



»» TRANSFORMING PROMISING IDEAS INTO COMMERCIAL REALITY

SAS in clinical trials – A relook at project management, tools and software engineering

Sameera Nandigama - Statistical Programmer
PhUSE 2014
AD07





Introduction

- The software development industry has adopted many new tools and methodologies to aid project management, software design and implementation.
- These tools improve the efficiency and communication in the software development world, in-turn making cost savings.
- The paper attempts to look at some of these in brief and the potential value these could bring to SAS clinical programming.



Technical Debts

What is a technical debt?

Is an accumulation of small low priority tasks.

- Generally, it is a temporary fix or shortcut that is taken immediately to deal with an issue.
 - E.g. a short term SAS code fix that is to be replaced by a more robust solution later
- Replacing the shortcut with a proper solution itself becomes a low-priority task for the future.

Why is technical debt undesirable?

- Causes unbalanced workloads and pressures
- Can result in a large volume of low priority work that as a whole becomes a risk to the project itself.



Technical Debt Management

How to manage?

- The key to manage tech-debts is:
 - › To ensure that the accumulated debt is effectively tracked
 - › To ensure that some amount of small low priority tasks are continually disposed.
 - › To ensure that the backlog of work, irrespective of priorities, remains manageable.

Tools that help manage

- Issue management tools .



Issue Management Tools - JIRA

An issue tracking system that manages and maintains lists of issues

The most commonly used makeshift tool - Spreadsheets

Difficulty with spreadsheets

- › Accidental changes can happen while using spreadsheets
- › Becomes increasingly complex with more and more entries.
- › Difficult to track or search for particular issues in a large spreadsheet.
- › The history of changes is not preserved on a spreadsheet.
- › Report generation with spreadsheets can often be clumsy.
- › Visibility management is not possible with a spreadsheet



JIRA - Features

JIRA – Popular project and issue management tool

JIRA users are arranged into groups

Privileges are assigned based on the user group

The groups could be

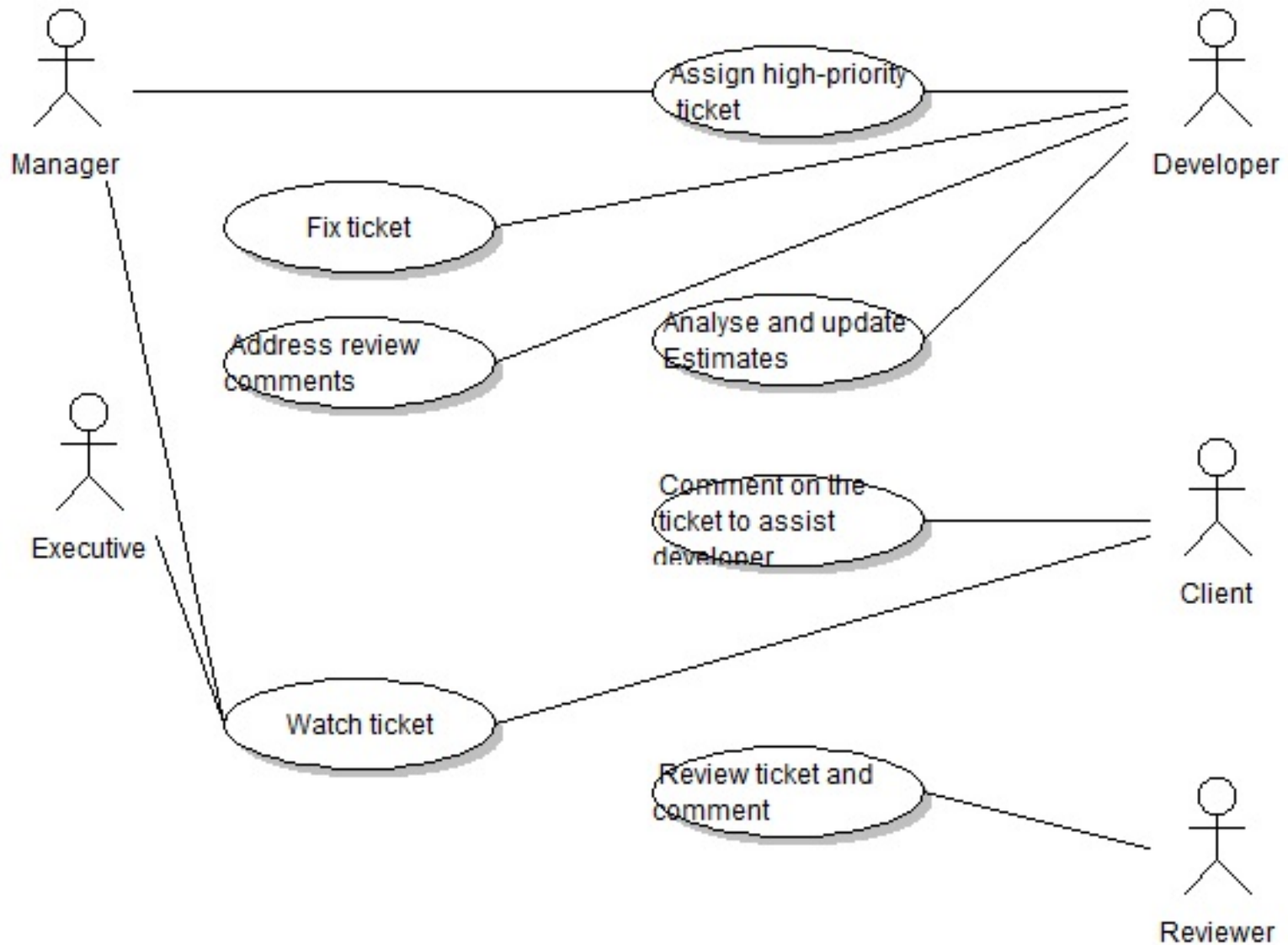
- Project manager
- Team lead
- Developers
- Scrum master
- Testers
- Statisticians.

