Paper Number: PharmaSUG China 2015-05

Getting Clouds Moving across the Pacific - a case study on working with a Chinese CRO on SAS ® Drug Development

Chen Shi, Santen Inc., Emeryville, CA

ABSTRACT:

California has been in a long drought and some parts across the Pacific have been suffering from storms during the past 2 years. Why not move the clouds over? This can be a dream but not only a dream if we know how to manage the clouds.

In this case study, we will share our experiences and lessons learned about delivering study packages on the SAS® Drug Discovery (SDD) platform with a newly on-boarded Chinese CRO. This paper will elaborate on topics including project setup, training, scheduling, and communications as well as debugging, quality control and project summarizations. It was a fun and challenging project as it proved that SDD can be a useful tool for enabling cross-continent teams to work together.

INTRODUCTION:

This was a Phase 2b trial Clinical Study Report (CSR) package done completely on SDD 4.2.1 where Santen and CRO programmers worked shoulder to shoulder for 3 months. The latest study data were uploaded to SDD weekly at the beginning stage of the study and daily at later stages by Santen up. CRO users were set to have read and write access to the assigned study folder. The project expectation for the CRO was to produce SDTM and ADaM level data sets, as well as TFL outputs, to be one button click away from final delivery. The study quality checking and summary process as well as the final run after unmasking was done at Santen. All ad hoc requests were handled in-house and are not discussed in this paper.

STAGE 1: SETUP AND TRAINING

FOLDER STRUCTURE SETUP:

Before the beginning of this project, CRO programmers were trained on folder structure by Santen SOP. Thanks to SDD's "Upload and Expand" button, a zipped folder containing standard folder structure was uploaded. Folder structures for each study were kept consistent after uploading and unzipping.

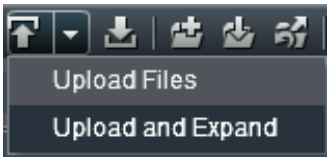


Figure 1: Upload Buttons

USER ACCESS SETUP:

A user group was created at first to host all the member users of a specific study. Unlike the Windows environment, SDD Study A members cannot automatically access Study B if he or she is not a member of Study B or an administration user. This setting prevented a lot of cases like misplacing files, overwriting files, and assured independent programming quality. Admin users can further set folder access for Programmers and non-programmer users which helps different functions to stay focused in their area and avoid distractions.

Under the Administration tab, Admin users can also set different privileges for each user type in a study. For example, the lead programmer will often have privileges such as monitoring a scheduled batch run, deleting files and managing files checked out by other users which other study programmers might not be assigned to. Members outside the programming team, such as data managers, statisticians and publishing team will usually have read-only access to most of the files. Figure 2 summarizes the different setups used for each type of user.

Figure 2: User Type Setup

Role	User Access to SAS session	Responsibility
Lead Programmer (CRO)	Read and write access, Batch job creation	Program and delivery quality control, create batch job file, offer solution and documentation of SAS issues.

Lead Programmer (Santen)	Read and write access, Schedule batch jobs, Administrator role	Upload/download raw data, run and schedule batch job file, delivery quality control, offers solution and documentation of SAS issues.
Study Programmer	Read and write access	Program and check- in of programs
Statistician/Data Management	Read only to all SAS session	Edit doc Files, Upload/download, Outputs reviewer, Sign-off on documents.

