ctys-uc-PHP-devenv(7)

Setup of a development environment for PHP

September 29, 2020

Contents

1 General **2**

2 Setup of Host-OS, Hypervisor, and GuestOS **3**

2.1 CentOS with QEMU/KVM . . . 3

2.2 Debian with VirtualBox . . . 4

2.3 CentOS with VMware-Server and Workstation . . . 4

3 Setup of the UnifiedSessionsManager **4**

3.1 Install tgz BASE-Package + DOC-Package on Debian . . . 4

3.2 Install rpm BASE-Package + DOC-Package on CentOS . . . 4

3.3 Setup of the Gnome Menu . . . 5

4 Creation of the Inventory - cacheDB **7**

5 Graphical Start of Virtual Machines **7**

6 Creation of a virtual Appliance for Eclipse **7**

7 Creation of a Development Desktop **7**

8 Creation of a Project Menu **7**

9 Manage the VM **7**

10 SEE ALSO **8**

11 AUTHOR **8**

12 COPYRIGHT **8**

List of Figures

1 A Distributed PHP development environment . . . 2

2 A Distributed PHP development environment . . . 3

3 Simulated clients . . . 3

4 Create Menu . . . 5

5 Custom project menu . . . 6

6 Custom tool related menu . . . 6

.

1 General

This document describes the setup of a virtualized development environment as a personal workspace. The intention for this described setup is the provisioning and versioning of a developemnt environment particularly for the development of PHP based web applications including a web servers and database servers. The setup described in this document consists of the following system components

- Database-server
- Web-Server
- Clients
- Remote Displays

All components could be executed on one physical machine but are described in the given example as a distributed setup located on different PMs. The following environment is used here as a virtual development environment, which is used for modultest too.

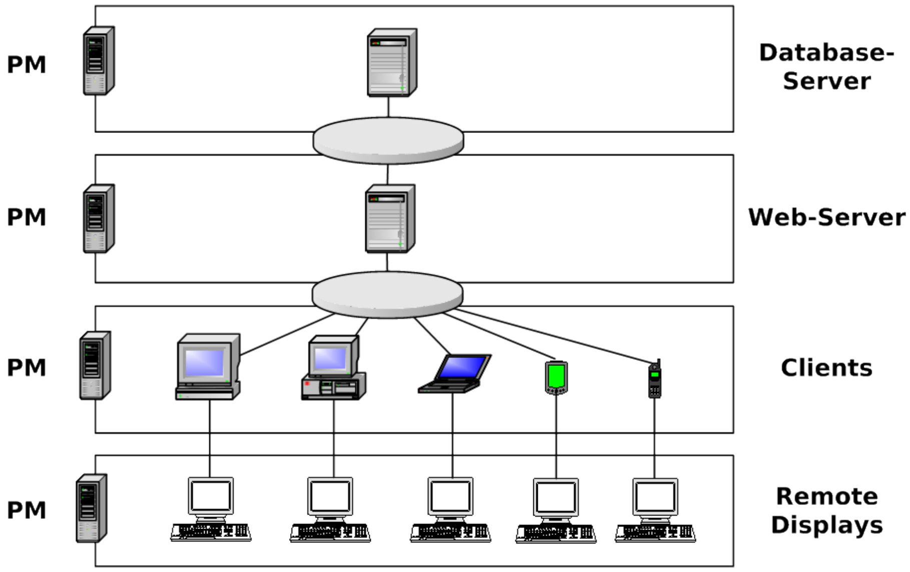

Figure 1: A Distributed PHP development environment

The ordinary development environment is extended by the administrative applications, which are hosted witin VMs too. This comprises the management applications for the MySQL server. The basic utility requires at least MS-Windows-XP as runtime environment, whereas the Eclipse based MySQL-Explorer could be executed on Linux as well. The second application is the Network Management and Monitoring entity based on Nagios. This is hosted on a virtual Linux machine running CentOS.

