



Getting rid of unnecessary ballast with Suse 9.1

Low-fat Suse

More features, more software – more ballast. Despite all the assurances that “Linux will run on older hardware”, non-state-of-the-art computers really look their age when you ask them to handle a modern distribution like Suse 9.1.

BY HAGEN HÖPFNER

In real life, things are not quite as rosy. As previously mentioned [1], automatic drive mounting using **subfs/submount** does not always work reliably. KDE has a reputation for being memory-hungry, and unnecessary services add to the security risk.

Disabling the latter is definitely recommended. If you replace the KDE desktop environment with another convenient file manager, even older hardware should be able to hold its own. If you have been using Linux long enough to handle mounting and unmounting media yourself, you can also ditch the *subfs* automation.

Now you might say that users who want to delve into the depths of the distro should look to an alternative like Gentoo, which is pre-destined for the do-it-yourself approach. However, even geeks can be lazy. After all, it is an unbelievably practical thing to be able to load software off the Suse DVD when your Internet connection happens to be down.

The *twm* Window Manager

The *twm* window manager was designed for low-powered computers, as was

fvwm2. Both can be a genuine alternative to waiting for KDE on an older machine with low memory, or if you simply prefer something less run-of-the-mill in the line of GUI environments. Normally, the **display manager** will show you a list of the installed alternative desktops when you log on. In the case of the Suse 9.1 installation routine, a single checkmark decides whether the GUI-based login screen appears at all, or if the default user will be logged on automatically, and thus have to make do with KDE.

In the latter case, the simplest way of launching *fvwm* is to select *Logout...* in the KDE menu, and then *Quit session*. When you do so, the display manager pops up, offering to log you on to KDE's own KDM by default, but also showing the alternative desktops installed on the machine in the *Session type* menu. The next time you log on (and this includes automatic login on rebooting), the system will default to the window manager you selected.

Users who tend to change window managers regularly are right in thinking that this is a roundabout approach. The

Fortunately, you can do something about it, and as a positive side effect, ditching that excess weight means more security and stability.

Drives are automatically mounted and removed from the **filesystem tree**, the system takes care of setting up your hardware, at least for non-exotic components, and KDE is so much better than the Windows desktop. Theoretically, Suse 9.1 could advertise itself as being “as easy to use as Windows”.

GLOSSARY

Filesystem tree: Files and directories that reside on a Linux computer are accessible via a root directory (/ on the root partition). If you trace the paths from the root to the individual files, the structure resembles an upside-down tree where directories with multiple subdirectories are the “branches”, and files (or empty directories) are the “leaves”. Each Linux filesystem has a tree structure of this kind that can be mounted into the system tree that starts at the root (/) level. After mounting a filesystem, the files stored in it become accessible.

subfs/submount: Submount, the name of a software tool that mounts removable media automatically when the mount point is accessed. It also takes care of unmounting unused media. To allow this to happen, the Linux kernel must have the “subfs” module loaded, and the “submount” daemon must be running.

Display manager: A program that runs the X server (and thus provides a GUI-based desktop), linking the X server to the screen and providing a graphical login screen.

Kernel module: The Linux operating system proper, or kernel, has functions that govern access to a machine's hardware, for example. Your individual system configuration needs specific functions – drivers for a special sound card for example (and not all the drivers for any sound cards that Linux supports). This prompted the developers to offload functionality into modules that can be loaded at runtime, instead of producing an enormous bloated kernel.