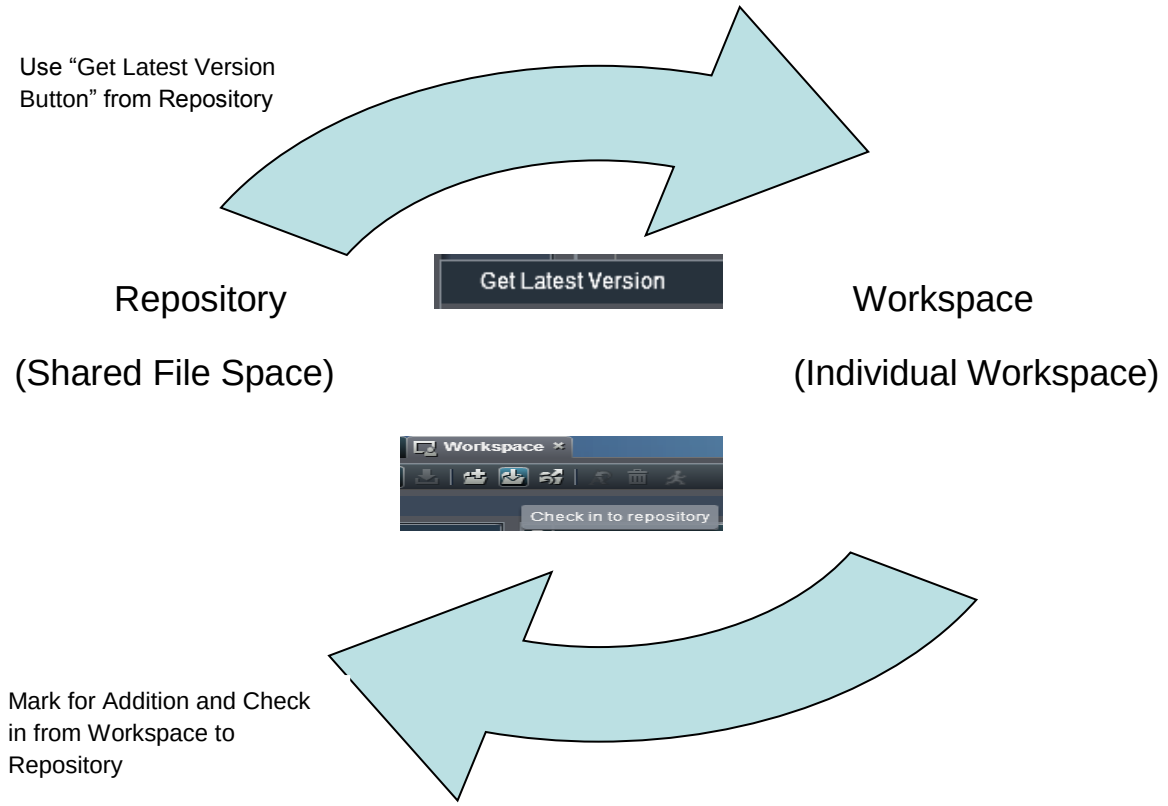
STAGE 2: CLOUD MANUFACTURING

HOUSEKEEPING RULES:

Different from a local file server or shared remote desktop (drive), SDD has a repository (a shared space) and a workspace (which is unique to each user). This helps to enforce version control, (leave in a priority) when multiple users from both sponsor and CRO are working in the same environment.

First, we used a delivery tracking sheet to assign tasks to different users to avoid overlapping. Every user used "Get Latest Version" instead of checking out the folder when starting mirroring of the same repository to workspace. At the end of each workday, programming users check- in the updated programs they have been working on. (This makes sure everybody gets updated on where we left off from the last update). If we make the analogy that we are in a "cloud manufacturing" factory, all the crew members need to follow the "house rule" which translates easily to "put things back to where you found them". Figure 3 is a snapshot of our housekeeping process.

Figure 3: Repository-Workspace Check in/out workflow



THINKING GLOBAL:

Now we had this great advantage to run programs 24/7 and work spontaneously anywhere with web access. Working across time zones still gave us a good learning experience, which you will never experience in quite the same way if all of your team is local.

We also ran into the problem of synchronization delay on SDD where both the CRO side and Santen side did not have the same real-time results. This was discovered at a later stage of the project when both teams had more overlap on real-time problem-solving. For example, the CRO side has dm.sas updated, but the Santen side would not see the updates until their current timing reached the timing of the CRO update which is always more than 10 hours ahead. (This problem was not discovered previously when Santen programmers were always checking the updates the next day.) We later decided to shift to Greenwich Mean Time (GMT) on both system timing and server timing to assure we both are always looking at the same page at the same time.

Figure 4: Time zone lagging behind results to delay of synchronization



ACTING LOCAL:

Another issue we met was an extremely low speed and unstable internet connection from our Chinese CRO to the SAS® Drug Development server. It took the longest hours to upload some files, while SAS sessions were logged off in the middle as well. While waiting for SAS® Institute and Santen IT to give out a solution, all the uploading and downloading was done on the Santen side before this issue was fixed. This brought to our attention potential risks of failure to connect in other emergency situations. As SDD is web-based, users can simply log on from anywhere. A solution will be using a VPN or changing to another IP address to connect to the SDD server if the Internet connection breaks down in China.

