BIOINFORMATION
Discovery at the interface of physical and biological sciences

open access www.bioinformation.net Views & Challenges
Volume 7(4)

An efficient approach to the deployment of complex open source information systems

Truong Van Chi Cong* & Eildert Groeneveld

Institute of Farm Animal Genetics (FLI), Höltystrasse 10, 31535 Neustadt/Mariensee, Germany; Truong Van Chi Cong - Email: cong.chi@fli.bund.de; *Corresponding author

Received September 27, 2011; Accepted October 02, 2011; Published October 14, 2011

Abstract:
Complex open source information systems are usually implemented as component-based software to inherit the available functionality of existing software packages developed by third parties. Consequently, the deployment of these systems not only requires the installation of operating system, application framework and the configuration of services but also needs to resolve the dependencies among components. The problem becomes more challenging when the application must be installed and used on different platforms such as Linux and Windows. To address this, an efficient approach using the virtualization technology is suggested and discussed in this paper. The approach has been applied in our project to deploy a web-based integrated information system in molecular genetics labs. It is a low-cost solution to benefit both software developers and end-users.

Background:

Advances in open source have lead to the development of complex information systems which are usually implemented as component-based packages comprising third party and new software. In recent years, many web-based database applications in the field of bioinformatics (e.g. MOLGENIS **[1]**, AGL-LIMS **[2]**, DraGnET **[3]**) have been developed and released under GPL which can all be described as complex open source information systems (COSIS). Not surprisingly, most require complex installation procedures for operating system (OS), compiler, software framework, web server, database server, libraries, and other software components. Traditionally, the installation and configuration has to be done in the deployment environment, which is not under the control of the developer. As a result, the installation is complicated and often fails. It can therefore be concluded, that the widespread adoption of COSISs is severely limited by their difficult installation process.

Description:

Over the last years, hardware virtualization or "hypervisor" technology has been rapidly adopted in computing centers to make efficient use of existing hardware. The installation of software packages on the basis of a hypervisor has the same level complexity as a fresh install of an OS and the associated software. OS virtualization is a different approach which is sometimes used to facilitate software demonstration. However, it is often associated with sluggish performance. The installation of such a virtual machine (VM) on a standard host OS like Windows or Linux is very simple always resulting in a fully configured system. Here, we are proposing to use the OS virtualization approach not for testing COSISs where unresponsiveness may be acceptable, but rather for production mode operation.

Virtualization technology integrates the OS and the fully-configured application into a unit called "Virtual Appliance" (VA). Since VAs are encoded in one large file, they can easily be distributed and executed in VM software such as VMware Player **[4]** or VirtualBox **[5]** which are freely available. Therefore, the installation becomes simple and amounts to downloading the VA file to the target machine which may be either Windows or Linux. CryoWeb **[6]**, which has been deployed using the OS virtualization, is a typical example in this context.

Obviously, a COSIS that is available as a VA would make it accessible to a much larger circle of users. However, use of VA for the deployment of COSISs will only be a good strategy if the

© 2011 Biomedical Informatics

appliance is sufficiently fast for normal production mode operations. Investigation of this issue is the objective of the case study.

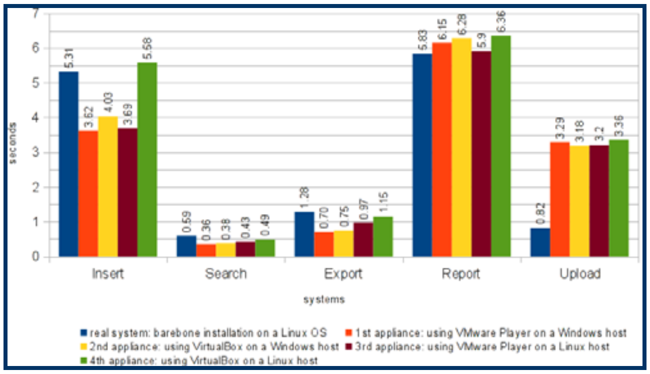

Figure 1: Benchmark results on five systems running MolabIS. The mean response time of ten runs are presented in seconds (vertical axis). Five test cases from left to right (horizontal axis) are: inserting 50 samples into the database (insert), searching 500 samples in the database (search), exporting 7000 microsatellites to a CSV file (export), reporting the details of 500 samples to a PDF file (report) and uploading a 3 MB electrophoresis file to the database (upload), respectively.

Case study:

MolabIS **[7]** is a web-based information system for storing and managing molecular data. All functionality of the system is operable through a web browser. In production mode, there may be multiple lab users accessing the system for data entry and retrieval. Typically, the use of MolabIS would require the lab users to install and configure Apache web server, Postgres database server, APIIS application framework **[8]**, Java libraries and Perl modules on a Linux platform. The installation will be painful and time consuming. In contrast, the installation of a ready-to-use appliance (available at the project homepage **[7]**) is done in a matter of minutes.

Our hypothesis is that the VAs' execution speed is fast enough for production mode operations of COSIS like MolabIS. To test this, we evaluated the performance of MolabIS in five environments. The first was a real system using barebone installation of the COSIS, the other four used Windows and Linux as host systems with VMware Player and VirtualBox as virtualization hypervisors on each **(Figure 1)**. Five test cases were chosen to represent typical production mode operations of MolabIS **(Figure 1)**. For each test case, the wall clock time between receiving the client request at the backend and its execution was measured in each environment. **Figure 1** presents the mean of ten such replicates. All tests were run on the same hardware (Intel Core i5 2x2.5 GHz processor and 6 GB of RAM).

The VA's execution speed is indeed fast enough for production mode as shown in **Figure 1**. In some cases the appliances-based COSIS even gives a better performance than the real system, a difference that is irrelevant in practical terms. Similarly, the differences among VM software found between Windows and Linux or between VMware Player and VirtualBox were negligible. As can be seen in the "Upload" case, the benchmarks indicate that VAs incur more overheads in data transactions increasing the size of virtual disks. However, for database applications like MolabIS, this virtualization overhead is acceptable.

Conclusion:

Virtualization is an efficient technique for the deployment of COSISs thereby expanding the target community considerably, opening up areas of use that would otherwise not be accessible. Apart from ease of installation, such VAs can run on different OSs, which would not be possible with a barebone installation. The benchmarks revealed no practically significant difference in response time between barebone installation and VAs, VM software and host OSs. Thus, VAs can be used effectively in the production mode for COSISs like MolabIS.

An added advantage of this approach is that VAs can be tested prior to distribution. Once packaged and tested misconfiguration during installation can be ruled out. VAs can be quickly deployed on different platforms regardless of the hardware variations of the physical servers.

References:

[1] Swertz MA *et al. Bioinformatics* 2004 **20**: 13 [PMID:15059831]
[2] Jayashree B *et al. BMC Bioinformatics* 2006 **7**: 383 [PMID:16914063]
[3] Dunca S *et al. BMC Bioinformatics* 2010 **11**: 100 [PMID:20175920]
[4] http://www.vmware.com
[5] http://www.virtualbox.org
[6] Duchev Z *et al. Bioinformation* 2010 **5**: 5 [PMID:21364801]
[7] http://www.molabis.org
[8] Groeneveld E. *Livest. Prod. Sci.* 2004 **89**: 297

Edited by P Kangueane
Citation: Truong & Groeneveld, Bioinformation 7(4): 152-153 (2011)

License statement: This is an open-access article, which permits unrestricted use, distribution, and reproduction in any medium, for non-commercial purposes, provided the original author and source are credited.