The issues are generally categorized as

- Tasks
- Bugs
- Support

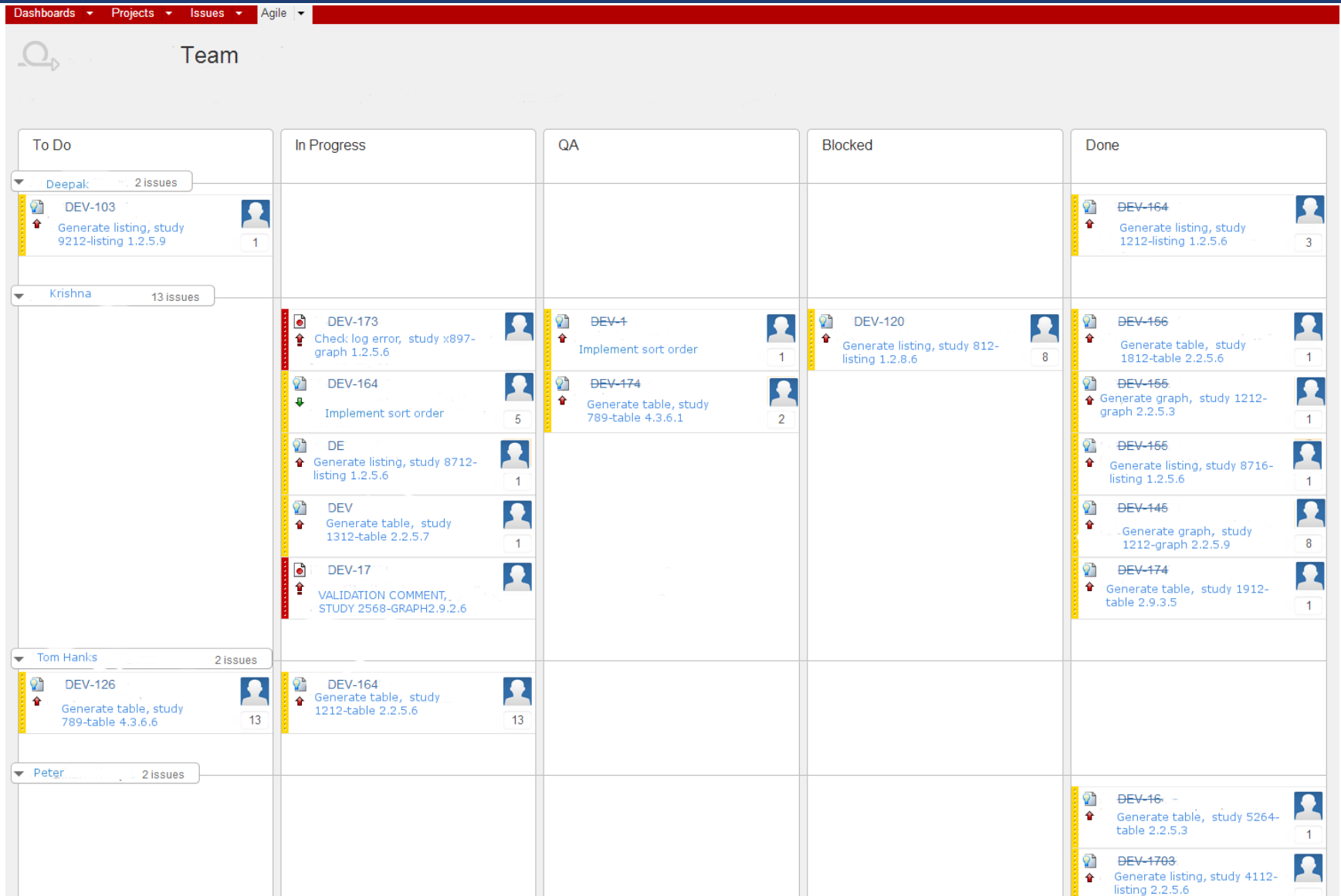


JIRA - Workflow





JIRA - Workflow dashboard





JIRA – Comments and discussions

Dev-1234

Implementation of XYZ plots

Edit

Assign

Assign To Me

Comment

More Actions ▾

Close Issue

Start Progress

Workflow ▾

Share

Views ▾

Plots cannot be generated with estimates being 0. @user2, can you have a look?

