NVEO
Natural Volatiles &
Essential Oils

Web Based Cloud Server Management System

Dr. Arif Abdul Rahuman[1] , Ms. K. Elaiyarani[2] , Sagar Kumar Pandey[3]

[1]Asso. Professor, Dhanalakshmi Srinivasan College of Engineering and Technology.

[2]Assistant Professor, Dhanalakshmi Srinivasan College of Engineering and Technology.

[3]Student, Dhanalakshmi Srinivasan College of Engineering and Technology.

ABSTRACT Virtualization has attained mainstream status in enterprise IT industry. Despite its widespread adoption, it is known that virtualization also introduces non-trivial overhead when tasks are executed on a virtual machine (VM). In particular, a combined effect from device virtualization overhead and CPU scheduling latency can cause performance degradation when computation intensive tasks and I/O intensive tasks are co-located on a VM. Such an interference also causes extra energy consumption. In this paper, we present Hylics , a novel solution that enables efficient data traverse paths for both I/O and computation intensive workloads. This is achieved with the provision of in-memory file system and network service at the hypervisor level. Several important design issues are pinpointed and addressed during our prototype implementation, including efficient intermediate data sharing, network service offloading, and QoS-aware memory usage management. Based on our real-world deployment on KVM, we show that Hylics can significantly improve computation and I/O performance for hybrid workloads. Moreover, this design also alleviates the existing virtualization overhead and naturally optimizes the overall energy efficiency.

INTRODUCTION

Architecturally, JSP may be viewed as a high-level abstraction of Java servlets. JSP pages are loaded in the server and are operated from a structuredspecial installed Java server packet called a Java EE Web Application, often packaged as a .war or .ear file archive.

JSP allows Java code and certain pre-defined actions to be interleaved with static web markup content, with the resulting page being compiled and executed on the server to deliver an HTML or XML document. The compiledpages and any dependent Java libraries use Java byte code rather than a native software format, and must therefore be executed within a Java virtual machine(JVM) that integrates with the host operating system to provide an abstract platform-neutral environment.

JSP syntax is a fluid mix of two basic content forms: script let elements and markup. Markup is typically standard HTML or XML, while script let elements are delimited blocks of Java code which may be intermixed with the markup. When the page is requested the Java code is executed and its output is added, in situ, with the surrounding markup to create the final page. JSP pages must be compiled to Java byte code classes before they can be executed, but such compilation is needed only when a change to the source JSP file has occurred.
that code, so markup inside an if block will only appear in the output when the if condition evaluates to true; likewise markup inside a loop construct may appear multiple times in the output depending upon how many times the loopbody runs.

2.**MATERILAS AND METHODS**

The JSP syntax adds additional XML-like tags, called JSP actions, to invoke built-in functionality. Additionally, the technology allows for the creation of JSP tag libraries that act as extensions to the standard HTML or XML tags. JVM operated tag libraries provide a platform independent way of extending the capabilities of a web server. Note that not all commercial Java servers are Java EE specification compliant. My SQL is a relational database management system (RDBMS)[2] that runs as a server providing multi-user access to a number of databases. The SQL phrase stands for Structured Query Language.
It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. Months or years of effort, thousands and millions of dollars, and untold professional embarrassment can be averted if an ill-conceived system is recognized early in the definition phase. Feasibility and risk analysis are related in many ways. If project risk is great, the feasibility of producing quality software is reduced. During product engineering, however, we concentrate our attention on four primary areas of interest.

Technical feasibility is the need of hardware and software, which are needed to implement the proposed system in the organiztion. Technical requirements is to be fulfilled to make the proposed system work. This should be necessarily predetermined so as to make the system more competent. The Economical feasibility must satisfy the needs of the technical feasibility and the operational feasibility. It involves the economic feasibility of developing and implementing the proposed system

The proposed system should used the internet level then the different types of end users are involved in the system, so it solves the user's needs and the organization needs. And it supports the all users environment

HTML came to existence in 1970. The development of HTML was initiated by Tim Burners Lee was later taken over by Dan Cannolly, Dave Ragett. The ancestor of HTML is SGML(Standard Generalized Markup Language). SGML was developed by ISO(International Standard Organization) and was first used by the US DOD(Department Of Defense). SGML and HTML have a parent-child relationship. HTML can be referred to as subset of SGML HTTP protocol used to transfer HTML

Java script is a compact; object based scripting language for developing client and server Internet applications.
Netscape Navigator 2.0 interprets JavaScript statements embedded directly in an HTML page, and live wire enable you to create server-based applications similar to common gateway interface (CGI) programs.

