

On Demand Feedback Analysis for Certification Process

A M Chandrashekhar

Assistant Professor,

Dept of Computer Science and Engineering,
Sri Jayachamarajendra College of Engineering(SJCE),
JSS S & T University Campus,
Mysuru, Karnataka

Muktha G S

Final year, M.Tech in Comp Engg,
Dept of Computer Science and Engineering,
Sri Jayachamarajendra College of Engineering(SJCE),
JSS S & T University Campus,
Mysuru, Karnataka

Abstract— Many companies provide certifications to other vendors to validate that the vendor's product functions efficiently on their technology. But it so happens that the process of certification might take longer due to errors that pop up which the vendors will not be able to resolve. The vendors need to resolve the errors with the help of product owner's engineers which is a lengthy and less efficient process. With this in regard, the solution of on-demand feedback analysis of the certification process proposed in this paper helps minimize the delay in the certification process.

Keywords- Workbench; plug-in; Error-recorder; Solution-database;

I. INTRODUCTION

Feedback is said to be part of a cause-and-effect loop where information about a system is returned to the controller of the system to improve its performance. A process that uses the conditions of one component to regulate the function of the other is called as a feedback mechanism. It is done to either increase or dampen the change in the system. Certification process is a procedure wherein a partner company runs certain functional tests provided by the product owner company on the component that they have produced to validate that, that specific component functions correctly in the environment of the product company produced technology. Incorporating the feedback mechanism in the certification process speeds up the process as the product engineers will be able to give solutions to multiple partners at the earliest instance.

Company XYZ owns and provides a technology, say, "TechA". Vendor V1 has a product P1, vendor v2 has a product P2 and so on. If P1, P2...Pn products work well for the said technology, then it is certified for that technology, TechA, by validating the product through functional tests. The products need to pass tests T1...Tn to qualify as a certified product for the specified technology^[1].

Figure 1 represents the design structure where multiple partners wanting to certify their component for a specific technology are running various functional tests on the workbench. The set of tests for certification are bundled up and installed on the workbench as a plug-in.

II. LITERATURE SURVEY

A. Workbench

Workbench is an eclipse based tool which acts as a base upon which multiple plug-ins are installed to run certification tests. Workbench Dashboard^[2] aids in the discovery and installation of plug-ins and keeps you up to date with the latest updates. A wide variety of available SDK plug-ins support specific products and technologies, providing IDE features such as syntax highlighting, code completion and one-click debugging. The workbench Provides remote consoles, object browsers, and other services.

B. Plug-ins

Applications or programs that can be installed and used easily as part of the underlying workbench are called as plug-ins. In computing, a plug-in is a software component that adds a specific feature to an existing computer program. They are also called as add-in, add-on or extensions^{[3][4]}. Few of the main reasons for applications to support plug-ins include:

- Plug-ins are used to enable third party developers to create ability to extend the application.
- They also support ability to easily add new features
- To reduce the size of an application

The host application provides services which the plug-in can use, including a way for plug-ins to register themselves with the host application and a protocol for the exchange of data with plug-ins. Plug-ins depend on the

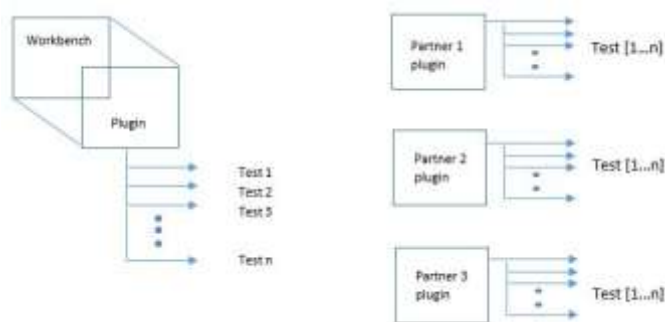


Figure 1: Existing Design Structure

services provided by the host application and do not usually work by themselves. Conversely, the host application operates independently of the plug-ins, making it possible for end-users to add and update plug-ins dynamically without needing to make changes to the host application. Programmers typically implement plug-in functionality using shared libraries installed in a place prescribed by the host application.

C. Certification Process

Step 1 - Inception: The vendor negotiates a partnership agreement with product parent company through some partner alliance program. The partner will be provided access to the relevant materials, certification kits and development kits via the company's site.

Step 2 - Development: The partner has to either develop or update the component to be certified as necessary: Time for this phase varies depending on the complexity of the component and availability of all resources.