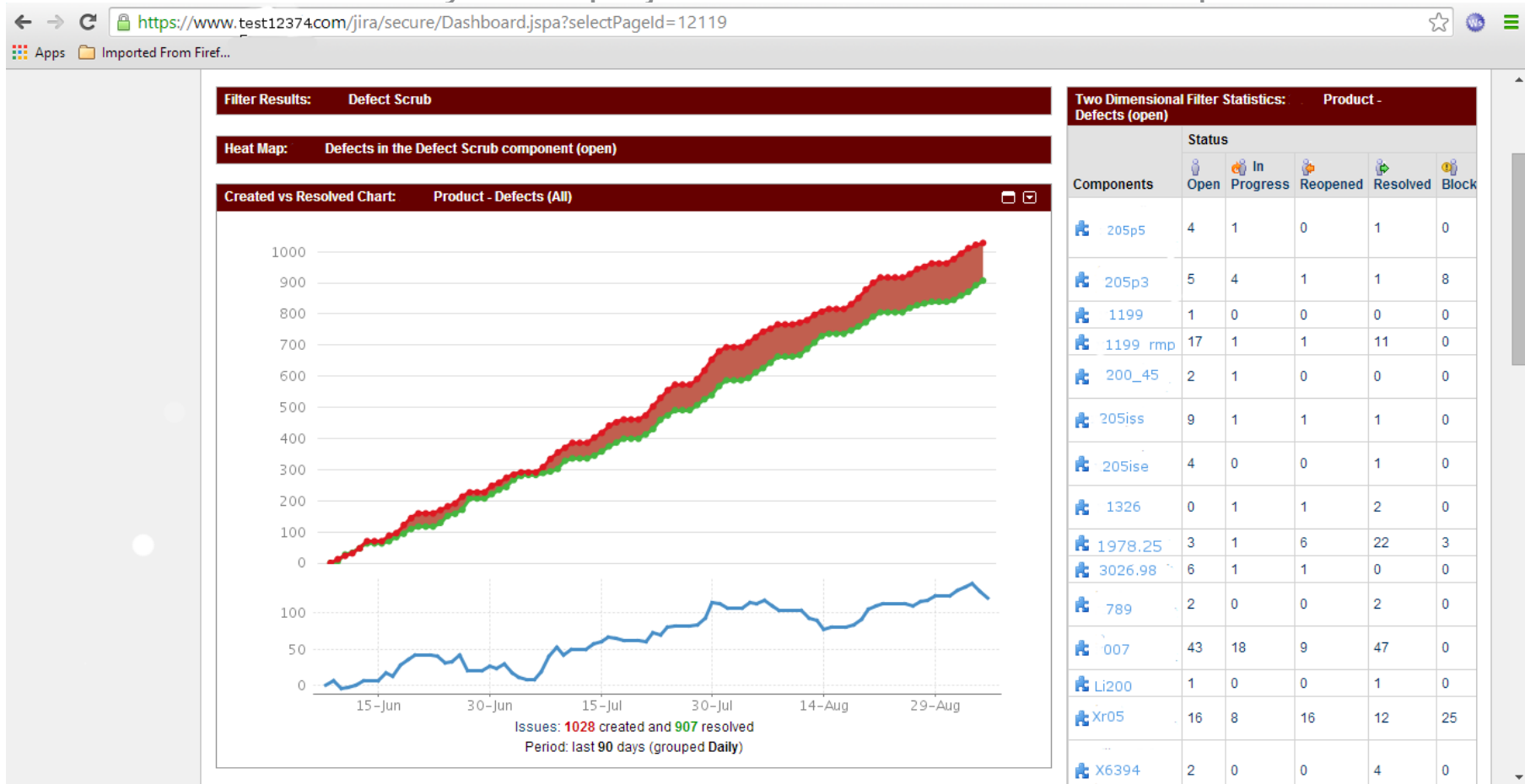
-  user2@inventiv.com, added a comment - 31/Jul/14 3:35 PM 
- I have tried with a very low value and it seems to work.
- @stats1, do you agree with the idea?
-  stats2@inventiv.com, added a comment - 01/Aug/14 11:39 AM
- Yes, the approach is right. I would suggest go real low
-  user1@inventiv.com, added a comment - 01/Aug/14 3:37 PM
- Went for very low value, that seems to produce a very good output
- @user2, do you concur?
-  user2@inventiv.com, added a comment - 05/Aug/14 4:20 PM
- I do concur that output looks good.
-  user1@inventiv.com, added a comment - 06/Aug/14 11:18 AM
- The change is on repository. Moving ticket to QA. @QA, can you validate please?
-  qa@inventiv.com, added a comment - 06/Aug/14 1:15 PM
- The code produces good outputs, will accept it in current form.
-  qa@inventiv, added a comment - 15/Aug/14 4:53 PM
- Closing the ticket as done.



JIRA – Visibility management

Solves the problem of visibility for all .

- Executives will be able to see a graph representation of progresses
- The extent of visibility of the project is based on the role of the person





Source Control Systems

A version control system manages changes to code and documents.

- Provides protection against accidental deletion of important files
- Managing code/report releases becomes simpler.
- Helps developer store his transient code.
- Full history of changes to any file is preserved.

Types of version control systems:

- **Centralized version control systems:** There is a central repository and all the clients synchronize with it, for example, CVS, SVN etc.
- **Distributed version control systems:** Follows a peer-to-peer model, every copy is a repository and pulls and pushes can happen between peers, for example, GIT, Mercurial etc.



Job Automation And Automatic Test Reports

JENKINS

- Is a web-based, free, open source tool that can be installed on any Windows or Linux servers.
- Can run any Windows or Linux scripts, commands or applications.
- Runs automatically based on configured rules and produces summary reports.
- Can be aligned with other tools like SVN and JIRA

