work in the standard in-house programming environment GPSII.

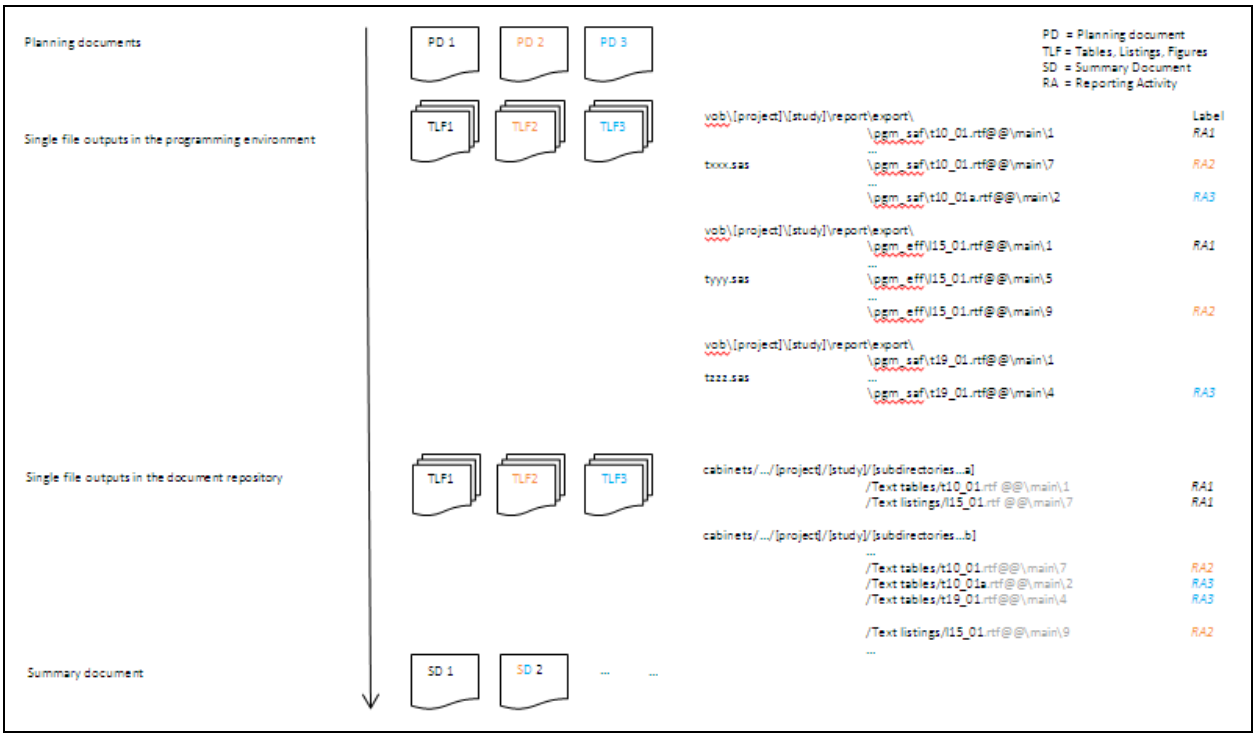
GPS II (Global Programming and Statistics) is a standard programming environment which used as a repository for data imported from in-house or external sources and for generating outputs (e.g. tables, listings) which form part of a regulatory submission report. The GPS II environment incorporates Version Control using Rational Clearcase software which records file history and provides traceability to guarantee 21CRF Part 11 compliance.

Document repository - Non-programming team members use the in-house document management system CREDI.

CREDI (Clinical Research Documentation and Information System) is a global document management system. It is a Documentum 6.5 Docbase with a web based client interface for managing documents. Users can access basic services, such as creating documents and participating in lifecycles and workflows using EMC Documentum Webtop.

WHAT WE NEEDED

Depending on the line function, different documents are referenced in order to gain an overview of the project. Statisticians lean towards the Statistical Analysis Plan and associated shells. Programmers tend to use tracking documents of SAS programs. Clinicians prefer to refer to final reports.



Planning documents (e.g. Statistical Analysis Plan and associated output shells, tracking documents of SAS programs) cover planned outputs per reporting activity. Those planning documents list planned outputs, but outputs might not have been programmed yet at a given point in time when the reporting is still ongoing. Information is spread out over multiple documents: one set of planning documents is available per reporting activity (CSR, SCS, SCE, 120 Day update, etc.).

Single file outputs in the programming environment are available once programmed and validated. The same program might be used for several reporting activities. In this situation the program would run on different versions of the datasets and for each set of data an output file is created. These output files might have identical file names and will differ in file version number, label and content.

