

## WebCall Ticket Tracking System

Updated: 5 May 2003

### Contents

|  |    |
|--|----|
| Scope .....  | 2  |
| Summary of Document.....                                     | 2  |
| Which Operating Systems does WebCall Support.....            | 2  |
| WebCall has been tested with ..                              | 2  |
| Install and Configuring WebCall from the Start.....          | 2  |
| DBI and DBD Explained.....                                   | 3  |
| How to Add a new department.....                             | 3  |
| Creating a Department Table in WebCall Database .....        | 3  |
| Edit Config.pl .....   | 4  |
| Create Group Table.....                                      | 4  |
| Searching/Opening Tickets for modification .....             | 5  |
| Selecting OPEN tickets only .....                            | 5  |
| Selecting by Customer .....                                  | 5  |
| Keyword Searching.....                                       | 6  |
| Database Corruption and Repair .....                         | 6  |
| APPENDIX – dbcheck.sh.....                                   | 7  |
| APPENDIX – WebCall Core Configuration File - Config.pl ..... | 7  |
| APPENDIX - How to install the Perl module DBI on Linux ..... | 9  |
| By Phil Jones, July 00 .....                                 | 9  |
| Index.....   | 9  |
| About this document .....                                    | 9  |
| Explanation of DBI and DBD.....                              | 10 |
| What this article doesn't do.....                            | 10 |
| Please correct my errors.....                                | 10 |
| Assumptions .....  | 10 |
| Starting point .....   | 10 |
| Overview.....  | 10 |
| Checking for prerequisites.....                              | 10 |
| Explanation of the packages the shell script tests for ..... | 12 |
| Installing prerequisites .....                               | 12 |
| Creating data in MySQL .....                                 | 12 |
| Obtaining DBI and DBD .....                                  | 12 |
| Install DBI.....   | 12 |
| Quick help.....  | 13 |
| Install DBD.....   | 13 |
| Quick help.....  | 13 |
| Testing.....   | 13 |

## Scope

This document explains inner workings of WebCall Ticket Tracking System, walks through the installation on a standard Linux System.

### Summary of Document

- 1) Install WebCall
- 2) How to Add new Departments
- 3) APPENDIX - Core Configuration of WebCall – Config.pl
- 4) Database Corruption and Repair

## Which Operating Systems does WebCall Support

Webcall was designed and developed on Linux. I have heard rumors that WebCall is running on Windows without too many modifications, although I have not seen this. If anybody is doing this, please send me some documentation and I will add it to the main WebCall Documentation.

### WebCall has been tested with ..

- 1) Redhat 7.x and 8.x
- 2) Mandrake 7.x, 8.x and 9.x
- 3) Suse 7.x Linux

WebCall should work with any Linux/BSD/HP-UX/Solaris System that supports mySQL, DBI/DBD and Apache.

## Install and Configuring WebCall from the Start

- 1) Login as root or su to root
- 2) Ungzip Webcall-1xx.tar.gz from Archive file into a directory under the root home directory in your webserver directory.  
E.g.
  - a) `cd /home/www`
  - b) `gzip -dc WebCall-xxx.tar.gz | tar -xvf -`

This will ungzipped all the files in /home/www/webcall

- 3) Edit /home/www/webcall/bin/config.pl  
Most of this is common sense variables, path names, domains etc
- 4) Setup the Tables for WebCall
  - `$HOME/webcall/bin/dbsetup-webcall` (Main webcall table)
  - `$HOME/webcall/bin/dbsetup-users` (users table)
- 5) Edit /home/www/webcall/index.html and set it to the correct URL if necessary, generally this can be left alone.
- 6) Point your browser to the URL <http://www.domain.com/webcall>  
You should be able to login with `admin/admin`

### DBI and DBD Explained

WebCall relies on PERL DBI and DBD modules to be installed. You might need to install these modules to get WebCall to work.

Configuring DBI and DBD is fairly straight forward, but can prove quite challenging to configure the various modules if done incorrectly. I have provided the following link to help out. The latest version of mySQL and Perl would be required.

An explanation how to install DBI and DBD can be found here. I also added the page as an Appendix in case the site ever goes down. Most newer Linux systems have these installed by default, so it is possible you don't have to do anything.

<http://timesync.gmu.edu/OSCR/howtoDBlonLinux.html#obtaining>

### How to Add a new department

This section explains how to add a new department to WebCall. It consists of two parts, once create a table in the WebCall Database, two add the Department name to the config file for WebCall

### Creating a Department Table in WebCall Database

Edit the dbsetup-gen file in the WebCall Directory and Modify the \$table variable to reflect department

```
$table="newdepartment";
```

*NOTE: It is recommended, to keep the name short and lowercase, no spaces*

Then run the dbsetup-gen script.

```
[WebCall/bin]# ./dbsetup-gen
```

WebCall, Copyright (c) 2001, Steven Wishart, All Rights Reserved

```
What is the root password for MySQL? <Enter>
```

```
Enter password: <Enter>
```

```
newdepartment Table Creation complete.
```

If you see the above, the Table has been successfully created.

