MULTI-NODE DATABASE APPLICATION DEPLOYMENT ON LAMP ARCHITECTURE

M. M. Chavan and Y. V. Dongre
Department of Computer Engineering,
Vishwakarma Institute of Information Technology, Pune

ABSTRACT

Recent trends and future application are immersed to have need of database to be deployed on multiple nodes so as to enhance the performance of database application. Recently many database applications in PHP and MySQL can be deployed on standalone node or multiple node but it is necessary to prove practical utility of application deployment on multiple node. This paper gives extract of such practical utility with Linux base as it is widely used in information technology industries and proposes a probabilistic approach to deploy the given database application on multiple node. The existing deployment method with new strategy in terms of efficiency on multiple node in Linux is presented in this paper.

KEYWORDS: *3-Tier architecture, PHP, MYSQL, LAMP, Apache Server Httpd.*

I. INTRODUCTION

In the last decade, the field of RDBMS application development has witnessed strong demands for LAMP based development and deployment. The two most commonly used ways in deployment as follow: (1) Single node where PHP as well as Mysql database resides on same node, (2) Multi-node where PHP with web server resides separately from Mysql database server. The advantage of second option is that the user is able to obtain required data from database server (MySQL) through web-server (PHP) on their browser efficiently and remotely. In this paper we present the technique that deal with some performance issues related to MySQL and PHP and an application built with this architecture. A large number of web-sites are built using PHP and MySQL on the Linux platform with Apache as the web server. This combination is known as the LAMP architecture whose components are Linux, Apache, MySQL and PHP.[1]

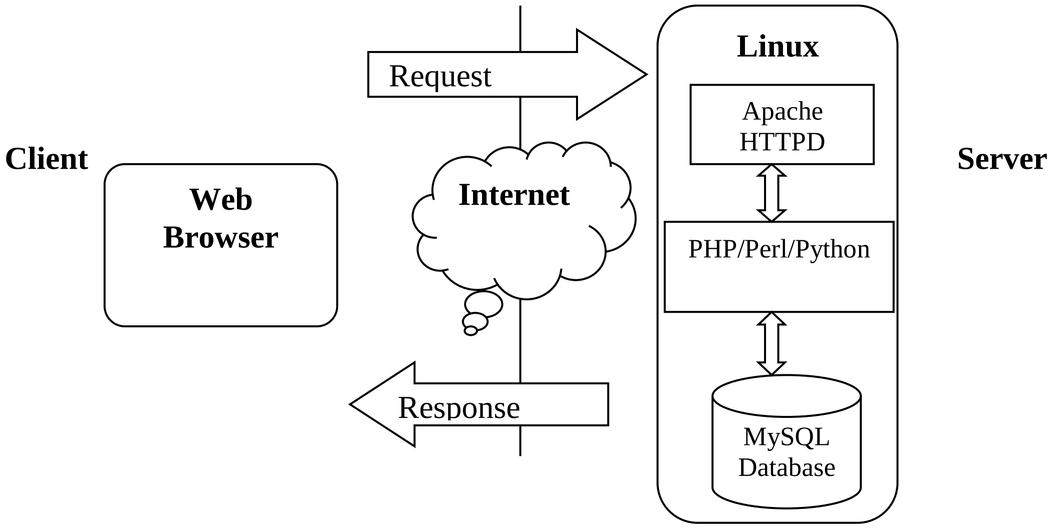


Fig.1 LAMP Architecture (Linux-Apache-Mysql-PHP)

The paper has been organized as follows: Related work is mentioned in section 2. Section 3 describes the technique and proposed approach. Section 4 presents the experimental results. Section 5 describes the experimental results and conclusion with future work is mentioned in Section 6.

II. RELATED WORK

In the paper [1] UV Ramana and TV Prabhakar investigated performance of the LAMP architecture, MySQL and PHP components then investigated the performance of the application when ported to Windows with running IIS and Apache with MySQL and PHP. In this work also issue of node independence for deployment is hidden. M. Syggouroglou et al in [2] developed application using PHP and Mysql but not mentioned where PHP and Mysql is deployed either on same node or different nodes. The author Musa Zafar et al in [3] compared various performance measures of Perl, Python and PHP separately without a database interface and in conjunction with their MySQL Database Interface Modules within a LAMP framework but programmatic solution to separation of database node (MySQL) from other node is not clearly written in this paper. The author Atul Mishra in [4] compared PHP with ASP.NET for development of web application also explore and compare suitability of selected technologies for comparing features involved in imposing restrictions on accessing navigation and functionality etc. In this papere LAMP server is used which includes: **L -** Linux Operating system, **A -** Apache server, **M -** MySQL database, **P -** PHP programming.