For each reporting activity, once an output has been approved by the statistician the output is transferred from the programming environment to the document repository.

Single file outputs in the document repository are stored in different directories depending on the reporting activity and publishing needs. The document repository has a company wide pre-defined structure.

CREDI directory study as pre-defined:

Reporting Environment	**Number of standard directories**	**Structure of standard directories**	**Example**
Study-level, Production	25 standard CREDI directories defined for single file outputs (Tables, Listings and Figures)	6 for study + 2,3 or 4 levels down -> Total: 7-10 levels in directory structure	Cabinets/CREDI Projects/C/COM123A/CREDI Studies/COM123A1234/CSR (Clinical Study Report)/16 Appendices/16.2 Patient data listings/16.2.01 Discontinued patients
Project level, Production	132 standard CREDI directories defined for single file outputs (Tables, Listings and Figures)	4 for project + 4 or 5 levels down -> Total: 8-9 levels in directory structure	Cabinets/CREDI Projects/C/COM123A/Registration Documents/CTD 5.3.5.3 Analyses from more than one study/5.3.5.3 Efficacy/5.3.5.3 Efficacy Appendix 1/Tables

In case of additional reports and ad-hoc requests special purpose directories are created, hence the number of directories and levels for directories may increase.

The document repository is accessed via a web top. EMC Documentum Webtop is a browser-based interface that provides access to the EMC Documentum repository and content management services. This Documentum client is ideal for users who prefer the point-and-click navigational ease of a hierarchal, folder-based user interface, particularly when content is accessed from remote locations across the Web. [1]

Given the amount of sub-directories and number of levels of sub-directories, the point-and click interface of Documentum Webtop proves very time consuming when accessing files. Search tools such as simple search, custom search or DLQ editor are not widely taken advantage of by the end users. One reason may be that these tools are not as intuitive as other search tools more widely in use within the MS Office environment.

Summary documents (e.g. Clinical Study Report, Summary of Clinical Safety, Summar y of Clinical Efficacy, etc.) will contain all single file outputs programmed, compiled into a report. They become available only some time after the production of the outputs due to time needed by Medical Writing and Publishing group to generate these documents. Tables, listings and figures produced may spread across multiple summary documents.

To bridge the gap that results from searching for outputs produced in any of the aforementioned referencing documents, the team was looking for a tool that:
- lists all outputs that have already been produced and validated
- flattens out the vertical structure of the programming environment
- pulls together files spread out across multiple subdirectories in the document repository


Output catalogue [Information collected from Programming environment / *Document repository*]

RA1	cabinets/.../[project]/[study]/[subdirectories...a]/Text tables/t10_01.rtf @@\main\1	<hyperlink> *Table 10.1*	*Summary table*
RA1	cabinets/.../[project]/[study]/[subdirectories...a]/Text listings/l15_01.rtf @@\main\7	<hyperlink> *Listing 15.1*	*Data listing*
RA2	cabinets/.../[project]/[study]/[subdirectories...b]/Text tables/t10_01.rtf@@\main\7	<hyperlink> *Table 10.1*	*Summary table*
RA3	cabinets/.../[project]/[study]/[subdirectories...b]/Text tables/t10_01a.rtf@@\main\2	<hyperlink> *Table 10.1a*	*Summary table – Subgroup*
RA3	cabinets/.../[project]/[study]/[subdirectories...b]/Text tables/t19_01.rtf@@\main\4	<hyperlink> *Table 19.1*	*Frequency table*
RA2	cabinets/.../[project]/[study]/[subdirectories...b]/Text listings/l15_01.rtf@@\main\9	<hyperlink> *Listing 15.1*	*Data listing*

In short, the team was looking for a simple, easily to update tool that can be used by programming and non-programming team members which serves as a catalogue of outputs produced during the submission.

WHAT WE DEVELOPED

Version Control in a nut-shell

ClearCase is a program for software configuration management and version control. Each ClearCase VOB stores version-controlled file system objects, termed elements. An element is a file or directory for which ClearCase maintains multiple versions. The versions of an element are logically organized into a hierarchical version tree. Each version of an element has a unique version-id, which indicates its location in the version tree. Any version of an element can be uniquely specified by appending its version-ID to its standard pathname. [2]

The ability to reference any version with a standard operating system pathname is a very important ClearCase feature. Within the in-house standard programming environment not only programs for generating outputs (e.g. tables, listings) which form part of a regulatory submission report are set under version control. Also utility files that that manage and log the file transfers to the document repository are version controlled.

