

Automation Process of testing scripts in the Aerospace Industry using VBA

Nelson A. Granda Marulanda
Manufacturing Engineering
Edgar Torres, Ph.D.
Industrial Engineering Department
Polytechnic University of Puerto Rico

Abstract — *The use of computers and their computer languages, created to help with the manually tasks done by an operator has become now days really imperative. The evolution of the software used on normal work environment has become tremendously fast and a lot of their advantages are still unknown by many users. This project highlights the use of the Visual Basic for Applications (VBA) language as an important tool in the creation of software code known as a script. This code is used in the testing environment of the Federal Aviation Agency (FAA) regulations. During this project we discuss how the use of the Six Sigma tools specifically the DMAIC methodology will help to change the process of the creation of these scripts. This new process will speed up the creation of the scripts and reduce the errors previously done, when there was manually created, this resulting in cost reduction and quality improvement.*

Key Terms — *Scripts, Six Sigma - DMAIC Test Procedures (TPs), Visual Basic for Applications (VBA).*

PROBLEM STATEMENT

In the fast pace environment that the Aerospace industry is working nowadays, driven by a high demand of testing and the certification process of the new technology by the Federal Aviation Agency (FAA), it is compulsory to come with new ideas of process automation in order to remain competitive in the industry.

Due to the necessity of the aerospace industry, this research will use various tools of process improvements, and the use of visual basic for application (VBA) as an automation tool to attack the testing-code generation and analysis. This research is expected to lead to a reduction of the

tedious manual work resulting in less man-hour efforts and better quality of the work done.

Research Description

When a new Electric Power Generation and Start System (EPGSS) is design their control system is very robust and complex, the use of a simulation software capable of emulate most of the real system is a great “real-time” tool for development and verification of the control system. In order to verify that the system requirements are correctly implemented and integrated in the control system, a vast number of Test Procedures (TPs) need to be developed, these TPs will try to verify that each one of the system requirements are met, and that the integration between all the EPGSS systems will not interact in a wrong way. The data input used to generate the testing code, or scripts, can be extracted from each one of the tests paragraphs mention on the TPs. This data can be manipulated using excel, which is the main application used with VBA, the data is converted into the input to be used by the VBA generating code and a Script Template previously define. When all the scripts on Tool Command Language/Tool Kit (TCL/TK) are generated they go to be run on the simulation environment to generate test results files, then this results will be used to validate the expected results of the TPs vs. the results obtain from the control system simulated. The data from the test results also is extracted by another VBA application into an excel file where the analysis of these results is more user friendly to manipulate and lean methodologies can be easily applied.

Research Objectives

With the completion of this automation project is expected to save up to 70% of the time used

when the script is manually generated, also is expected to reduce in 80% error created by the human factor, therefore increasing the quality of the results.

By accomplishing these two objectives an additional benefit is derive, since the cost of producing these scripts is base on man-hour effort, by speeding up the process and maintaining, or increasing, the same quality of work we can derive a very important objective: cost savings, which is strictly connected to the amount of time saved by the completion of this automation process.

Research Contributions

After the development and completion of this project, is expected to have a tremendous cost savings as a main contribution. Other side contribution will be the development of a standard procedure document for this new testing-code generating method. This document can be easily followed and adjusted when other manually code generating tasks, with similar expectations is requested, saving set-up time for the new project. Also this project can be used as an example or pilot, when new project proposals are submitted for consideration.

LITERATURE REVIEW

First there is a defined methodology call Six Sigma which can be used to solve huge diversity of problems in the industry. This project will empathies on the problem solving process steps propose by the Six Sigma method. In second place but still important to understand is the code language and tools used to generate the scripts, in a more technical aspect of the project.