III. PROPOSED SOLUTION

3.1 Service Oriented servers:

3.1.1 *Database Server:* A computer program that provides database services to other computer programs or computers is termed to be a database server e.g. client–server model as mentioned in [5]. Database server functionality are often provided by Database management systems. Such a server is accessed from user’s computer which displays requested data or the server that handles tasks such as data analysis and storage.
3.1.2 *Web Server:* Each client request is being processed and the requested web pages are delivered to the client by the web server. These web pages are nothing but HTML documents and any additional content that may be included by a document, such as images, style sheets and scripts. [5] describes that a user agent called web browser initiates communication by making a request for a specific resource using HTTP and the server responds with the content of that resource.
3.1.3 *Application Server:* The function of application server is to present data to the client, i.e. deliver the contents of application using web server and database server to the client. PHP programming language is used to deliver contents that interacts with database.

3.2 Experimental setup:

(a) 10.11.9.2 (Client) **(b) 10.11.9.1 (Application/Web server)** **(c) 10.11.9.3 (Database server)**

Fig.2 Experimental Setup

The above Fig.2 represents the experimental setup. In which Fig.2(a) shows a node having ip address for example,10.11.9.2 where web browser retrieves HTML page from another node on which Apache Httpd server is running with ip address for example 10.11.9.1. Fig. 2(b) contains HTML page and application written in PHP programming language. Fig. 2(c) shows node with ip address for example 10.11.9.3 on which MySQL database resides which is accessed by PHP program and execution result is displayed on client node with the help of web browser. Following steps are required for implementing above concept:

A) To connect PHP program with MySQL database consider following steps:

1) Connection with MySQL database

```
<?php
    $con=mysqli_connect("localhost", "root", "mysql", "test");
if (mysqli_connect_errno ())
{
  echo "Failed to connect: " . mysqli_connect_errorno();
}
```

//Create database

```
$sql= "Create database test";
if (mysqli_query($con,$sql))
   {
      echo "Database test created successfully";
   }
else
{
      echo "Error creating database: " . mysqli_error($con);
}
```

3) **Perform a database query.**

4) Display data on application

5) Close the connection

B) Sample code:

Step1)Goto terminal and start mysqld service using following commands

```
            #Systemctl start mysqld
            #Systemctl enable mysqld.service
            #mysql –u root –p
            password:mysql
            #connect mysql
```

Step 2) Create database test

Step 3) use test

Step 4) create table student (id integer, name varchar (20), age integer);

Step 5) Goto the following location and type insert.php [6]

/var/www/html/insert.php

insert.php

```
<?php
             $ Roll No =$_POST['Roll No'];
            $First name=$_POST['First name'];
            $Last name=$_POST['Last name'];
            $GR NO=$_POST['GR NO'];
            $Year=$_POST['Year'];
            $con=mysqli_connect("localhost", "root", "mysql", "test");
if (mysqli_connect_errno ())
{
  echo "Failed to connect: " . mysqli_connect_errorno();
}
 mysqli_query($con, "insert into student values ($Roll No, '$First name', '$Last name', $GR NO,
$Year)");
$result=mysqli_query($con, "select * from student");
while($row=mysqli_fetch_array($result))
{
```

```
echo $row['Roll No']. "  ". $row['First name']. "   " . $['Last name']."   " . $['GR NO']. "   " .
$['Year'];;
   echo "<br>";
}
mysqli_close($con);
?>
```

Step 6) In the same location type TEDemo1.html

TEDemo1.html

```
     <html><body>
     <form action= "insert.php" method= "POST">
     <br>  Roll No:<input type= "text" name= "Roll No"/>
      <br> First name:<input type= "text" name= "First name"/>
      <br> Last name:<input type= "text" name= "Last name"/>
     <br> GR NO:<input type= "text" name= "GR NO"/>
     <br> Year:<input type= "text" name= "Year"/>
     <br> <input type= "Show" value= "Show" />
      <br> <input type= "insert" value= "insert" />
     </form>
     </body>  </html>
```

Step7) Open httpd.conf file from following location

computer\etc\httpd\httpd.conf and type following lines

```
Listen  xxx.xxx.xxx.xxx (ip of local machine)
ServerAdmin root @ xxx.xxx.xxx.xxx
ServerName xxx.xxx.xxx.xxx:80
```

And save the file.