The best way to understand Java Server Pages work is by contrasting a Web server that supports Java Server Pages with Web server that doesn't. JSP is easy to learn, robust, scalable, and cross-platform.

Prior to the introduction of Java server Pages, the main function of Tomcat Server was to serve static HTML pages. When someone requested a web page from a web site using Tomcat server, the server would fetch a static HTML fileform disk or memory and send it out to the person's browsers. The primary responsibility of Tomcat Server was to act as an efficient interface between browser and a bunch of files sitting on the Web server's hard drive.

Java Server Pages (JSPs) are Web pages that contain server-side scripts in addition to the usual mixture of text and HTML (Hypertext Markup Language) tags. Server-side scripts are special commands you put in Web pagesthat are processed before the pages are sent from your Personal Web Server to the Web browser of someone who's visiting your Web site. . When you type a URL in the Address box or click a link on a Web page, you're asking a Web server on a computer somewhere to send a file to the Web browser (sometimes called a "client") on your computer. If that file is a normal HTML file, it looks exactly the same when your Web browser receives it as it did before the Web server sent it. After receiving the file, your Web browser displays its contents asa combination of text, images, and sounds.

In the case of an Java Server Page, the process is similar, except there's anextra processing step that takes place just before the Web server sends the file. Before the Web server sends the Java Server Page to the Web browser, it runs all server-side scripts contained in the page. Some of these scripts display the current date, time, and other information. Others process information the user has just typed into a form, such as a page in the Web site's guest book.

The appearance of an Java Server Page depends on who or what is viewingit. To the Web browser that receives it, an Java Server Page looks just like a normal HTML page. If a visitor to your Web site views the source code of an Java Server Page, that's what they see: a normal HTML page. However, the filelocated in the server looks very different. In addition to text and HTML tags, you also see server-side scripts. This is what the Java Server Page looks like to the Web server before it is processed and sent in response to a request. Forms are a convenient way to communicate with visitors to your Web site. Using forms, you can create a survey form and ask visitors to fill it out. When they fill out the form, you can process the results automatically. With forms, there are two steps: first you create the form, and then you process it. To create a form for an Java Server Page, just create a standard HTML form.Java Server Pages provide a mechanism for processing forms that, unlike CGI scripting, doesn't involve serious programming: the Request Form. Considering the form above, we may create the file bellow and get a response.

It is most often used to access environment information. One of the more common pieces of information accessed by the application object is objects thatare stored in the Servlet Context. These objects are stored there so that they will be available the whole time the servlet engine is running.

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format. Database systems are more powerful than data files in that data is morehighly organized. In a well-designed database, there are no duplicate pieces of data are grouped together in a single structure or record, and relationships can be defined between these structures, and records.

When working with data files, an application must be coded to work with the specific structure of each data file. In contrast, a database contains a catalog that applications use to determine how data is organized. Generic database applications can use the catalog to present users with data fromdifferent databases dynamically, without being tied to a specific data format.

3.RESULTS AND DISCUSSIONS

Based on the user requirements and the detailed analysis of a new system, the new system must be designed. This is the phase of system designing. It is a most crucial phase in the development of a system.

In the preliminary or general design, the features of the new system are specified. The costs of implementing these features and the benefits to be derivedare estimated. If the project is still considered to be feasible, we move to the detailed design stage. In the detailed design stage, computer oriented work begins in earnest. At this stage, the design of the system becomes more structure design is a blue print of a computer system solution to a given problem having the same components and interrelationship among the same components as the original problem. Input, output and processing specification are drawn up in detail. In thedesign stage, the programming language and the platform in which the new system will run are also decided. Database Design is a crucial factor in the performance of a system both in terms of system timings and in the case with which the system can be maintained or modifies.