Step 3 - Component Certification Tests: The partner downloads the relevant Certification Kits and Tools^[5]. The partner runs the Functional Verification Tests for driver/device certification. If the company makes any critical changes or additions to the tests during the partner's certification test period, the company will request the partner to run the new tests.

Step 4 - Certification Test Verification: Once all functional verification tests pass, the partner submits the results and files a request, via the workbench, to conclude the certification and enlist the partner's component in the certified list.

Step 5 – Release: The information provided with the certification submission is used to update the certified components list. This completes the certification process and allows the partner to state their component as that specific technology certified.

III. PROBLEM DEFINITION

It takes a long time for a partner to run the tests and return the error reports when a failure is encountered. Once the error report is received, the engineer needs to go through the logs and find the issue and figure out a solution for each failed test. This is a cumbersome process as a lot partner's time is wasted on waiting for the test engineer to provide a solution.

IV. PROPOSED DESIGN

The objective is to develop an automated knowledge base feedback system as part of an effort to facilitate a dynamic repository of errors and solutions. The system collects error

data from the partner's plug-in. This data is dynamically updated at the product owner's end. Here the engineers analyze the data and formulate a solution and update the solution to the corresponding error in every partner's solution database present within their workbench. The solution report would usually include a set of suggested modifications to the underlying setup to make the process run in an error free manner.^{[6][7]}

Incorporate a piece of code or software that records the on goings of the certification suite. The software would count the number of times each test is run. The higher the count, the more the chance that the test isn't working correctly. Also, the error with which the test is failing and the warnings thrown out are recorded and stored.

A. Error Recorder

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

B. Solution Database

Maintaining a database of solved issues also gives an added advantage. With this database, the partner can try and resolve the issue on own rather than contacting the certification team. The database will be updated with solutions to every occurring new issue.

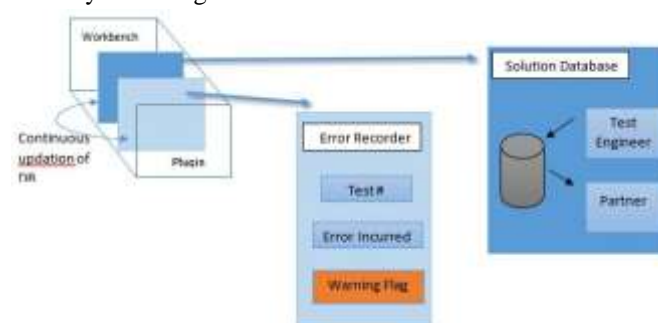


Figure 2: Proposed Design Structure

The proposed design structure includes an additional two add-ons, namely, error recorder and solution database as shown above in figure 2. The first interface is placed beneath the test suite plug-in to record all the test data. The error recorder captures the number of times each test is run, test result – pass/fail, if the test failed then the error occurred is also captured. All this data from each partner run is instantaneously updated in the product database. Each time a test is run the count goes up and each time a test fails with the

same error the count goes up. This indicates to the test engineer of a predominant issue to which he/she has to give high priority and solve quicker. The second interface placed beneath the error recorder layer is the solution database. It is a repository to store all the workarounds the test engineer had found to an arisen error. With the presence of this repository and its continuous updating, the partner need not raise an error request to the product owner and get the solution; they can find it on-hand in the solution database. Thus, this increases the efficiency of the process.

This paper is proposed with respect to the functionality of hardware drivers on a virtualization environment.

C. Tools Device Drivers

Device drivers smooth mouse operations, make VMware features such as folder sharing available, and improve sound, graphics, and networking performance. If you do a custom VMware Tools installation or reinstallation, you can choose which drivers to install. Which drivers are installed when you install VMware Tools also depends on the guest operating system and the VMware product^{[8][9]}. The following device drivers can be included with VMware Tools:

SVGA driver This virtual driver enables 32-bit displays, high display resolution, significantly faster graphics performance. When you install VMware Tools, a virtual SVGA driver replaces the default VGA driver, which allows for only 640 X 480 resolution and 16-color graphics. On a Windows guest operating system whose operating system is Windows Vista or later, the VMware SVGA 3D (Microsoft - WDDM) driver is installed. This driver provides the same base functionality as the SVGA driver, and it adds Windows Aero support.

SCSI driver Included with VMware Tools is the VMware Paravirtual SCSI driver for use with paravirtual SCSI devices. Drivers for other storage adapters are either bundled with the operating system, or they are available from third-party vendors. For example, Windows Server 2008 defaults to LSI Logic SAS^[10], which provides the best performance for that operating system. In this case, the LSI Logic SAS driver provided by the operating system is used.