Step 8) Start the apache server using following commands

```
#sudo service httpd start
 # service httpd enable
```

Step 9) Stop the services of firewall as well as iptables and allow MySQL databse only from a certain network on port 3306 using following commands

```
#service iptables stop
#service firewalld stop
#/sbin/iptables –A INPUT –i eth0 –s XXX.XXX.XXX.0/24 –p tcp –destination-port   3306 –
j ACCEPT
#service iptables save
```

Step 10) Open web browser and type the URL as

XXX.XXX.XXX .XXX/TEDemo1.html

Here XXX.XXX.XXX .XXX is an ip address of machine on which TEDemo1.html is present. Following table describes the different commands which are required for establishing connection among 3 tiers i.e. bottom tier (Database server),middle tier(Apache server-HTTP) and top tier(Client application-PHP).

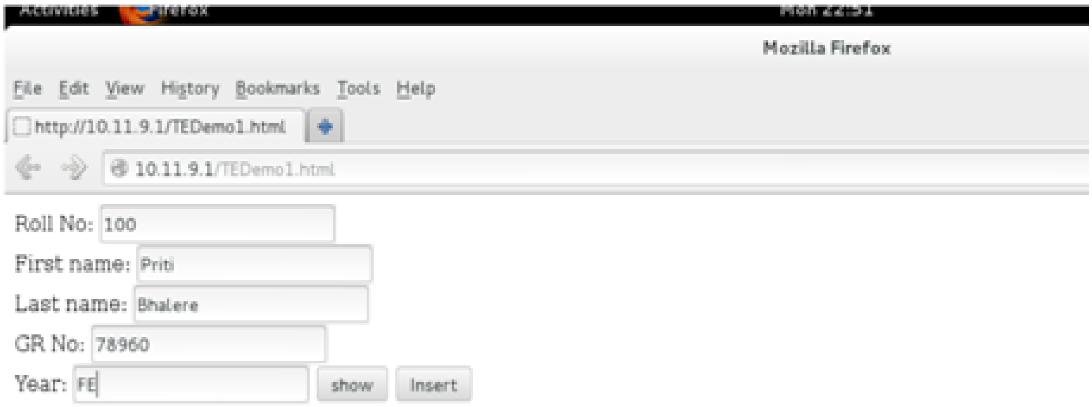
Table 1. Commands required for connection in network

Steps	Command	Description
1	#systemctl start mysqld	To start the mysql service
2	#systemctl enable mysqld.service	To enable the mysql service
3	#mysql –u root –p	These are command line options passed to mysql. *-u* stands for the username and -p stands for the password.
4	#connect mysql	Connecting to mysql from command line
5	#create user myuser identified by 'myuser'	Creating user
6	#grant All privileges on *.* to 'myuser'@' XXX.XXX.XXX .%'	The GRANT statement grants privileges to MySQL user accounts.
7	#grant usage on *.* to 'myuser'@' XXX.XXX.XXX .%' identified by 'myuser'	The GRANT statement grants usgae to MySQL user accounts.
8	#Commit	Commit changes

9	#service iptables stop	To disable iptables on Linux
10	#service firewalld stop	To disable firewall on Linux
11	#/sbin/iptables –A INPUT –i eth0 –s XXX.XXX.XXX.0/24 –p tcp –destination-port 3306 –j ACCEPT	Allow MySQL only from a certain network on port 3306
12	service iptables save	To *save* net filter rules sbin.
13	#mysql – u myuser –h XXX.XXX.XXX .xxx –p db	Connect with database db from certain network
14	#sudo service httpd start	To start the Apache httpd server
15	# service httpd enable	To enable the Apache httpd server

IV. EXPERIMENTAL RESULTS

Here the client sends request with the help of browser by including an ip address of the machine on which php/html page is residing. The resultant page is displayed on client machine as shown in Fig.3 in which client on first node inserts data and submits the information.