As a combined effort, both Santen and CRO project leaders encouraged every team member to speak up about what they met and discovered along the way when using SDD. We invested time and efforts intensively in the following project meetings to bring the teams together and share knowledge about what they learned and struggled with, also to melt down the culture barrier of communication. One best way to host meetings is to be as visual as possible. “One picture is worth a thousand words.” Whether it is program debugging or output formatting, it is always efficient to show screenshots, a process demo, or a live webinar to share the information in real-time using visual input so we can share as much as information as possible.

STAGE 3: EZ-JOB AND EVERYTHING SCHEDULED

We now come to one of my favorite parts of SDD. SDD allows user to create job files to run programs. We developed a macro program, %M_CREATEJOB, where each program can call this macro to create a job file with same name as the program under same path. As we finished programming all the individual data sets, we put all SDTM source programs into one master job file which ran all the SDTM programs and thus created a master log. There is a small bar (Figure 5) capturing Error and Warning counts on the bottom left corner of each log file which is a very helpful tool for the final delivery wrap-up.

SDD also has a Schedules panel where you can schedule the jobs to be run at singly or concurrently. For example, in Figure 6, we scheduled all the validation side jobs 10 minutes after the scheduled source side job. This tool gave out a very clean timing stamp after the batch run. Besides checking the timing stamps, the little signs under the “Workspace status” column will tell you the consistency between your workspace and repository. So this is the final step before checking in to repository and locking down the whole delivery. As a follow-up, you can also set up an email alert to catch any changes to important study documents or a completed (locked) delivery.

Figure 5: Error/Warning counts in log file

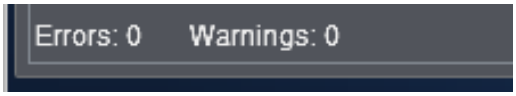
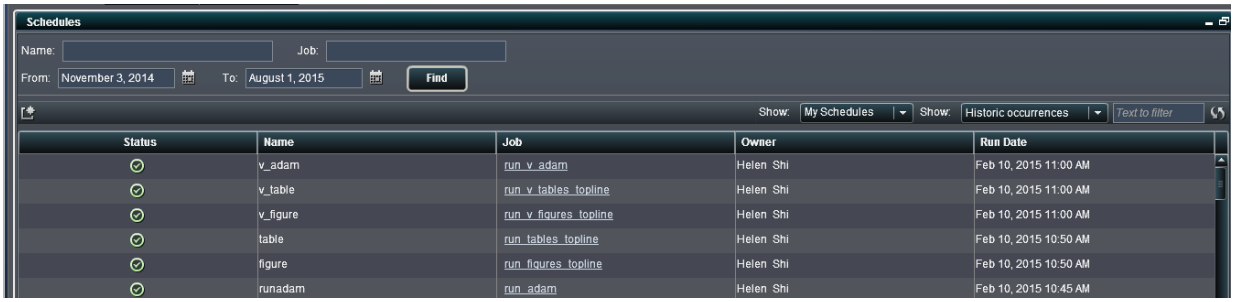


Figure 6: History of scheduled jobs



At the time of this paper, Santen is still working collaboratively with SAS Institute about SDD tools to make both programming deliveries and process tracking more user-friendly.

CONCLUSION

In this project, we discovered some gaps in mastering the new tools together with quality control of outputs, an upgraded procedure map and developed time management skills that fit into a global scale. It was a steep learning curve but very fun one to go through with a young and spirited CRO team on SAS® Drug Development.

We together had proved that SAS® Drug Development works for a cross-Pacific study delivery and can be used in the future. We are very happy that “clouds” can be moved to rescue resource limitations when needed. Now it’s time. Let it rain (run)!

CONTACT INFORMATION

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

Name: Chen Shi
Enterprise: Santen Inc.
Address: 2100 Powell Street, 16th Floor
City, State ZIP: Emeryville, CA 94608
Work Phone: (415) 268-9078
E-mail: hshi@santeninc.com
Web: https://www.linkedin.com/pub/chen-helen-shi/1a/507/847