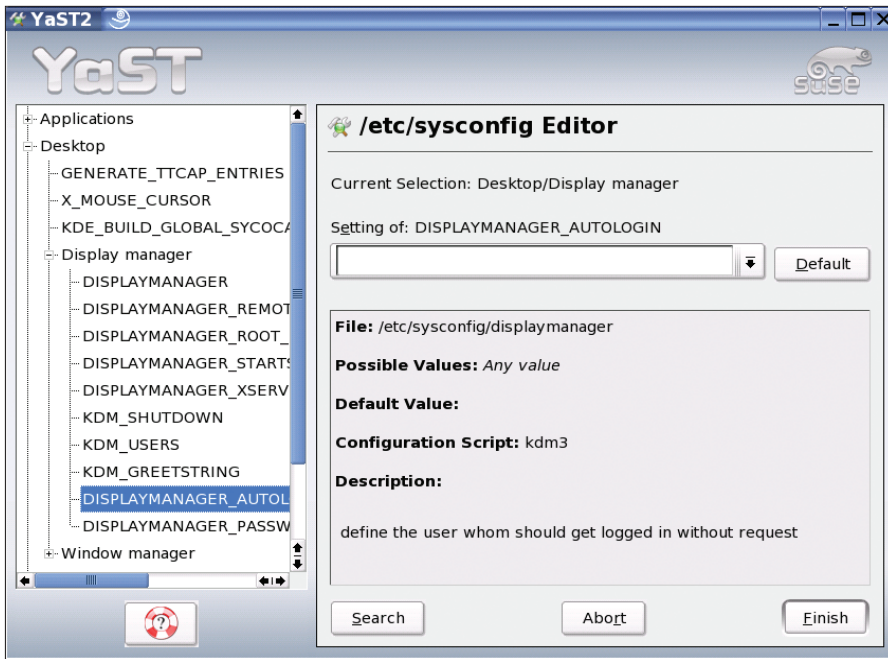


Figure 1: The blank entry in the list of values for the “DISPLAYMANAGER_AUTOLOGIN” variable disables autologin.

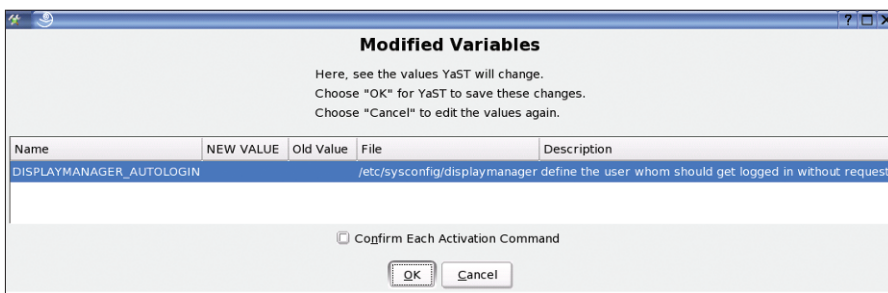


Figure 2: Suse 9.1 just making sure that you really want those sysconfig file changes.

best approach is to tell Suse to launch the display manager whenever you reboot your machine. This approach is also preferable for security reasons, as it means that a password is required to access the default user’s data.

To do this, you need to select *System | YaST* in the KDE menu to launch YaST2. Enter the *root* password, to prove that you have the privileges required to modify the system, and the YaST2 window should appear. This window contains the configuration dialog.

Select *Editor for /etc/sysconfig/ files* in the *System* section. Then modify the *Desktop | Display manager | DISPLAYMANAGER_AUTOLOGIN* entry so that a blank entry in the dropdown list on the right of the window is selected instead of the username (see Figure 1). Then click *Finish* and confirm the prompt in the dialog shown in Figure 2 to tell Suse to ask you for your password on re-booting,

and to display a selection of window managers.

As a side note, let me just mention that you can use the same *sysconfig* editor to select an alternative display manager. If you do not like KDM, you can opt for the Gnome alternative, GDM, WDM (the variant for fans of WindowMaker), or the classic XDM in *Desktop | Display manager | DISPLAYMANAGER*. The *console* list entry prevents a GUI-based desktop from running automatically. If you opt for GDM or WDM, you will need to install the appropriate package before rebooting your machine. If you fail to do so, the system will default to XDM instead.

New Is Not Always Better

One of the biggest hurdles for Linux newbies is the fact that they need to mount removable media like CD or DVD ROMs, or USB sticks, prior to access, and to unmount them when finished. Suse 9.1 removes this problem by automatically mounting *submount* (available from [2]) media, on access, and unmounting them again when you press the eject button on the CD drive. *submount* requires the *subfs* kernel module, and replaces the somewhat ancient *automount* [3].

As the software is under active development, it is hardly surprising that bugs occur. On some machines you might need to insert USB sticks a few times to get them to work [1]. My own test system failed to mount an external USB hard disk cleanly (see Figure 3), but detected a USB stick four times to compensate!