Paravirtual SCSI driver This driver is for VMware Paravirtual SCSI adapters, which enhance the performance of some virtualized applications.

VMXNet NIC drivers The vmxnet and vmxnet3 networking drivers improve network performance. Which driver is used depends on how you configure device settings for the virtual machine. Search the VMware Knowledge Base for information on which guest operating systems support these drivers. When you install VMware Tools, a VMXNet NIC driver replaces the default vlane driver.

Mouse driver The virtual mouse driver improves mouse performance. This driver is required if you use some third-party tools such as Microsoft Terminal Services.

Audio driver This sound driver is required for all 64-bit Windows guest operating systems and 32-bit Windows Server 2003, Windows Server 2008, and Windows Vista guest operating systems if you use the virtual machine with VMware Server, Workstation, or Fusion.^[11]

D. Virtualization

The VMware virtualization platform is built on a business-ready architecture. VMware virtualization works by inserting a thin layer of software directly on the computer hardware or on a host operating system. This contains a virtual machine monitor or “hypervisor” that allocates hardware resources dynamically and transparently.

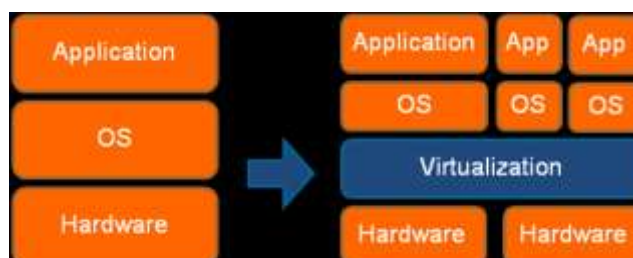


Figure 3: Virtual environment in a system

Virtualization refers to the creation of a virtual resource such as a server, desktop, operating system, file, storage, or network. The main goal of virtualization is to manage workloads by radically transforming traditional computing to make it more scalable^[12]. Virtualization has been a part of the IT landscape for decades now, and today it can be applied to a wide range of system layers, including operating system-level virtualization, hardware level virtualization and server virtualization.

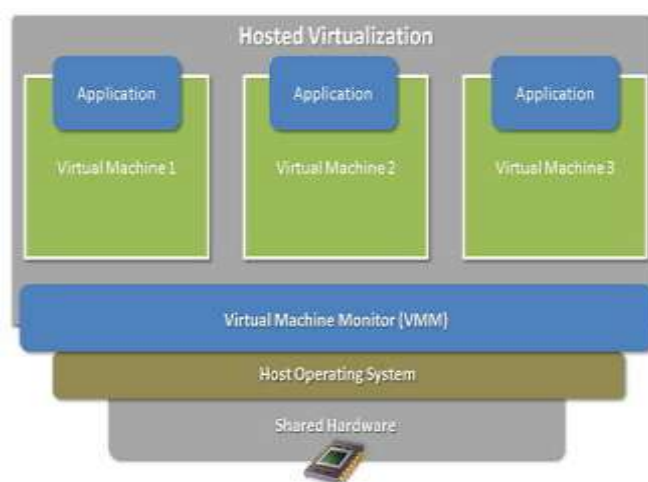


Figure 4: Virtual Machine Structure

A virtual machine is nothing but a data file on a physical computer that can be moved and copied to another computer, just like a normal data file^[13]. The computers in the virtual environment use two types of file structures: one defining the hardware and the other defining the hard drive. The virtualization software, or the hypervisor, offers caching technology that can be used to cache changes to the virtual hardware or the virtual hard disk for writing at a later time. Virtualization can be categorized into different layers: desktop, server, file, storage and network^[14]. Each layer of virtualization has its own set of advantages and complexities.

Virtual machines are completely independent from their underlying physical hardware. For example, we can configure a virtual machine with virtual components (eg, CPU, network card, SCSI controller) that are completely different the physical components that are present on the underlying hardware^{[15][16]}. Virtual machines on the same physical server can even run different kinds of operating systems (Windows, Linux, etc).

When coupled with the properties of encapsulation and compatibility, hardware independence gives us the freedom to move a virtual machine from one type of x86^[17] computers to another without making any changes to the device drivers, operating system, or applications. Hardware independence also means that we can run a heterogeneous mixture of operating systems and applications on a single physical computer^[18].