http://10.11.9.1/TEDemo1.html

10.11.9.1/TEDemo1.html

Roll No: 100

First name: Priti

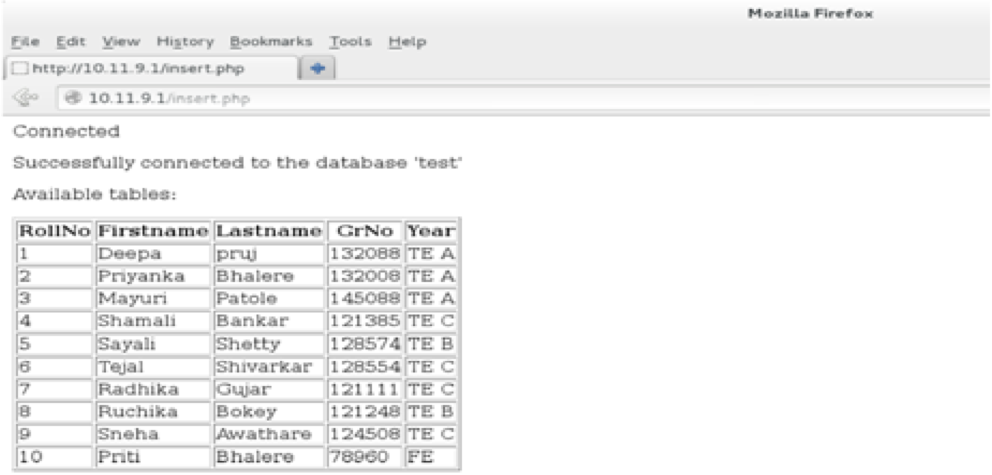
Last name: Bhalere

GR No: 78960

Year: FE

Fig.3 Client Request

After substituting the above values, the Apache Httpd server will be connected to the database server in turn updating the database as shown in Fig 4.



http://10.11.9.1/insert.php

10.11.9.1/insert.php

Connected

Successfully connected to the database 'test'

Available tables:

RollNo	Firstname	Lastname	GrNo	Year
1	Deepa	pruj	132088	TE A
2	Priyanka	Bhalere	132008	TE A
3	Mayuri	Patole	145088	TE A
4	Shamali	Bankar	121385	TE C
5	Sayali	Shetty	128574	TE B
6	Tejal	Shivarkar	128554	TE C
7	Radhika	Gujar	121111	TE C
8	Ruchika	Bokey	121248	TE B
9	Sneha	Awathare	124508	TE C
10	Priti	Bhalere	78960	FE

Fig.4 Server Response

Above experiment shows glimpse of data retrieval and submission on multiple nodes using LAMP architecture.

V. CONCLUSION AND FUTURE WORK

In this paper, the performance of a web server on Linux platform is analyzed with the help of LAMP process. Two web technologies viz. Apache Http server and PHP are used to develop the applications. Such applications developed using PHP are more efficient and reliable. Though it is clear that LAMP is a reliable and effective platform used for deploying web application on multi-node architecture, use of other scripting languages in order to get better results can be implemented and compared with the above concept in future. For example, the same database deployment application can be implemented with the use of JSP/PERL web technologies as well as the comparison analysis of the same can be done with the above proposed approach in future.

REFERENCES

1]. UV Ramana, TV Prabhakar, "Some Experiments with the Performance of LAMP Architecture", Proceedings of the The Fifth International Conference on Computer and Information Technology by IEEE Computer Society Washington, DC, USA, 2005. pp 916-921

2]. M. Syggouroglou, L. Frangidis, V. Mardiris & V. Chatzis, "An Internet Application for Lab-Group Booking Using Open Source Software", Proceedings of the 3rd WSEAS/IASME International Conference on Educational Technologies, Arcachon, France, October 13-15, 2007

3]. Musa Jafar, Russell Anderson and Amjad Abdullat,"Comparison of Dynamic Web Content Processing Language Performance under a LAMP Architecture", Proc CONISAR 2008, v1 (Phoenix), 2008

4]. Atul Mishra, "Critical Comparison of PHP and ASP.NET for Web Development" INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 3, ISSUE 7, JULY 2014

5] Mr. J. Mark Jain ME, J.Ferlin Getziah , A.Hameeda , N.Naveena , K.Nasren Subaitha Banu ," Virtually Integrated Multi-Purpose Server", Vol. 3 (2) , 2012,3719-3725,ISSN:0975-9646.

6] K. Valarmathi, "How to Connect PHP into MYSQL Database", CSI Communications, Volume No.37 Issue No 12, 2014.

AUTHORS

Madhuri M. Chavan received the ME degree in Computer sci. & Engg. from D.Y Patil college of Engg. & tech., Kolhapur in 2012. She is working as Assistant Professor in Vishwakarma Institute of Information Technology, Pune. Her research interests include Data Mining, Database Management Systems.

Y. V. Dongare has completed ME(CSE) in year 2009 from Vishwakarma Institute of Technology, Pune He is presently working as assistant professor in the department of computer engineering at Vishwakarma Institute of Information Technology, Pune, Maharashtra state, India. His research interests include Database Management Systems and Design, Software Architecture.