DMAIC

The problem solving process that Six Sigma uses is called DMAIC, it is an acronym for the five phase processes of Define, Measure, Analyze, Improve and Control. DMAIC is the backbone of Six Sigma and is considered as such because it provides the fundamental structure and processes from which all Six Sigma activity is to emanate and

evolve. Like most Total Quality Management (TQM) processes or tools, Six Sigma is practiced typically in teams of diverse backgrounds and different organizational areas. In bringing a diverse team together to work a problem or initiative across a large enterprise, it is critical to have a common process or model that all members can follow. This process for Six Sigma is DMAIC. [5]

Define Phase

In the define phase, it is determined that a candidate project will be selected for Six Sigma. The intent of the define phase is to clarify the goals and value of a project. The selected issue will be a clear source of pain for the organization and either currently or potentially for the customer. This first step is in many ways the most difficult for a team. They must deal with many important questions. Although they may seem like rather basic questions, they are critical to the success of the team and project and must be answered in this early stage. Questions such as: “Who is the customer?”; “What are the customers’ requirements?” and “What are the benefits of making improvement?”

An important aspect of this phase is for the identified team to define the scope and goals of the project such that it will be clear to all when success is achieved. The tools used in this phase are not unique to Six Sigma and are typical of many TQM programs. It is fairly typical for teams to use fishbone diagrams in this phase. The define phase is considered a critical phase in that it is the first and the other four phases of DMAIC build upon it. [1]

Measure Phase

The define phase identifies the boundaries for the team to work within, aids in identifying the process variables, as well as guiding the team to develop a means to keep score. In the measure phase the team will assess the amount of variation within the variables. Teams are expected to identify and measure both input and output process variation.

Once the team has selected a measurement tool with high repeatability and reproducibility, they

devote their energy to measuring and charting the variables associated with the process. The typical type of tools used is again found in most TQM programs such as histograms, Pareto charts, and statistical process control charts. An important milestone in the measurement step is to get an initial sigma measurement of the output of the process under review. This preliminary measure of the process sigma is helpful in the team getting a better handle on the extent of their issue and potential impact on the customer. [5]

Analyze Phase

The goal of the team in this phase is to determine if the processes being studied are capable of obtaining their goals. Teams conduct capability studies to look for causes of why some processes are not obtaining the desired results. In this phase, computer software tools such as Microsoft Excel® often aid teams. Tools such as Excel® assist teams in reducing and organizing large quantities of data and thereby better understanding trends and relevant factors.

It is in this phase that teams really dig into the details and truly gain an understanding of the process in which they are studying. The end goal of the analyze phase is to identify the root cause of the project. In some projects, root cause is readily evident and teams move quickly through the analyze phase. In other projects, root cause is not immediately visible and the teams are required to do a significant amount of work. This work can last weeks and in some cases even months drawing on a number of tools and testing various ideas before getting to true root cause.

Good DMAIC problem solving generally includes close consideration of the types of causes. Six Sigma practitioners are known to look at common cause categories. These are methods, machines, materials, measures, Mother Nature, and people. The above mentioned categories are commonly referred to as the “4 M’s and 1 P”. [2]

Improve Phase

The improve phase is where many teams are tempted to go right at the very start of the DMAIC process. One of the key attributes and benefits of Six Sigma is the rigor that it employs in keeping teams from rushing immediately to the improve phase. In as much as teams, especially inexperienced ones, are initially frustrated that they cannot move immediately to the improve phase, they generally gain a deep appreciation and respect for Six Sigma once they arrive at this phase. This is due to the recognition of the methodical, and in general, high quality work that they have performed up to this point. Early on in the improve phase, teams many times go back and modify the scope of their project because they now have a better appreciation and understanding of the problem as well as the subject process. Regardless of a revisit of the project scope or not, the improve phase is where teams are expected to achieve results.

Achieving results, even after completing all of the rigorous work associated with DMAIC thus far, is often not an easy task. It is generally very difficult to identify creative solutions that truly address the root cause of the problem. Once several potential solutions have been identified, they must be tested. At this point, the team goes back into the analytical phase and collects and analyzes data. In many cases, the team may identify more than one solution to the problem. If this is the case, various proposed solutions are judged against criteria such as cost and timing of implementation as well as likely benefits. The final solution must be approved by the team champion and generally the technical leadership of the subject process. At this point in the DMAIC process, the “I” no longer stands for improve, it now stands for implement. [2]

Control Phase

The team’s objective in the control phase is to identify and implement a control plan that will successfully monitor the process and will readily indicate to the appropriate personnel when the process degrades or goes off track. An appropriate control plan will identify the process owner(s) and

will include a flow chart and standard operating procedure that contains the previously determined improvements.