Testing is a process of executing a program with the intent of finding an error. A good testing case is one that has a high probability of finding an as- yet undiscovered error. A successful test is one that uncovers an as-yet undiscovered error. If testing is conducted successfully, it will uncover errors in the software. As a secondary benefit, testing demonstrates that software functions appear to be working according to specification and that performance requirement appear to have been met.

All tests should be traceable to customer requirements. The objectiveof software testing is to uncover errors. Tests should be planned long before testing begins. Test planning can design as soon as the requirements model is complete. All tests should be planned and designed before any code has been generated.

The parento principle applies to software testing. The parento principle implies that 80 percent of all errors uncovered during testing will likely be traceable to 20 percent of all program modules. The testing should begin “in the small” and progress toward testing “inthe large”. To be more effective testing should be conducted by an independentthird party. By “most effective”, testing means that has the highest probability of finding errors.

A successful test case is one that uncovers an as-yet-undiscovered error.Software testability is how a computer program can be tested. The following listof categories shows that this software has been tested. Software testing is often referred as verification and validation. Verification refers to the set of activities that ensures that the software correctlyimplements the specific function. Validation refers to a different set of activitiesthat ensures that the software that has been built is traceable to the customer requirements.

Unit testing comprises the set of test performed by an individual programmer prior to integration of the unit into a larger system. There are different types of tests to be performed on a programming unit.

Functional Test cases involves exercising the code with normal input values for which the expected results are known, as well as boundary values andspecial values.

Stress Tests are those tests designed to intentionally break the limit. A great deal can be learnt about the strengths and limitations of a program by examining the manner in which a program unit breaks.

Structure Tests are concerned with exercising the internal logic of a program and traversing particular execution paths to exercise, deriving test date,determining the criterion to be used, executing test cases.

It addresses the issues associated with the dual problems of verification and program construction. Black box test case design techniques are the most prevalent during integration, although a limited amount of white box testing maybe used to ensure converge of major control paths. Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objectiveis to take unit-tested modules and build a program structure that has been depictedby design.

This test was performed with the users of the system and made sure thatit performed as they expected. It was verified that all functionality required by the user have been satisfied. So for, the system has been found defect free and isworking well.

After having the user acceptance of the proposed system developed, theimplementation phase begins. Implementation is the stage of a project during which theory is turned into practice. During this phase, all the programs of the system are loaded onto the user’s computer. After loading the system, training ofthe users starts.

Maintenance phase is the final stage of the system life cycle. It is necessaryto eliminate errors in the system during its working life and to tune the system toany variations in its working environment. It has been seen that there are alwayssome errors found in the system that must be noted and corrected. It also means the review of the system from time to time.

4. CONCLUSION

A computerized alumni management system has been developed and the system was tested with sample data. The system results in regular timely preparations of required outputs. In comparison with manual system the benefitsunder a computer system are considerable in the saving of man power working hours. It also increases the feasibility Provision for addition and deletion of student /faculty is there in the system. It is possible to view information of otherstudent/faculty after successfully logging in. The entire project runs on windowsenvironments. The system can be used to increase student faculty interaction and also increase their involvement with the college.

It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. Months or years of effort, thousands and millions of dollars, and untold professional embarrassment can be averted if an ill-conceivedsystem is recognized early in the definition phase.

REFERENCE

1. J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "Passwordsand the evolution of imperfect authentication," Communications of the ACM, vol. 58, no. 7, pp. 78–87, Jun. 2015.

2. M. A. S. Gokhale and V. S. Waghmare, "The shoulder surfing resistant graphical password authentication technique," Procedia Computer Science, vol. 79, pp. 490–498, 2016.

3. J. Ma, W. Yang, M. Luo, and N. Li, "A study of probabilistic password models," in Proceedings of 2014 IEEE Symposium on Security and Privacy, May 2014, pp. 689–704.