A virtual machine is a tightly isolated software container that can run its own operating systems and applications as if it were a physical computer. A virtual machine behaves exactly like a physical computer and contains its own virtual (i.e. software-based) CPU, RAM, hard disk and network interface card (NIC)^{[19][20]}. An operating system can't tell the difference between a virtual machine and a physical machine, nor can applications or other computers on a network. Even the virtual machine thinks it is a "real" computer. Nevertheless, a virtual machine is composed entirely of software and contains no hardware components whatsoever. As a result, virtual machines offer a number of distinct advantages over physical hardware.

E. Open Virtualization Format

The plugin is an OVF or OVA package bundle. An OVF refers to the Open Virtualization Format, which is a "packaging standard designed to address the portability and deployment of virtualization appliances" (Source). The OVF format standard was formed by the Distributed Management Task Force, or DMTF, which is an industry working group comprised of over 160-member companies and organizations.

The DMTF board is comprised of 15 technology companies and includes Dell, EMC, VMware, Oracle, &

Microsoft. As announced at VMworld 2010, DMTF's OVF standard was adopted as a National Standard by ANSI.

An OVF package structure consists of a number of files: a descriptor file, optional manifest and certificate files, optional disk images, and optional resource files (such as ISO's). The optional disk image files can be VMware vmk's, or any other supported disk image file. **OVA and OVF: The Differences**

OVF is not only the name of the packaging format standard, but it also refers to the package when distributed as a group of files. An OVA (open virtual appliance or application) is merely a single file distribution of the same file package, stored in the TAR format. The DMTF's Open Virtualization Format (OVF) standard provides the industry with a standard packaging format for software solutions based on virtual systems, solving critical business needs for software vendors and cloud computing service providers. OVF has been adopted and published by the International Organization for Standardization (ISO) as ISO 17203. The rapid adoption of virtual infrastructure has highlighted the need for a standard, portable metadata format for the distribution of virtual systems onto and between virtualization platforms.

In the modern cloud computing era, OVF provides a platform independent, efficient, open and extensible packaging and distribution format that facilitates the mobility of virtual machines and gives customers platform independence.

An OVF package can be used: By an independent software vendor (ISV) to publish a software solution, by a data center operator to transport a software solution from one data center to another and by a customer to archive a software solution. In any other use case that can be met by having a standardized package for a software solution. From the user's point of view, OVF is a packaging format for virtual appliances. Once installed, an OVF package adds to the user's infrastructure a self-contained, self-consistent, software application that provides a particular service or services.

V. RESULTS

| Test Name | Test ID | No of times run | Result | Error(if any) |
|-----------|---------|-----------------|--------|--|
| Test1 | 101 | 1 | Pass | - |
| Test2 | 102 | 5 | Fail | STAF issue, Uninitialized pointer |
| Test5 | 105 | 3 | Fail | Base OS error, failed to get management IP |

Table 1: Test data

Table 1 is a record of the test case and their corresponding runs and results. This table in turn creates

Table 2 which is a error data table sent to engineer for analysis. This is the feedback analysis mechanism. The test engineer returns the solution and updates it in the same table for every partner irrespective of the errors they have incurred for future access.

| Error | Error ID | Warning Flag | Solution |
|-----------------------------|----------|--------------|---|
| STAF issue | E1 | | Install STAF tools |
| Uninitialized pointer | E2 | !! | Separate input values for different subsets |
| Base OS error | E3 | !! | |
| Failed to get management IP | E4 | | Restart network |

Table 2: Solution table

Solution table is constantly updated with solutions by test engineer. The warning flag is raised when many partners are facing the same issue multiple times. The flag is an alert message to the solver to indicate the severity of the condition. Solutions are added to the repository on the go, thus speeding up the entire process.

The chart below depicts the number of times a particular error has occurred. So the error with the utmost score gets the first preference for evaluation.



VI. CONCLUSION

When a partner need to certify their component against a technology, the certification process usually takes a lot of time due to manual exchange of data between the partner and the engineer. Hence to reduce this delay, we proposed a feedback analysis solution to incorporate two new interfaces within the workbench for continuous error recording and solution updating. The mean time to finish a certification process reduces greatly. This leaves time on the partner's side to take up more certifications or add more functionality to their components and release it.

Future enhancement can involve a carry forward of the similar mechanism to the partner's next certifications. In the corporate world, when new features come in, the partners don't

redo the component altogether; they just add the new feature to it and register under partial certification for the added feature. In this regard, building an interface that can use existing repository and adding new error and solution data to it can be considered for future enhancement.