### Edit Config.pl

**Add Department name** to the Department Array as the Full name of the Department, spaces are allowed.

```
Departments = ("Finance", "Accounting Department");
```

Make sure you close all double quotes “ correctly.

**Create a Group Email address**, with a group email, or consisting of email address separated by comma's. Make sure you add a slash \ in front of the AT Symbol for escaping that character

```
$AccountingDepartment = "accounts\@domainname.com";
```

### Create Group Table

Remember the table name is the Table you put in dbsetup-gen, this must be the same, and CASE is important.

```
$AccountingDepartmentTable = "accounting";
```

### Searching/Opening Tickets for modification

Searching tickets is probably the trickiest part to understand fully. There are different scenario's that allow you to successfully search for a ticket.

If you click on Modify a Ticket or Query a ticket, it will default to your username and department in Accepted mode. Just clicking on the VIEW button will list all tickets assigned to your user ID with this years date starting at January. Change the month say from Jan to May will result in a quick find of tickets within the last month

### Modify/Update A Ticket

This form allows you to lookup your cases and check their status.

Ticket No.:

OR

Department:

Status:  ALL  OPEN  ACCEPTED  CLOSED  ONHOLD

Search:    -

Text Search:

### Selecting OPEN tickets only

If you click on the OPEN checkbox, it will ignore EVERYTHING, including DATE, USERNAME, Customer etc and will list all tickets in STATUS MODE OPEN. This is great for seeing current open tickets only.

### Selecting by Customer

You can select all tickets by Customer. It will use the Users field with this search as well as the DATE fields for this search. So it will only search the Customer tickets for this USER and/OR within the Date

## WebCall Ticket Tracking System

---

time from of the ticket. If no user is selected, it will search All users in which ever mode you select. Eg. ACCE:PTED, CLOSED, ONHOLD.

### Keyword Searching

Keyword searching will search for a KEY in every field in the ticket. It does only simple searches. So just enter in one keyword, such as 'MODEM', it will not find say 'MODEM REPAIRED TODAY'. If you enter MODEM TODAY, it will not find the text.

### Database Corruption and Repair

If you determine your database is Corrupt, you need to Shutdown MySQL.

- 1) Shutdown MySQL Complete (/etc/rc.d/init.d/mysql\* stop)
- 2) Run ./checkdb.sh in the \$HOME/WebCall/bin/dbcheck.sh against the suspect TABLE  
This should tell you if there is a problem with the suspected table
- 3) **BACKUP** the /var/lib/mysql/webcall directory  
**tar -cvf /tmp/webcall.backup.tar /var/lib/mysql/webcall**
- 4) If there is a problem, uncomment the second part of script to repair the database  
Uncomment lines  
./checkdb.sh
- 5) Comment last lines back, re-run dbcheck.sh  
No errors should be displayed
- 6) Restart Mysql (/etc/rc.d/init.d/mysql\* start)

### APPENDIX – dbcheck.sh

```
#!/bin/sh
# http://www.mysql.com/doc/R/e/Reproducible_test_case.html
# This script will run a check against the DB's
# Stop MySQL Daemon First
mysamchk -s /var/lib/mysql/webcall/*.MYI

# Run Repair against TABLE - Uncomment when needed
# mysamchk -r /var/lib/mysql/webcall/calls.MYI
# mysamchk -r /var/lib/mysql/webcall/users.MYI
```

### APPENDIX – WebCall Core Configuration File - Config.pl

Config.pl is the hear of WebCall. It contains variables, departments, email address which all make WebCall work.

```
# Global Variables

# Site Information
$Site = "http://www.domain.com/WebCall";
$Site_Name = "WebCall, Copyright (c) 2002, All Rights Reserved.";

# Path Configuration
$ProgramDir = "/home/www/WebCall";
$BinDir = "bin";
$DataDir = "data";
$ConfigDir = "config";
$ImageDir = "images";
$HelpDir = "help";
$error_Log = "$ProgramDir/logs/error.log";
$Syslog_File = "$ProgramDir/logs/webcall.log";
$Users_File = "$ProgramDir/$ConfigDir/users.dat";
$Severity_Levels = "$Site/$HelpDir/priority.html";
$Company_Logo = "$Site/$ImageDir/logo.gif";

# E-Mail address of System Administrator
$AdminEmail = "user\@domain.com";

# Admin Description
$AdminName = "WebCall Center";

# Admin Level
$AdminLevel = 6;

# Email Domain
$EmailDomain = "domain.com";
```