An appropriate control plan will also include a response plan for dealing with problems that may arise. Additional responsibilities of the team in the control phase are to “sell” the project through presentation and demonstrations. Also, to hand off responsibility for the project to the people that use and own the subject process on a day-to-day basis. Finally, it is the team’s responsibility to ensure support from management for the long-term goals for the project. [2]

In a technical aspect of this Project the main tool used is Excel® and its ability to work with Visual Basic for Applications (VBA). Most of the automation process developed during the implementation part of the DMAIC is based on the flexible and easy use of VBA language. Following it’s described in a short manner the basic concept of VBA.

Visual Basic for Applications (VBA)

Visual Basic for Application is a programming environment designed to work with Microsoft’s Office® applications (Excel®, Word®, Access®, ect) components in each application (for example, worksheets or documents) are exposed as objects and made available to the programmer to use and manipulate to desired end. Anything you can do through normal use of the Office® applications can also be automated through programming.

You can also extend the abilities of the application through the use of additional reusable objects provided for the programmer. These reusable objects are referred to as Active X controls. These controls are pre built reusable programming components that you can add to your own programming projects. Common examples include text boxes, buttons, labels, and image controls. They are very useful to program developers because they are reusable and serve to handle common programming tasks. Because Active X controls are reusable the only have to be developed once, thus saving valuable time. VBA

includes several common ActiveX controls for use in Office Projects. [3]

Object-Based Approach Toward Programming

The object model forms the basis of the object-based approach. Object-based methods are based on the conceptual framework provided by the object model. The object model refers to the logical arrangement of objects in an object-based application. An object is a tangible entity that may exhibit some well-defined behavior. For example, a football is an object; it is a tangible, visible entity with a visible boundary. It is not necessary, however, for all objects to have a physical boundary. An organization does not have a physical boundary, but it has a conceptual boundary. Therefore, an organization is also an object. An object has a state, it may display a behavior, and it has a unique identity. The state of an object is indicated by a set of attributes and its values. For example, a bicycle can have states such as two wheels, number of gears, color, and so on. Behavior refers to the change of these attributes over a period of time. Consider the example of the bicycle again. The bicycle can exhibit behavior such as braking, accelerating, changing gears, and so on.

Every object has a unique identity, which distinguishes it from another object. For example, an animal is identified by its species. Two objects may have the same behavior, may or may not have the same state, and will never have the same identity. The identity of an object never changes during its lifetime. Following is a look at the benefits of the object-based approach toward programming. [4]

METHODOLOGY

This chapter will be describe the process to follow in order to accomplish the outcome stated on the objectives section of chapter one. The use of the tools mention on chapter two will be the backbone and guide to fallow to solve the problem the research is refer to. The process used on this research will emphasize on the last three steps or

phases of the DMAIC method, since the problem was already define and its metrics were previously done.

Analyze phase. This step of the DMAIC will organize all data gather for the Test Procedures using Excel; this datasheets will be used later to design the Macro who will create the test code script. With the data acquired and organized on the Excel spreadsheets it is simpler to do modification if it's required later on the Control phase.

Implementation phase. On this stage of project is expected to create the Macro using the VBA tools to generate a huge amount of test scripts that can be run on the simulator without any format error that can be crash or stop the simulator to run. First a script template have to be created, in order to generate this template a vast understanding of the objectives and goals of the Test Procedure is needed, this template should incorporate all the needs of the testing requirements into a document to fill with each variation of the test. The template should be only one document and have to have the ability to be flexible and at the same time standard.

After the generation of the template the VBA tool have to acquire the data from the spreadsheet generated during the analyze phase and then populate the template with each one of the test cases required by the Test Procedure.