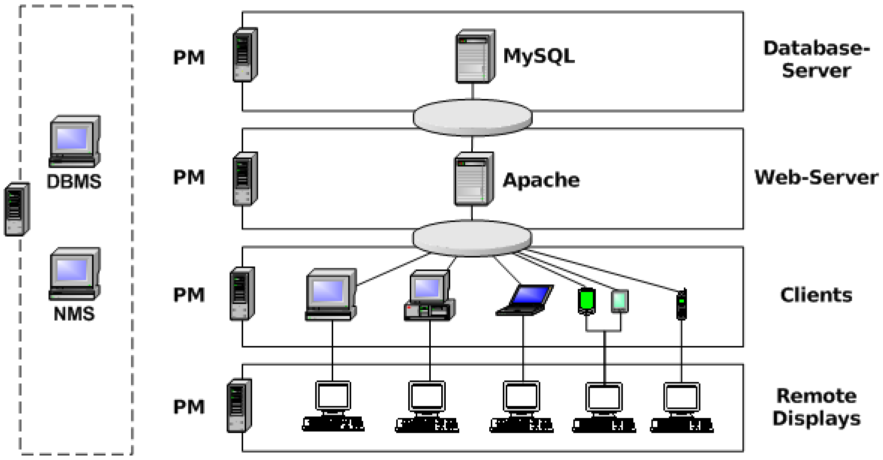

Figure 2: A Distributed PHP development environment

The client virtualization particularly provides a cheap and simplified facility for the development and test of various clients with miscellaneous operating systems and applications. This particularly simplifies the versioning and storage of pre-configured complete system sets and provides for development, test and systems integration environments.

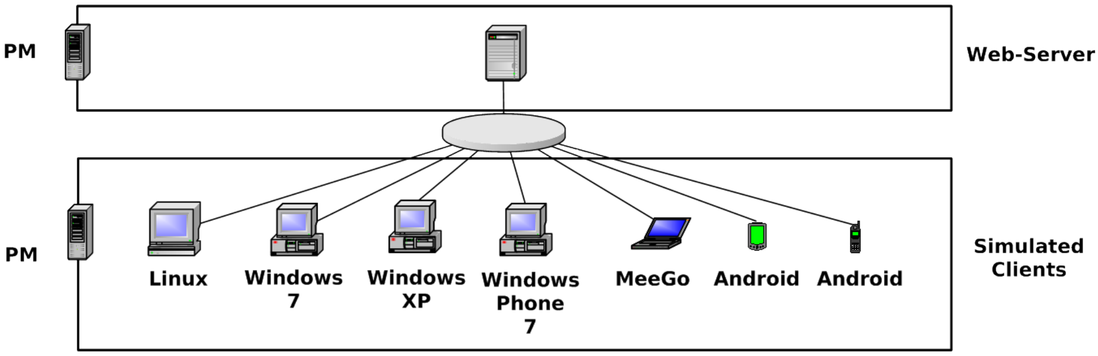

Figure 3: Simulated clients

.

2 Setup of Host-OS, Hypervisor, and GuestOS

The installation for the following variants has to be performed by the appropriate standard setup of the HostOS and hypervisor.

2.1 CentOS with QEMU/KVM

Here the standard distribution is installed. Additional packages are vde2-2.2.3 and Qemu-0.12.2, which are build and installed to '/opt'. The vde2-2.2.3 package for network encapsulation requires a symbolic link

```
ln -s /opt/vde2-2.2.3 /opt/vde
```

The wrapper vde may not be required, when the Qemu support option is compiled in, but this is not yet widely the case. Thus vde2 is still utilized as standard. For additional information refer to ctys-configuration-QEMU(7) and ctys-uc-CentOS(7) .

2.2 Debian with VirtualBox

The OSE edition is missing RDP access, thus supports synchronous consoles for display only. When RDP is required, which is recommended, install the download version instead of the OSE edition shipped with the distribution. The alternative is to install the OSE edition and additionally the extension components which are licnesed slightly different. For additional information refer to ctys-configuration-VBOX(7) and ctys-uc-debian(7) .

2.3 CentOS with VMware-Server and Workstation

This straight forward. For additional information refer to ctys-configuration-VMW(7) and ctys-uc-CentOS(7) .

3 Setup of the UnifiedSessionsManager

3.1 Install tgz BASE-Package + DOC-Package on Debian

1. Unpack the tar-gzip-archive and apply the standard installation procedure, where the call has to be executed by typing the fully qualified absolute path when ambiguity could occur. This is due to automatic usage of consistent libraries for the install procedure.

   ```
   ctys-distribute -F 2 -P UserHomeCopy root@lab02
   ```

2. Open a Remote Shell by call of CLI plugin:

   ```
   ctys -t cli -a create=l:tst137 root@tlab02
   ```

3. Check the plugins states by calling ctys-plugins:

   ```
   ctys-plugins -T all -E
   ```

For additional information refer to ctys-multisite-install(7) and ctys-distribute(1) .

3.2 Install rpm BASE-Package + DOC-Package on CentOS

The following steps are required for a RPM based setup on CentOS. The installation is relocatable, but located at '/opt', and installed locally by 'ctys-distribute'.