## WebCall Ticket Tracking System

---

```
# System Domain
$DomainName = "domain.com";

# Number of Priority / Severity Levels
$SeverityLevels = 9;

# Group Emails
$SystemsEngineering = "user\@domain.com";

# Departments
# Note it is CASE sensitive and must match the below department tables
# EXACTLY without the spaces
@Departments = ("Systems Engineering");

# Department Tables matching the above name, with space, including case
$SystemsEngineeringTable = "calls";

# Customers
@Customers = ("TC", "WC");

@CustomersDesc = ("Test Customer", "Test Department");

# Topics
@SystemEngineeringTopics = ("PC Issues", "Network Issues", "Office
Applications", "Create Accounts", "Printer Issues", "Application
Issues", "Add/Fix Routing", "System Upgrades", "HPUX Server
Configuration", "LINUX Server Configuration", "SCO Server
Configuration", "SUN Server Configuration", "CITRIX Server
Configuration", "Virus Issues", "MQ Series", "Icon Issues", "Email
Issues", "Samba", "PC Move", "Fax Modem Issues", "SarReports", "Capacity
Issues", "Patch Installation", "Disk Failure", "Server Performance",
"WebCall Modifications", "MC/ServiceGuard", "Account Disabled",
"Password Reset", "Operations Monitor Issue", "DNS Configuration",
"Other");

# Standard Phone Numbers
@PhoneNumbers = ("(555) 555-5555");

# System Variables
$TRUE = 1;                # Set True to 1
$FALSE = 0;              # Set False to 0
@Month = ("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep",
"Oct", "Nov", "Dec");
@Month2 = ("January", "February", "March", "April", "May", "June",
"July", "August", "September", "October", "November", "December");

# Mode of files written.
$CHMOD_MODE = 00666;
$CHOWN_MODE = 1000;
$CHGRP_MODE = 1000;

# Database Settings
```

## WebCall Ticket Tracking System

---

```
$table = "calls";  
$database = "webcall";  
$data_source = "dbi:mysql";  
$user = "webcall";  
$passwd = "webcall";  
  
1; # Return true for library
```

### APPENDIX - How to install the Perl module DBI on Linux By [Phil Jones](#), July 00

## Index

1. [About this document](#)
2. [Explanation of DBI and DBD](#)
3. [What this article doesn't do](#)
4. [Please correct my errors](#)
5. [Assumptions](#)
6. [Starting point](#)
7. [Overview](#)
8. [Checking for prerequisites](#)
9. [Explanation of the packages the shell script tests for](#)
10. [Installing prerequisites](#)
11. [Creating data in MySQL](#)
12. [Obtaining DBI and DBD](#)
13. [Install DBI](#)
14. [Install DBD](#)
15. [Testing](#)
16. [Feedback](#)

## About this document

The purpose of this document is to help with setting up Linux in order to write a Perl program that gets data from MySQL. Hopefully it will relieve some of the traffic on the [DBI mailing list](#). These instructions cover installing, testing and hopefully understanding more about the Perl module DBI, and its DBD for MySQL. This document is intended for those that wish to use RPM packages as much as possible.

### Explanation of DBI and DBD

DBI stands for Data Base Interface. DBI is an add-on, or 'module' for Perl which gives Perl programs access to databases. Mysql-DBD is another module, a Data Base Driver for MySQL. DBI uses the Driver as a translator to talk to MySQL. So, both modules are required.

### What this article doesn't do

This article is about making a working system in the easiest (ie, least secure) way possible. Use these instructions on a testing or learning only computer that isn't connected to the Internet.

### Please correct my errors

I am no Linux, Unix, DBI or MySQL guru. These instructions are Red Hat oriented simply because that is the Linux I am using. Although this procedure has worked for me, I can't guarantee it will work on another platform. Also I'm not all that familiar with compiling Perl modules and programs in general. If you know better, and you see something wrong with this document, please let me know so I can fix it.

### Assumptions

I assume you are using a recent Red Hat Linux distribution, or a distribution based on Red Hat which uses the Red Hat Package Manager (RPM) for installing and verifying programs. You must have root access to the system.

### Starting point

I used a Red Hat Linux 6.0 Intel computer with Apache, Perl and development tools installed. The way I set this up was to select the Apache server, development tools and development libraries when installing Red Hat.

### Overview

The outline of the procedure is:

1. Check required packages are installed
2. Install required packages that are not installed
3. Obtain Perl DBI and DBD for MySQL
4. Compile and install Perl DBI and DBD for MySQL
5. Create a sample database in MySQL
6. Create a sample table in that database
7. Create some sample data in that table
8. Create a MySQL user with access to the sample data
9. Write a simple Perl program that prints data from MySQL.