Control Phase. During this phase it's expected to validate the VBA tool and all its support documents, by retrieving the results files obtained after run the test scripts on the simulated environment. On this step another VBA tool will be created to extract all the data results including failures and passes from the test scripts created by the generator tool. All this recuperated data will be use as a feedback to any change required to the generated code or to the data on the spreadsheets.

RESULTS AND DISCUSSIONS

During this chapter we are going to discuss the analysis done before and after the implementation of the automation tool, we are going to present the data acquire before and wish was the main

precursor of this project and then compare it with the results obtain after the implementation of the VBA tool.

From the metrics obtained from previous data gathering we have the current time to produce a working script to be between 8 hours per script depending on the difficulty level of the test procedure working on. We have been divided the tasks needed to complete a script into: receiving the TP, organize/segregate work, extract the data, and perform scripting. The Figure 1 will help to understand how the process was performed before the implementation of the VBA tool to generate the scripts.

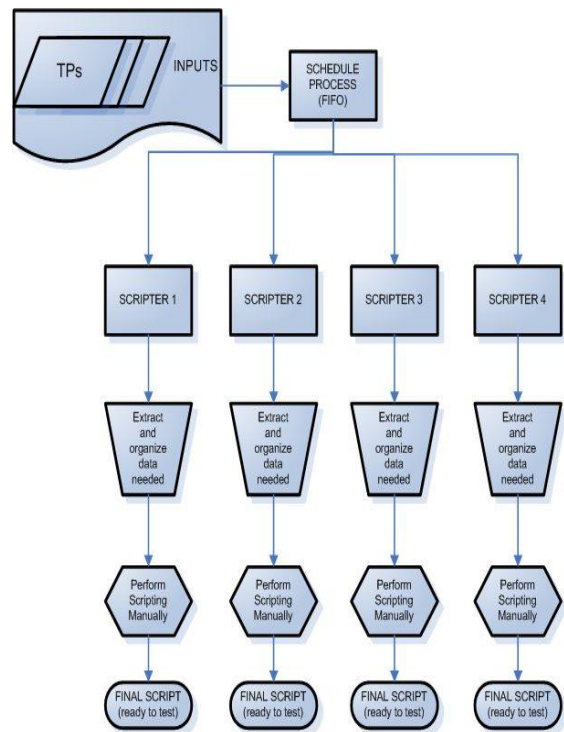


Figure 1
Process before implementations of VBA tool

From the Figure 1 we can see that the process follow a straight sequence having each one of the scripiter's perform almost the complete development process, except from organize - segregate task, which is perform by the team lead.

On Table 1 we can appreciate the times for each task of the process, having manually creation of the script the most time consuming. Also we can

found the amount of errors found per TP finished; this data will help us to measure the impact this project will have.

Table 1
Script Generation Historic Data

Task per TP	TP-1	TP-2	TP-3	Total Time	Average Time (hrs)
TOTAL Scripts	21	26	19		
Extract data	44	53	39	136	45.3
Organize data	33	39	30	102	34.0
Manual scripting	94	123	81	298	99.3
TOTAL per TP	171	215	150	536	178.7
Errors Found	7	8	6		
Time per script	8.143	8.27	7.89		8.1

With this process running the customer presents the time and budget constrain leading to the problems stated on this project. Been the most important constrain the time spend, during the initial phase those 3 baseline TP take about 5 weeks to complete 66 scripts. For the initial phase of the certification process of the System the customer requires creating about 1700 scripts divided into 38 TPs in less than 30 weeks. After gathering the team members and using the fishbone tool we acquire a list of possible causes leading to the problem. On Figure 2 we can see the fishbone analysis performed.