1. Install BASE package.

   ```
   rpm -i ctys-base-01.11.011.noarch.rpm
   ```

2. Now install a a local version, here by copy. The PATH prefix is important here, particularly in case of updates. The path is resolved to it's actual path by eliminating any symbolic link, and used for consistent link of libraries.

   ```
   /opt/ctys-01.11.011/bin/ctys-distribute -F 2 -P UserHomeCopy
   ```

3. Next the menu is setup.

   ```
   ctys-xdg --menu-create
   ```

4. Now the help is available as eihter a Gnome or KDE menu. Alternatively could be called from the commandline.

For additional information refer to ctys-multisite-install(7) and ctys-distribute(1) .

3.3 Setup of the Gnome Menu

The first supported user interface is Gnome, others are going to follow. The current version provides a desktop based GUI only, thus the provided command line interfaces are integrated into the desktop and could be started by means of the windows manager(screenshots). The next step is the setup of a custom menue, which could be extended by the templates provided by the tool ctys-xdg(1) . The first supported user interface is Gnome, others are going to follow. The current version provides a desktop based GUI only, thus the provided command line interfaces are intergated into the desktop and could be started by means of the window manager. Additional information is available by ctys-configuration-Gnome(7)
The setup of the Gnome Menu is quite simple, the call

```
ctys-xdg --menu-create
```

sets up a basic standard menu:

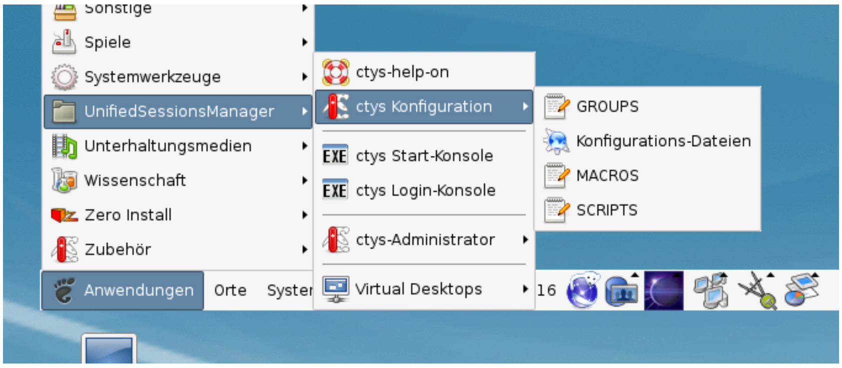

Figure 4: Create Menu

The setup could be targeted either for private menus or shared menus. Both setups are based on a menu template, which is stored in the configuration subdirectory 'xdg.d'. The call

```
ctys-xdg --menu-cancel
```

removes the installed files. For current version no checks for changed files is done. The menues could be edited and extended by the call

```
ctys-xdg --menu-edit
```

which opens the related directories for modification of '*.menu', '*.desktop', and '*.directory' files.

This could be easily extende to a project specific menu set containing single machines as well as complete desktops and documentation sets.

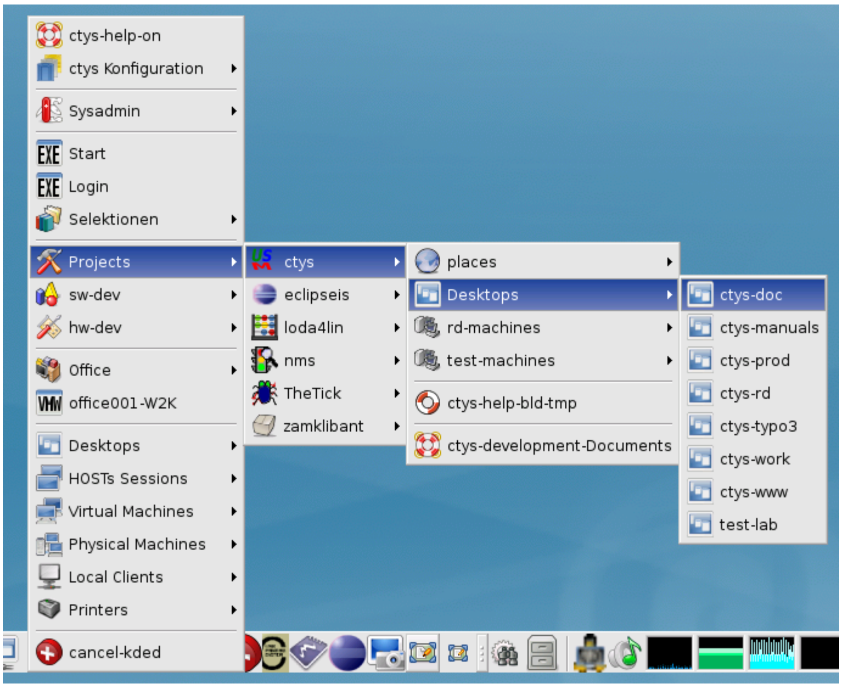

Figure 5: Custom project menu

The alternative could be a tool related environment.