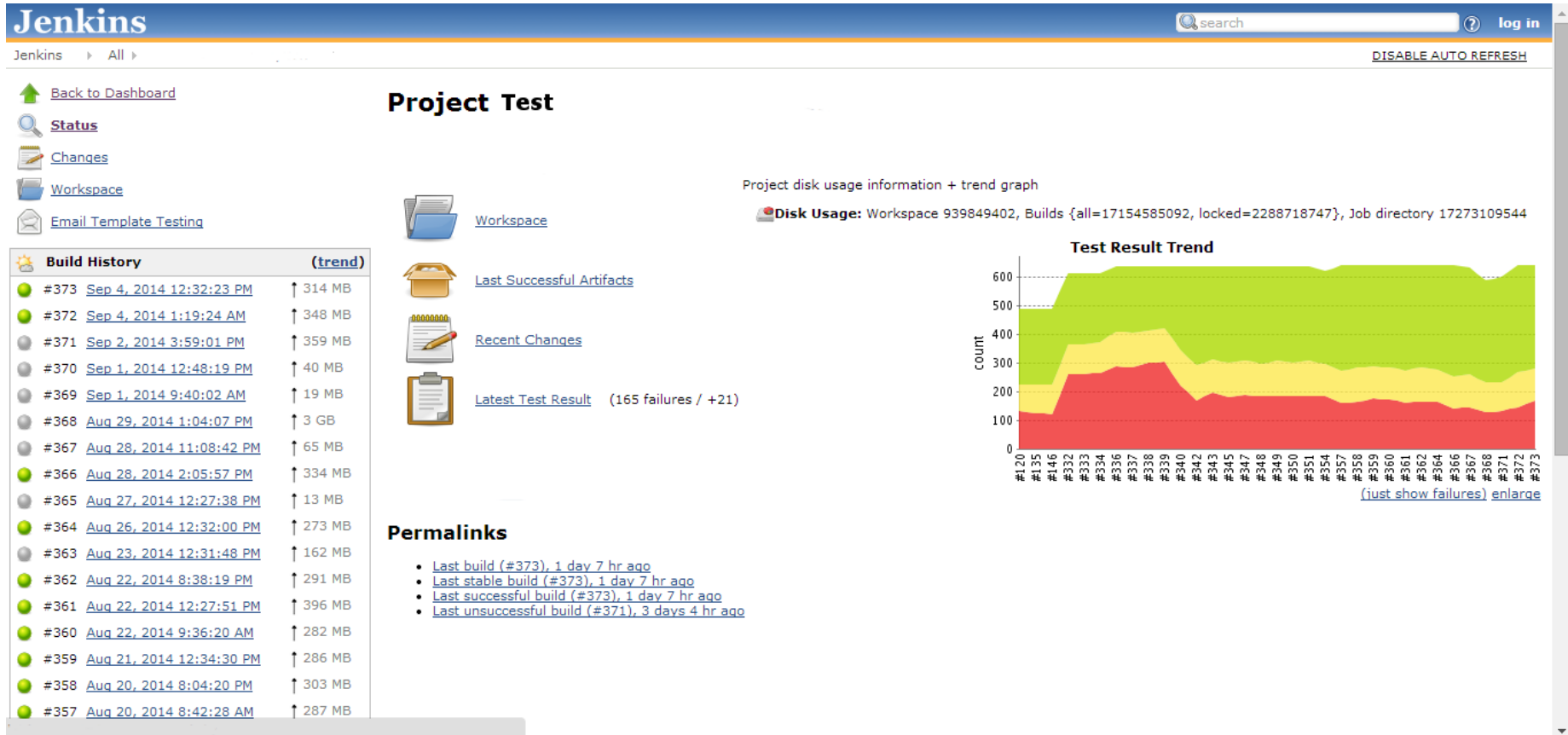
Potential use for Jenkins in SAS programming :

- Perform automated 'Good Programming Practice' check.
- Compare the outputs of both the primary and validation programs.
- Automate output count.
- Performs log check and notify via email.
- Verify that outputs match the templates.



Jenkins Reports

Reports trend towards green when programs are implemented correctly





Documenting Code and Doxygen

Tool for used for writing software reference documents.

- The documentation is written within code - easy to keep up to date
- Doxygen can cross reference documentation and code
- Can generate documents either in HTML, PDF or RTF
- Used in many programming languages

How is Doxygen used in SAS programming?