REFERENCES

- [1] "An Introduction to the Programmer's Workbench" by T. A. Dolotta and J. R. Mashey, Bell Laboratories Piscataway, New Jersey
- [2] I/O Vendor Partner Program Guide v3.11 by VMware Incorporated
- [3] "Developing a methodology for online feedback and assessment" - Loughborough University Institutional Repository
- [4] A Summary of Research on "The Effectiveness of K-12 Online Learning" by Susan Patrick and Allison Powell, iNACOL
- [5] <https://www.computerhope.com/jargon/p/plugin.htm>
- [6] A. M. Chandrashekhar and K. Raghuvver, "Confederation of FCM Clustering, ANN and SVM Techniques of Data mining to Implement Hybrid NIDS Using Corrected KDD Cup Dataset", Communication and Signal Processing (ICCSP) IEEE International Conference, 2014, Page 672-676.
- [7] A. M. Chandrashekhar and K. Raghuvver, "Improvising Intrusion detection precision of ANN based NIDS by incorporating various data Normalization Technique – A Performance Appraisal", IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 2, Issue 2, Apr-May, 2014.
- [8] <https://code.vmware.com/kits-lifecycle-policy>
- [9] A. M Chandrashekhar and K. Raghuvver, "Diverse and Conglomerate modi-operandi for Anomaly Intrusion Detection Systems", International Journal of Computer Application (IJCA) Special Issue on "Network Security and Cryptography (NSC)", 2011
- [10] A. M Chandrashekhar and K. Raghuvver, "Hard Clustering Vs. Soft Clustering: A Close Contest for Attaining Supremacy in Hybrid NIDS Development", Proceedings of International Conference on Communication and Computing (ICCC - 2014), Elsevier science and Technology Publications.
- [11] A. M. Chandrashekhar and K. Raghuvver, "Amalgamation of K-means clustering algorithm with standard MLP and SVM based neural networks to implement network intrusion detection system", Advanced Computing, Networking, and Informatics –Volume 2(June 2014), Volume 28 of the series Smart Innovation, Systems and Technologies pp 273-283.
- [12] A. M. Chandrashekhar and K. Raghuvver, "Fusion of Multiple Data Mining Techniques for Effective Network Intrusion Detection – A Contemporary Approach", Proceedings of Fifth International Conference on Security of Information and Networks (SIN 2012), 2012, Page 178-182.
- [13] A. M. Chandrashekhar, Jagadish Revapgol, Vinayaka Pattanashetti, "Big Data Security Issues in Networking", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Volume 2, Issue 1, JAN-2016.
- [14] "Efficient Query Integrity for Outsourced Dynamic Databases" by Qingji Zheng, Shouhuai Xu and Giuseppe Ateniese, Department of Computer Science, University Of Texas at San Antonio [15] "A Study of Dynamic Database in Mobile GIS" by Ka-wai Kwan and Zhong Shi, Advanced Research Centre for Spatial Information Technology, Hong Kong
- [16] Puneeth L Sankadal, A. M Chandrashekhar, Prashanth Chillabatte, "Network Security situation awareness system" International Journal of Advanced Research in Information and Communication Engineering(IJARICE), Volume 3, Issue 5, May 2015.
- [17] P. Koushik, A.M.Chandrashekhar, Jagadeesh Takkalakaki, "Information security threats, awareness and cognizance" International Journal for Technical research in Engineering(IJTRE), Volume 2, Issue 9, May 2015.

-
- [18] A.M.Chandrashekhar, Yadunandan Huded, H S Sachin Kumar, “Advances in Information security risk practices” International Journal of Advanced Research in data mining and Cloud computing (IJARDC), Volume 3, Issue 5, May 2015.
- [19] A. M. Chandrashekhar, Muktha G S, Anjana D K, “Cyberstalking and Cyberbullying: Effects and prevention measures”, Imperial Journal of Interdisciplinary Research (IJIR), Volume 2, Issue 2, JAN-2016.
- [20] A.M.Chandrashekhar, Huda Mirza Saifuddin, Spoorthi B.S, “Exploration of the ingredients of original security” International Journal of Advanced Research in Computer Science and Applications(IJARCSA), Volume 3, Issue 5, May 2015.
- [21] A.M.Chandrashekhar, Rahil kumar Gupta, Shivaraj H. P, “Role of information security awareness in success of an organization” International Journal of Research(IJR), Volume 2, Issue 6, May 2015.
- [22] A.M.Chandrasekhar, Nagaveni Bhavi, Pushpanjali M K, “Hierarchical Group Communication Security”, International journal of Advanced research in Computer science and Applications (IJARCSA), Volume 4, Issue 1, Feb-2016.