Those of you with enough Linux experience to mount drives manually will probably give up on the automount feature at that point. The question is how can you get rid of *subfs*? The distributor has a howto at [4] for doing just that. In the case of non-USB devices, you simply edit the entries in */etc/fstab*. For example, on my lab machine, I changed this line:

```
/dev/dvdrwrecorder /media
/dvdrwrecorder subfs
fs=cdfss,ro,
procluid,nosuid,nodev,exec,
iocharset=utf8 0 0
```

to the following:

```
/dev/dvdrwrecorder /media
/dvdrwrecorder auto ro,
noauto,user,exec,
iocharset=utf8 0 0
```

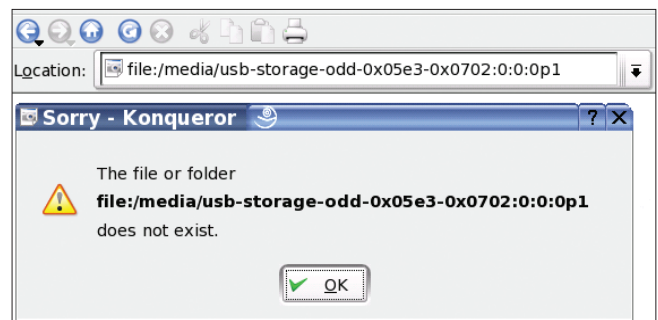


Figure 3: “submount” has enormous difficulty with USB devices.

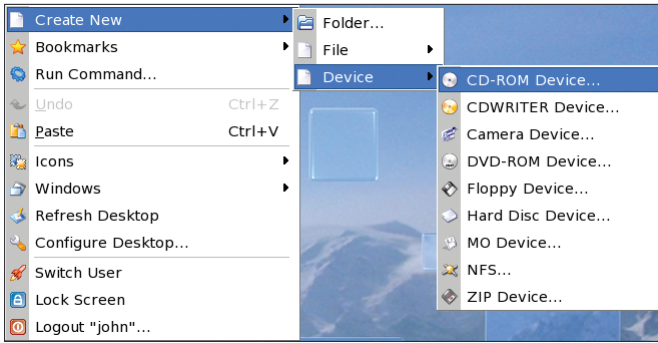


Figure 4: If you do not have “subfs”, you can at least have neat desktop icons.

This tells Linux that non-privileged users (*user*) are allowed to mount DVDs in `/media/dvdrecorder` via the DVD recorder represented by the `/dev/dvdrecorder` device file. Box 1 explains the other mount options in detail. Now, if you insert a CD or DVD into the drive after rebooting, nothing will happen at first, as you need to manually mount the medium. The following command does just that

```
mount /media/dvdrecorder
```

or – if you continue to use KDE – you can simply click the special KDE icons. To create an icon, right click on the KDE

desktop. In the dropdown menu that then appears, select *Create new | Device | DVD ROM drive...* (see Figure 4).

This opens a dialog where you can define the device icon characteristics (see Figure 5). Simply

select the appropriate `/etc/fstab` entry in the *Device* tab, and then click on the OK button to confirm.

Clicking on a device icon on the desktop (see Figure 6) will provide read-only access to the DVD drive. The icon changes to indicate that the device is mounted (see Figure 7). To unmount the DVD, simply right-click the device icon (see Figure 8), and select *Unmount*. To unmount and eject the DVD, select *Action | Eject*. Note that you will be unable to unmount the DVD while another program is accessing it!

Mounting a USB device without *subfs* takes a bit more effort, as they use hotplug scripts for mounting and

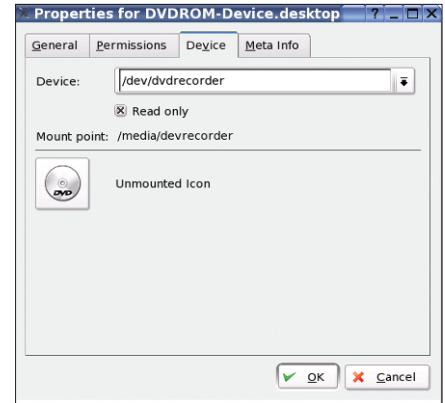


Figure 5: You need to get the entries in this tab right, to be able to click on the desktop icon to automatically access the right device.

unmounting, rather than `/etc/fstab` entries. After all, how is Linux to know that you want to use a USB stick. To disable automatic detection, you need to modify `/etc/hotplug/hotplug.subfs.functions`. Surf to [4] for a modified file, which you can download and use to overwrite the original file, assuming you have *root* privileges.

Note that Suse security updates for the *hotplug* package will return this file to its original state, meaning that you have to modify it once again.