- Write their code using Doxygen standards
- Programs reside in a directory called “Project” on the “C” drive
- Doxygen command runs on command prompt to generate documentation

Benefits

- Documentation writing happens alongside SAS code development
- Document then changes with the comments and hence is always up to date
- Results in less effort in knowledge sharing and improved efficiency



Documenting Code And Doxygen

SASUnit Examples: File List x

← → ↺ sasunit.sourceforge.net/v13/doc/files.html ☆ ☰

SASUnit Examples Version 1.3.0

Main Page Modules **Files**

File List

▼ SASUnit Examples

- Modules
- ▼ Files
 - ▼ File List

_abspath.sas
_assertlibrary.sas
_asserts.sas
_checklog.sas
_checkscenario.sas
_closehtmlpage.sas
linux/_copydir.sas
unix_aix/_copydir.sas
windows/_copydir.sas
_copyfile.sas
_createtestsubfolder.sas
_deletescenariofiles.sas
_delfile.sas
_deltempfiles.sas
_detectsymbols.sas
linux/_dir.sas
unix_aix/_dir.sas
windows/_dir.sas
_doc.sas
_doxygroups.sas
linux/_escapeblanks.sas
unix_aix/_escapeblanks.sas
windows/_escapeblanks.sas

File List

Here is a list of all documented files with brief descriptions:

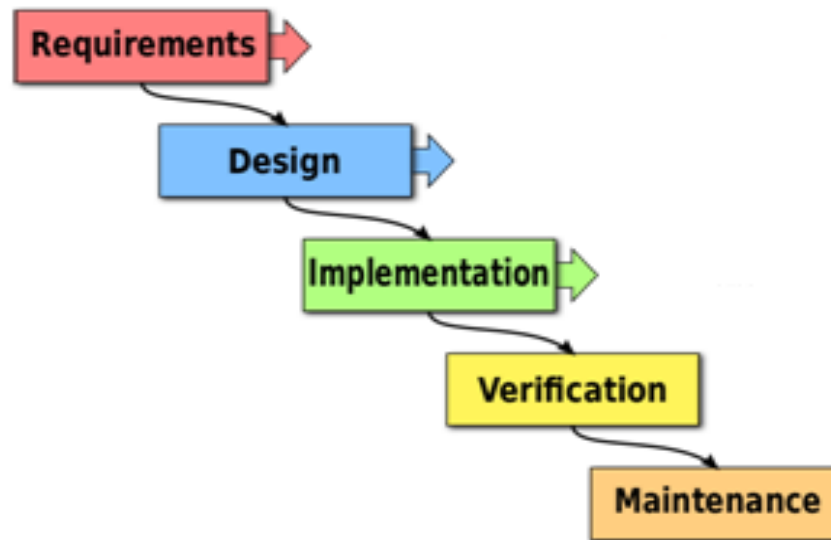
 _abspath.sas	Check whether &i_path is absolute or empty. If not, append to &i_root
 _assertlibrary.sas	Main routine of assertLibrary
 _asserts.sas	Called by assert macros, fills table tst
 _checklog.sas	Check log for errors or warnings
 _checkscenario.sas	Determine whether a test scenario has to be executed
 _closehtmlpage.sas	This macro is part of the HTML speedup solution and "closes" an open HTML page
 linux/_copydir.sas	Copy a complete directory tree. Uses Windows XCOPY or Unix cp
 unix_aix/_copydir.sas	Copy a complete directory tree. Uses Windows XCOPY or Unix cp
 windows/_copydir.sas	Copy a complete directory tree. Uses Windows XCOPY or Unix cp
 _copyfile.sas	Copy file byte by byte
 _createtestsubfolder.sas	Creates test subfolders
 _deletescenariofiles.sas	Deletes all appendant files in the log, tst and rep folders. Called before a scenario is being executed in order to avoid disused files
 _delfile.sas	Delete an external file if it exists
 _deltempfiles.sas	Delete all SAS datasets in the form WORK.DATAxxx, see tempFileName.sas
 _detectsymbols.sas	Determine the language dependant symbols used for NOTE, ERROR, WARNING in the SAS log
 linux/_dir.sas	Generates a dataset with the names of all files in a directory or directory tree. Wildcards may be used to specify the files to be included
 unix_aix/_dir.sas	Generates a dataset with the names of all files in a directory or directory tree. Wildcards may be used to specify the files to be included
 windows/_dir.sas	Generates a dataset with the names of all files in a directory or directory tree. Wildcards may be used to specify the files to be included



Software Development Life Cycle (SDLC) In SAS Clinical Trials

WATERFALL

- The waterfall model is a sequential design process that progresses to the next stage only after current stage is completed.
- Cannot easily adjust to changes of requirements at an advanced stage of the project.