Content Management in a nut-shell

Before word processors, all paper documents had to be stored in a folder which was then stored in a physical filing cabinet. The onset of word processors enabled electronic storage. Networks enabled users to share each others information. The information was still not accessible as the information was not in one central repository. This prompted the need for an Information Management Solution.

A document management system coordinates the changes and availability of business critical documents on a global basis. Users store documents and other objects in a central repository called a docbase. A docbase stores two kinds of information for a document:

Content	Text, Graph, Spreadsheet	information _in_ the document
Properties	Name, Title, Owner, Status	information _about_ the document

The characteristics of both systems, i.e. the storing of historical information in the programming environment (GPS/ClearCase) and the availability of document properties in the document repository (CREDI/Documentum Docbase) can be leveraged to create a tool to catalogue all single file outputs.

Technical workflow

Obtain information from programming environment (GPS):

1) Identify project directories that are active and/or part of the submission project.
2) Retrieve detailed information on upload to document repository (CREDI) by screening through utility files and all their historic versions. This is done via SAS in an automated fashion using x command and SAS macro language. [3]

Obtain information from document repository (CREDI):

1) Extract file properties of documents in relevant subdirectories in the document repository as identified in previous step.

Combine information from programming environment (GPS) and document repository (CREDI):

1) Read in data from both sources (*.txt, *.csv)
2) Combine information and create SAS dataset (CREDI information on directory and file properties, GPS upload information)
3) Manipulate data (Create hyperlinks, prepare master spreadsheet, prepare summary reports, etc)
4) Create reports using SAS ExcelXP tagset

Why SAS Excel XP tagsets?

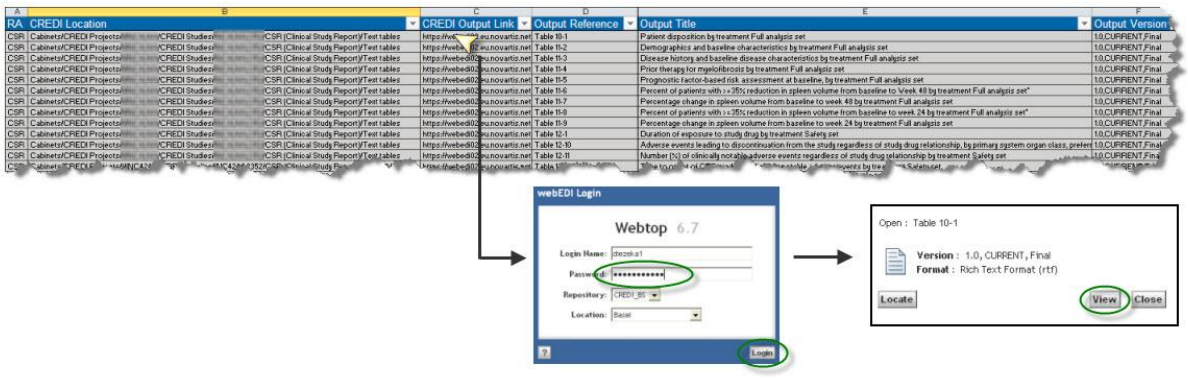
One of the requirements of the tool was that it should be simple and easy to use. Everyone should be able to access the information without installing a new software or having to perform additional training. As the extraction and combination of data was done using SAS it was a natural choice to use the power of SAS ExcelXP tagsets when creating reports.

The ExcelXP tagset creates an XML file that, when opened by Excel, is rendered as a multi-sheet workbook. All formatting and layout are performed by SAS; there is no need to "hand-edit" the Excel workbook. The users can simply use Excel to open the file created by ODS. [4]

As it looks and feels like an ordinary MS Office spreadsheet the acceptance by the user was guaranteed.

Where are we today?

Currently about 1750 single outputs files from 95 documentum directories and 15 reporting activities are listed in the master spreadsheet, distributed to the team on a monthly basis.



Information can be searched and filtered as needed. Files and directories in the document repository can be directly accessed by clicking on hyperlinks.

None of the single output files are duplicated. The tool merely references existing copies in the computing environment. No additional copy of the output file is generated and the user continues to use the computer system as intended. The concept of traceability is maintained and the company-wide, pre-defined standard workflow kept.

WHAT WE GAINED

With the master spreadsheet a one-stop-shop for outputs is now available to the whole team. In the past a user would need to search through multiple output planning documents in order to identify the output needed, log into CREDI, click on and navigate down to the main project folder and sub-folders (up to 10 levels down in the directory structure), find the required output file in a list of documents in that folder and click to open. Now the user can simply search for the outputs in the spreadsheet by using the filter functionality, click on hyperlink, sign in (only once per session), chose to view or locate the file and the file opens or its location is displayed.