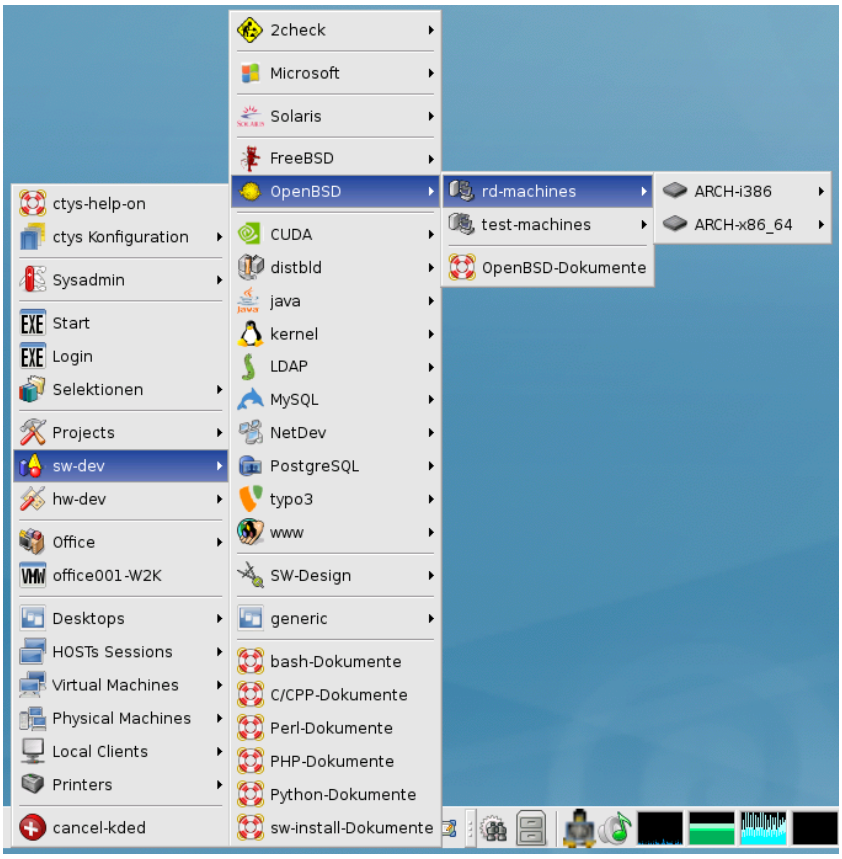

Figure 6: Custom tool related menu

4 Creation of the Inventory - cacheDB

The optional creation of a inventory database enables the use of user defined shortcuts, so called labels. This database could be created fully automatic with a little preconfiguration such as providing the list of the remote servers and the remote directory tree to be scanned. Additional information is available from **ctys-vdbgenVM(1)** and e.g. **ctys-uc-CentOS(7)**

5 Graphical Start of Virtual Machines

Once the menu is created and the VMs are either scanned into the database or added manually, these could be started by the systems menu entries. This is independent from the actual used OS. The basics patterns and configuration options are depicted in the document **ctys-configuration-Gnome(7)**

6 Creation of a virtual Appliance for Eclipse

Eclipse is one of the most accepted development and applications environment, but lacks in some cases from version dependencies of it's components. Thus the creation of a virtual appliance with a consistent set of components porvides a simple means for the distribution of comprising development environments.

7 Creation of a Development Desktop

ffs.

8 Creation of a Project Menu

ffs.

9 Manage the VM

ffs.

10 SEE ALSO

ctys(1) , *ctys-configuration-QEMU(7)* , *ctys-configuration-VBOX(7)* , *ctys-createConfVM(1)* , *ctys-QEMU(1)* , *ctys-uc-QEMU(7)* , *ctys-uc-VBOX(7)* , *ctys-VBOX(1)* , *ctys-vhost(1)*
For System Tools:
Android: [http://www.android.com]
Android-x86: [http://www.android-x86.org]

11 AUTHOR

Arno-Can Uestuensoez <https://arnocan.wordpress.com/>
<https://unifiedsessionsmanager.sourceforge.io/>
<https://github.com/unifiedsessionsmanager>

Member #8673

12 COPYRIGHT