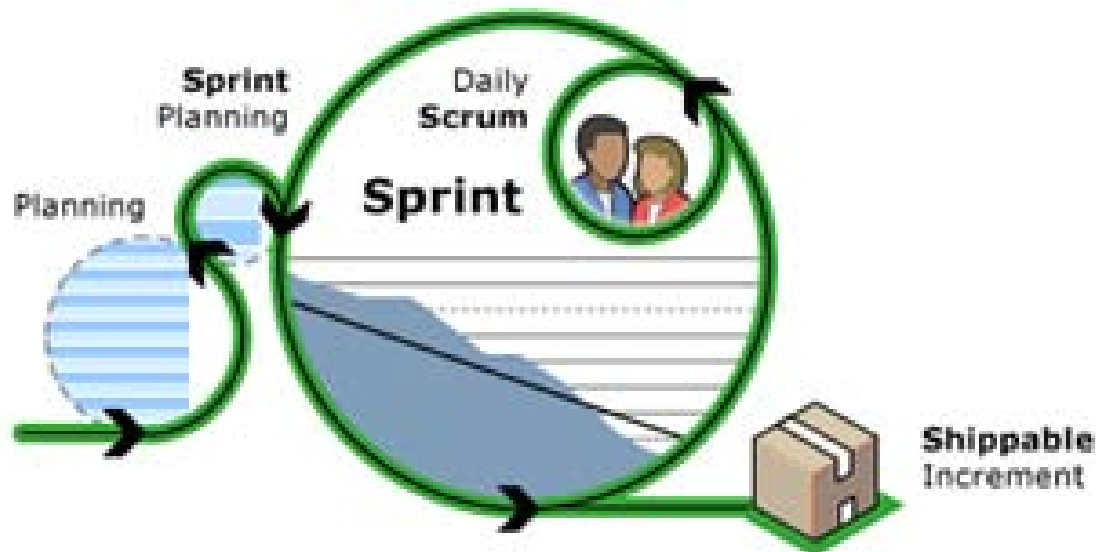


Software Development Life Cycle (SDLC) - Agile

Agile “think a little, do a little, deliver a little”

SCRUM - has 4 stages

- **Project analysis planning:** The initial Project Statistical Analysis Plan (PSAP) is planned and revisited during any stage of the project
- **Sprint planning:** A sprint is small planned incremental development
- **Daily scrum:** Is the daily meeting of the team to monitor the progress
- **Incremental release:** The end of the sprint release





Benefits Of Agile

Benefits of Agile

- Parallelization of work. Reports are developed along with analysis datasets
- If the analysis datasets are not moving forward, the developers can plan some other project into their sprints
- Achieve results in a short time with high quality
- SCRUM offers a lot of flexibility
- The complete output report code is produced at the same time as the final version of the analysis datasets



UML –Unified Model Language

- Graphical and visual representation model
- Makes solution easy to understand
- Converting a model to an implementation can be trivial

Types of UML diagrams:

Structure diagrams - Represent the various components of a product and the relationship between the components.

Behavior diagrams - Model the behavior and the interaction of the system among its components.

Benefits of UML

- Pictorial representation for clear communication, quick development and lower costs
- UML modeling facilitates clean design.
- System maintenance costs will be lower because of reuse of UML designs.
- Learning for a new developer will be easier;
- Efficient communication with programmers and outside contractors



Conclusion

- Use of the tools like JIRA, Jenkins, Doxygen and SVN can improve efficiency and thereby bring cost benefits to the clinical trial industry.
- Adopting methodologies like Agile can bring flexibility and improve overall time needed to complete a study.



Thank You And Any Questions?