Box 1: `/etc/fstab` Mount Options

The `/etc/fstab` (“File System Table”) file stores information on **filesystems**, which can be a permanent or temporary part of the filesystem tree on the machine. The file has one line per filesystem. The lines contain various tab, or space separated fields.

The first entry specifies the device file used to access the file system. The first primary partition on the master IDE drive attached to the first IDE bus is `/dev/hda1` for example. In a similar way, `/dev/sda1` is the primary partition on the first SCSI hard disk. As USB mass storage devices, such as memory sticks, are handled like SCSI drives, this is the right device for USB sticks.

The second field specifies the mount point in the filesystem tree, that is a directory that needs to exist prior to mounting. Only *root* is allowed to use the `mount` command to access media from arbitrary mount points. Normal users need to be assigned predefined mount points. Newer Suse distributions use a directory below `/media/` to mount removable drives. Other Linux distributions use the `/mnt/` directory, although there is nothing to stop you using an arbitrary

directory. Suse follows the Windows naming conventions and mounts Windows partitions below `/windows/C`, where *C* reflects the drive letter of the partition on Windows.

The third field defines the filesystem type. Microsoft has used NTFS more or less exclusively since introducing Windows XP; older Windows systems and new USB sticks will use VFAT. In the colorful world of Linux, *reiserfs*, *ext2*, *ext3*, *xfs*, or even *jfs* are common. CD and DVD-ROMs have an ISO9660 compatible file system. The `fstab` manpage has a complete list of supported filesystem types. You can type `man 5 fstab` to view the list. If you want to use *submount*, the correct entry in the `fstab` column is *subfs*. If you want to prevent *subfs* from guessing the file system, and leave this up to Linux instead, you should opt for *auto*.

The fourth column has comma-separated values that typically depend on the filesystem type. `man 8 mount` tells you the options. For our previous example of a DVD recorder, these are `ro,noauto,user,exec,ioccharset=utf8`. Note that spaces are between the options

are not allowed – `fstab` uses spaces as separators. The first option, *ro* (“read-only”), prevents attempts to write to mounted DVDs. *noauto* prevents Linux from automounting the filesystem on booting – after all, there is no way of knowing if there will be a DVD in the drive at this point.

user allows non-privileged users to mount the filesystem at the specified mount point – `mount /media/dvdrecorder` in our example. *exec* allows users to run programs from the mounted medium, and *ioccharset=utf8* specifies how non-standard characters and long filenames will be handled.

The fifth column is for the *dump* backup program, and indicates which filesystems you want to back up. This is a fairly uncommon usage. Most systems will either have a zero or no value, indicating that the medium should be ignored for backup purposes. The last column specifies the order in which the Linux filesystem should be checked for consistency on booting. A value of zero in this column prevents a filesystem check for the medium.

Also, Suse's hotplug system uses mount directories with extremely cryptic names below the `/media` directory (the stick on my lab system used a mount point called `/media/usb-storage-1183005103402:0:0:0p1`), for example. To prevent this from happening, you need to change:

```
HOTPLUG_SKIP_EVENTS=""
```

in `/etc/sysconfig/hotplug` to:

```
HOTPLUG_SKIP_EVENTS="block"
```

and re-launch the hotplug system by typing the following:



Figure 6: The device icon bottom right shows that the DVD is not mounted.



Figure 7: Go ahead for read access after mounting the DVD.

```
/etc/init.d/hotplug restart
```

This tells the distribution just to load the kernel module needed to access the memory stick, whenever you insert a USB stick.

What you need now is an entry in `/etc/fstab` to allow "normal" users to mount USB devices. Create a new directory with a mnemonic name below `/media`. This will be your mount point. To do so, and working as `root`, you can simply type the following command:

```
mkdir /media/stick
```

The entry in `fstab` would then be as follows:

```
/dev/sda1 /media/stick auto ↵  
noauto,user,exec 0 0
```

In our test, we had to reboot the computer to get it to handle USB devices in

GLOSSARY

Filesystem: Describes how the data on a medium (e.g. a hard disk partition, a USB stick, a CD, DVD, or floppy) are organized.

Runlevel: Describes the operating modes of a Linux system. Runlevels are distinguished mainly by the services that they launch when booted.