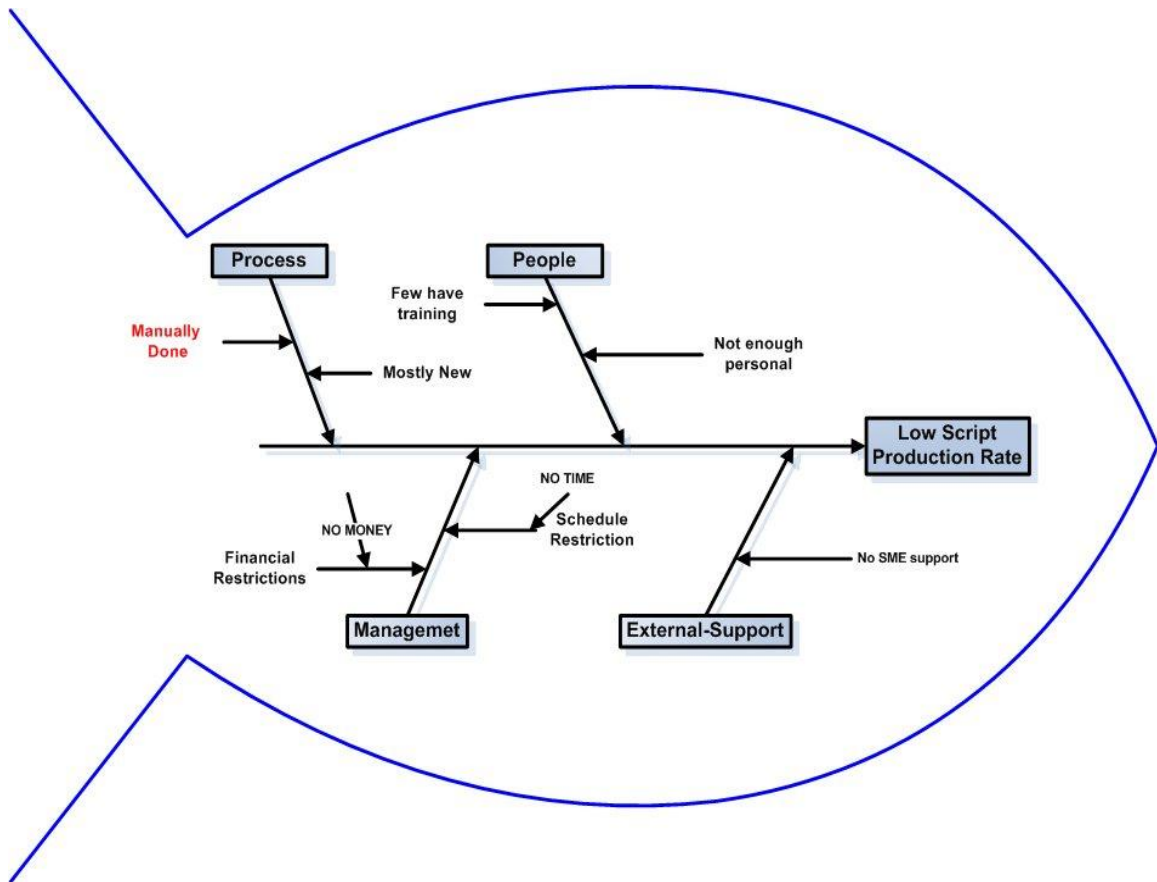


Figure 2
Fishbone Analysis

After this effort concludes we get to some possible solutions having the implemented by this project the more cost efficient one. One of the solutions propose to solve the problem were

increment the amount of scripiter's, which is a common solution found in any service company, "increment the personal will increment the production". This solution was evaluated and

having the restriction of the technical knowledge needed by the scripter in order to perform his task and the time and financial constraints presented by the customer, allocating more scripter's to perform the required tasks will alleviate the problem, but was not a feasible solution. Then we continue to analyze other options that were provided to solve the problem of low rate production.

Later another analysis made to the causes found on the fishbone analysis, the idea of convert the manually part of the process which is the task who have the most consuming time was selected and a new process flow down was design one of the solutions proposes the automation of the manually scripting task. This project will present the result of implementing this solution and the process changes involved. The main concept of this solution gives the possibility of increasing the production without incrementing the personal and complies with the same time frame requested by the customer.