Many aspects of set up and conduct of a reporting activity can now be monitored in an automated fashion. Frequent reports help to ensure consistency of programming across reporting activities and address potential issues at an early stage. Onboarding of new team members is sped up with regards to understanding the project and programming environment. Training needs can be identified and addressed on a team and individual level.

Experiences

The idea of the output catalogue was put forward and shared by all line functions. Every line function has a different need and use for the data extracted from the two systems. The information collected from the systems can be split into sub-sets according to needs and provided accordingly in order to avoiding overwhelming the users with un-necessary information:

Line Function	Information provided	Information used for	Impact
Statistical Reporting - Project level	Access to all information extracted from both systems, Access to summary reports on process adherence, Access to master spreadsheet	- project status and process adherence checks can be automated - monitor process adherence and project status done more consistently across project	- quality, consistency of SR deliverables - process awareness and training within the team
Statistical Reporting - Reporting activity level	Access to master spreadsheet	- checking if a newly requested output is truly new - identify programs/macros/outputs to be reused within and across reporting activities - ensure consistency of outputs by referring to tables already done	- increased speed - increased consistency - increase quality due to re-use of program and macros code
Biostatistics	Access to master spreadsheet	- rapid access and overview of already delivered outputs - check if an output request is truly new - verification of outputs across different reporting events - draw shells based on existing outputs	- speedier cross verification - quicker generation of shells - consistency of shells across project - consistency of analysis across project
Outside BIOS&SR	Information of master spreadsheet embedded in communication, e.g. subsets of spreadsheets, hyperlinks to documents in CREDI in order to avoid misleading conclusions outputs are presented in context	- identify existing outputs that may answer a question - identify new outputs to be generated - reference outputs by hyperlinks instead of distribution of physical files	- quicker access to outputs produced - shorter turn-around time for newly requested outputs - easier use of document repository

Starting of with information on outputs produced by one group for one reporting activity the tool was quickly extended to cover all outputs produced for the submission. Due to a common working environment and standardized processes the collected information includes not only reporting activities from Full Development, but also files produced by Clinical Pharmacology and Global Medical Affairs. It brings closer together the different programming groups and allows efficient sharing of tools and programs. At the same time it ensures consistency across the project.

DISCUSSION

Searching for an optimized solution to access single file outputs helped to improve understanding of the in-house systems, recognize potential and current limitations of different systems and develop a better understanding of wants/needs from different line functions within the teams.

The solution put in place addresses the need of the team for a simple tool that catalogues all outputs. It is easily adapted to include upcoming reporting activities and new submission projects. As it uses information stored in the in-house systems it highly depends on these systems in place and a change of those would impact the underlying data extraction and data manipulation process.

Time spent to explore the computing environment, to develop the tool and to provide monthly updates is minimal in comparison to the time saved across the project short- and long-term.

CONCLUSION

Programming does not stop at the clinical data level. Thinking outside the clinical data reporting box, SAS programmers can create new opportunities to work together and ease the daily burden for everyone on the team.

REFERENCES

[1] EMC Documentum
http://www.emc.com/enterprise-content-management/documentum-webtop.htm

[2] IBM Rational ClearCase
http://publib.boulder.ibm.com/infocenter/cchelp/v7r0m0/index.jsp?topic=/com.ibm.rational.clearcase.cc_ref.doc

[3] Executing Operating System Commands from Your SAS Session
http://support.sas.com/documentation/cdl/en/hostunx/63053/HTML/default/viewer.htm#p0w085btd5r0a4n1km4bcdpgqibt.htm

[4] An Introduction to Creating Multi-Sheet Microsoft Excel Workbooks the Easy Way with SAS®
Vincent DelGobbo, SAS Institute Inc., Cary, NC
http://support.sas.com/resources/papers/proceedings12/150-2012.pdf

ACKNOWLEDGMENTS

The author would like to thank the Basel BIOS&SR team (Adam Boyd, Norbert Hollander, Viktoriya Stalbovskaya, Alberto Montironi, Mark Buckland) for their constant input and feedback.

CONTACT INFORMATION

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

Katja Diezel
Novartis Pharma AG
Postfach
CH-4056 Basel
Work Phone: +41 61 69 67499
Fax: +41 61 3240046 (direct)
Email: katja.diezel@novartis.com
Web: www.novartis.com