### Checking for prerequisites

Here are the basic requirements, as far as I know, for installing Perl DBI and the DBD for MySQL. This is *not* an exhaustive list. There may be other required packages which I don't know about.

1. Perl programming language
2. Make
3. Kernel headers

4. A C compiler, eg EGCS
5. C extensions for the C compiler
6. Glibc library
7. Glibc development library
8. MySQL server
9. MySQL client
10. MySQL development tools

For your convenience I have written a [shell script](#) which tests your system to see if you have these packages installed in RPM format. It only checks to see if the package exists. It doesn't verify the packages are completely installed, you can do that if you wish. Run this script, like this:

```
sh checkdbi.sh
```

If all is well, you should get output like this:

```
If there are no errors in the following list, you should
have everything you need to write a Perl program that gets
data from MySQL.
```

```
Checking to see which RPM packages are installed...
```

```
perl-5.00503-2
make-3.77-6
kernel-headers-2.2.5-15
egcs-1.1.2-12
egcs-cl1.1.2-12
glibc-2.1.1-6
glibc-devel-2.1.1-6
MySQL-3.22.32-1
MySQL-client-3.22.32-1
MySQL-devel-3.22.32-1
```

The version numbers may vary according to which Linux you are using. If you see any '`...is not installed`' messages in the output, you may need to install that package. Read the next section to learn more about what you may be missing.

## Explanation of the packages the shell script tests for

1. Perl is definitely required.
2. Make is a standard program used to simplify compiling new programs. DBI and DBD both use Make as part of the compiling process. You must have this.
3. Kernel headers are structures and constants needed for compiling most standard programs. You very likely need this.
4. EGCS is the C compiler supplied with Red Hat Linux 6.0. The compiler turns human-readable code into an executable program. If you don't have the EGCS package installed, perhaps you have another C compiler installed instead. You don't have to have EGCS, but you do need a C compiler.
5. EGCS-C adds C support to the GNU C compiler. I don't know whether C extensions are required but it seems like a good idea to have it anyway. If you are using another C compiler, this package may be different also.
6. Glibc contains standard libraries used by multiple programs on the system. This is one of the base packages in Linux so it is very likely to be installed on your system. Required.
7. Glibc-devel contains header files and object files needed when compiling programs that use the standard C libraries. Required.
8. MySQL is the database server. Required.
9. MySQL-client provides a simple, command line interface to MySQL. This is needed so we can type some data into the database, and to see whether it's been entered correctly. Required.
10. MySQL-devel makes it possible for other programs, such as DBI, to work with MySQL. Required.

## Installing prerequisites

Many up to date Linux distributions come with all these packages on the installation media. Others, including Red Hat Linux 6.0, do not supply the MySQL packages. In this case you can download them in RPM format from <http://www.mysql.com/downloads/index.html>.

## Creating data in MySQL

Read the [online MySQL documentation](#). This explains how to connect to MySQL, create a database, create a table, add some data to a table, and create a MySQL user with access to that data.

## Obtaining DBI and DBD

Download the [latest DBI module](#) for Perl from CPAN. Download it to a safe location, for example, your home directory.

Then download the [latest DBD database Driver for MySQL](#) also from CPAN. As before, download it to a safe location, like your home directory.

## Install DBI

Become root, and enter the following commands:

```
tar xvzf DBI-1.14.tar.gz
```

```
cd DBI-1.14
```

Then read and follow the instructions for compiling DBI.

### Quick help

**On perl Makefile.PL: Unable to open MakeMaker.tmp ... permission denied**

**Meaning:** You are not root.

**Solution:** Become root.

**On make test or make install: Lots of error messages like 'can't find stdio.h, math.h'...**

**Meaning:** The standard C development libraries are not installed.

**Solution:** Install the glibc-devel RPM package.

## Install DBD

Enter the following commands:

```
tar xvzf Mysql-Mysql-modules-1.2214.tar.gz
```

```
cd Mysql-Mysql-modules-1.2214
```

Then read and follow the instructions for compiling the DBD for MySQL.

### Quick help

**Problem:** You don't know the answer to the question about where MySQL is installed on your system. Specifically, it asks for an include directory, but it's not on your computer.

**Meaning:** MySQL-devel is not installed.

**Solution:** Install the MySQL-devel RPM package.

## Testing

To test your DBI installation, try [Jeffery Baker's DBI Examples](#). They are written for Oracle but to make them work with MySQL requires only a simple adjustment. There are other DBI documents at the [DBI main page](#) including examples specially for MySQL.