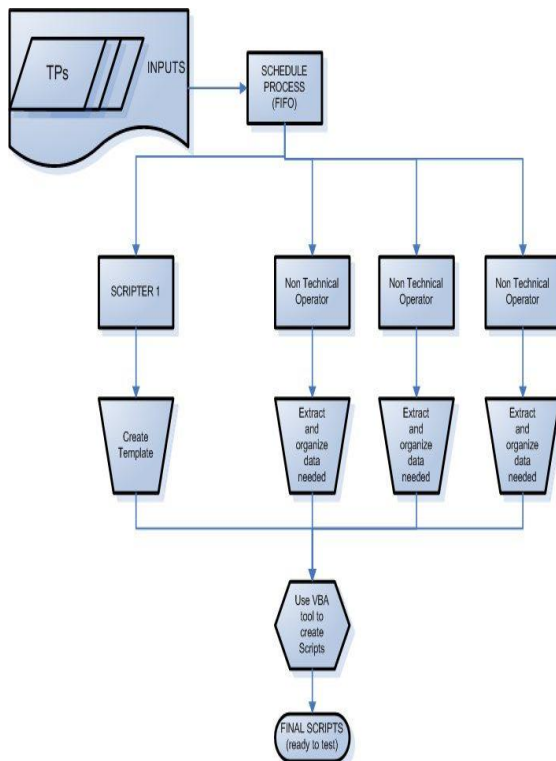


Figure 3
New Propose Process

In the Figure 3 we can see the propose solution, this solution incorporate the use of an automation tool to facilitate the process of the code generation. Selecting the VBA as the primary automation tool for this procedure was made mostly due to the compatibility with Excel which is where all the data acquire is stored during the extraction phase of the process. This part of the project was performed by the software developer team using the requirements we give them. We work hand to hand to help the team to design this VBA tool to perform with the requested requirements. After the completion of the design of the tool we continue with the implementation of the solution to the scripting process.

The new process flow down found on Figure 3 we can see that some changes were made to incorporate the use of the VBA tool, now we have the availability to use non technical personnel to perform an activity that was mostly clerical and released the scripter from this task and let them focus only on the creation of the template which required a vast amount of technical knowledge. This template was going to be used later in combination with the data extracted by the non technical personal on the VBA tool to create the scripts. With this increase in productivity for one TP by using the new approach, including the use of the new VBA tool the new process can now be expanded to use in a parallel way. In Figure 4 we can see how this new parallel process is implemented. Using this new approach we accelerate the process by having multiple TPs or Test Procedures assigned to scripter's in parallel, this generate the scripts needed to perform testing as fast as possible.

After the new process was completely incorporated the time needed to produce a script was reduce greatly, making possible the production of all the scripts required by the customer, between the time and budget constraints. Using the VBA tool to create the scripts was a significant productivity improvement to this new process.

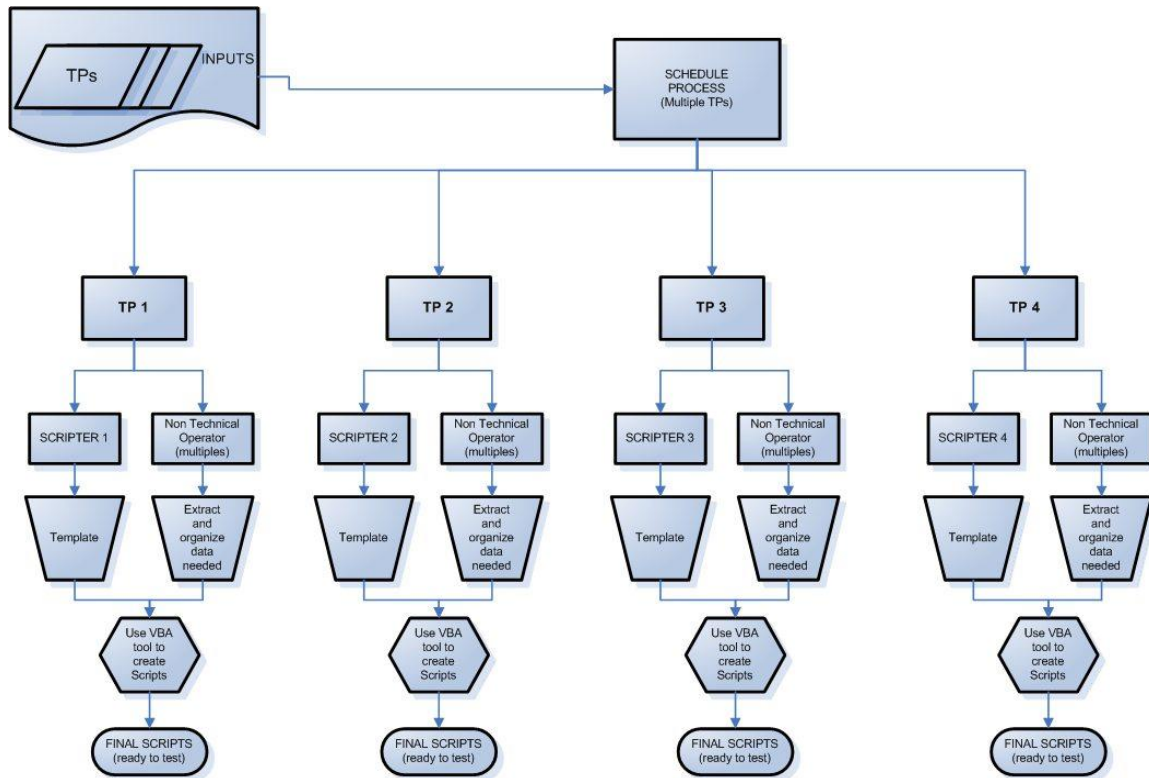


Figure 4
New Process Implemented

Table 2
New Process Implemented Data

Task per TP	time(hrs)	time(hrs)	time(hrs)	Total Time	Average Time
	TP-4	TP-5	TP-6		
TOTAL Scripts	35	62	41		
Extract data	51	82	53	186	62.0
Create Template	18	15	12	45	15.0
VBA(tool) scripting	8	9	8	25	8.3
TOTAL per TP	77	106	73	256	85.3
Erros Found	0	1	1		
Time per script	2.2	1.71	1.78		1.9

On the Table 2 we can see the amount of time saved by the implementation of the new approach and the use of the automation tool. Also we can say that this new process reduce vastly the amount of errors per TP found and therefore increase the quality of the scripts. With the automation tool in place we reduce the amount of errors since the only manual part of the process is now the creation of the template. In this part of the process the scripiter have more time to verify the compliance of the template assuring more quality at the end of the

process. The only errors found were cases were special modifications were needed in order to comply with the TP requirements.

By the implementation of this new process we are increasing the productivity by approximately 76%, which means that the 1700 scripts were completed in about 20 weeks leaving time to review the output results of the scripts after ran on the simulator. With this review we can validate that the automation tool and that the new process implemented is working without any special issue.

On this phase we create a standard document (SOP) which described in detail the new process; this document will help in the future if similar work is received.

CONCLUSION

After the implementation of this project we can conclude that the use of automation tools to create software code is a feasible solution and that the process followed to do that code need to be revised

and modified to adhere to the new approach of automation.

For this project we can say that all the objectives stated on chapter one were surpassed and that the results obtained were evidently better than expected. After this project was implemented the customer decided to send all the Test Procedures required on the certification process to our team, this work load lead to a contract to create more than 8000 scripts.

REFERENCES

- [1] Munro, R, *Six sigma for the shop floor: A pocket guide*. Vol. Milwaukee: Quality Press 2002.
- [2] Pande, P., & Holpp, L, *What is six sigma?*, New York: McGraw-Hill 2002.
- [3] Duane, B, "Microsoft Excel VBA Programming for the Absolute Beginner", *For the Absolute Beginner (Series)*, Indianapolis, Indiana: Course PTR. 2002.
- [4] C, Rachna.; G, Taruna, *VBA Professional Projects* Rocklin, Calif. Course PTR. 2002.
- [5] C Hutton, T., *ACE vs Six Sigma* Boston, Ma. MIT. 2004.