Listing 1: `/boot/grub/menu.lst` (excerpt)

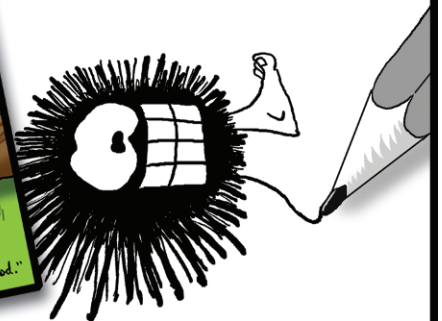
```
01 ###Don't change this comment - YaST2 identifier: Original name:  
    linux###  
02 title Linux  
03     kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 vga=0x317  
    splash=silent desktop resume=/dev/hda5 showopts  
04     initrd (hd0,5)/boot/initrd
```

"There Goes the Neighbourhood"

An Original Full-Colour Print by J.D. "Iliad" Frazer



Available in May of 2004
at the UserFriendly.Org
Online Store



UserFriendly.Org
The Original Geek Humour Comic Strip

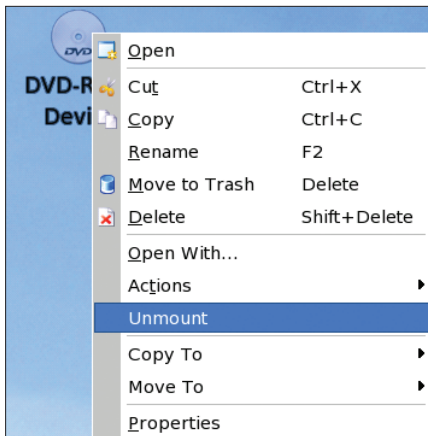


Figure 8: You can use the context menu to unmount media.

the old way. Of course, KDE users can create desktop icons for USB sticks just like for DVD drives. Instead of selecting the *Create new | Device | DVD ROM drive...*, you should use the *Create new | Device | Hard disk...* entry instead.

Putting Suse on a Diet

One of the reasons why Suse needs a lot of memory is the fact that the distro automatically launches a lot of services on booting [5]. Some of them (e.g. the keyboard configuration) are unavoidable, as they make the system usable. However, there are many services that you can discard. To put Suse on a diet, we will need to enlist the help of the YaST2 *Runlevel Editor* running in expert

mode (see Figure 9). Check out the YaST2 *System* menu for this item. As we will be messing around with system settings, it makes sense to make notes of any changes so that you can undo them if the worst comes to the worst.

The expert user interface mainly comprises a table and a list of checkboxes. The first column in the table contains the name of the service, the second tells you if the service is running. These columns are followed by more, telling you the **Runlevels** at which the service is launched automatically, and finally a short description.

If you only use the default runlevel, 5, which displays a GUI-based login, or logs you in automatically when the boot process has completed, you can restrict your changes to removing checkmarks from checkbox #5. For example, a desktop PC attached to a WLAN, but without a printer, does not need either the *cups* printing subsystem or ISDN. To disable both services select them in the list, and remove the checkmark for runlevel 5. This has a positive side effect, as malevolent hackers can not attack disabled services.

Booting Like in the Good Old Days

If your machine's boot process is so garish that it hurts your eyes, you might like to disable the bootsplash effects. To do

so, set the value for *splash* in */boot/grub/menu.lst* to 0. This value configures the boot menu that the grub bootloader displays when you start up your computer.

To change the value, you can either launch the YaST *Bootloader configuration* module, which again is in *System*, or *Edit* the *available sections* – or if you are *root*, just use your favorite editor. All you need to do is replace the word *silent* in line three of Listing 1 with a 0 in the */boot/grub/menu.lst* file, so that the line reads *splash = 0*. The next time you boot, you will be treated to boot messages on a black background rather than the graphical splash screens that are usually displayed.

Attentive readers may be wondering why you can not simply use the Runlevel Editor to disable the *splash* boot script. A classic example of a usability problem, but unfortunately, the documentation does not have the answer. ■

INFO

- [1] Suse 9.1 Professional reviewed: John Southern, "New Core", Linux Magazine, Issue 44 July 2004 p38
- [2] Submount: <http://submount.sourceforge.net/>
- [3] Automount mini howto: <http://www.faqs.org/docs/Linux-mini/Automount.html>
- [4] Classic mount vs. subfs: http://portal.suse.com/sdb/en/2004/05/hmeyer_g1_revert_from_subfs.html
- [5] The Linux boot process: Marc André Selig, "Ready – Steady – Go!", Linux Magazine, Issue 27, February 2003, p48 <http://www.linux-magazine.com/issue/27/Initialization.pdf>

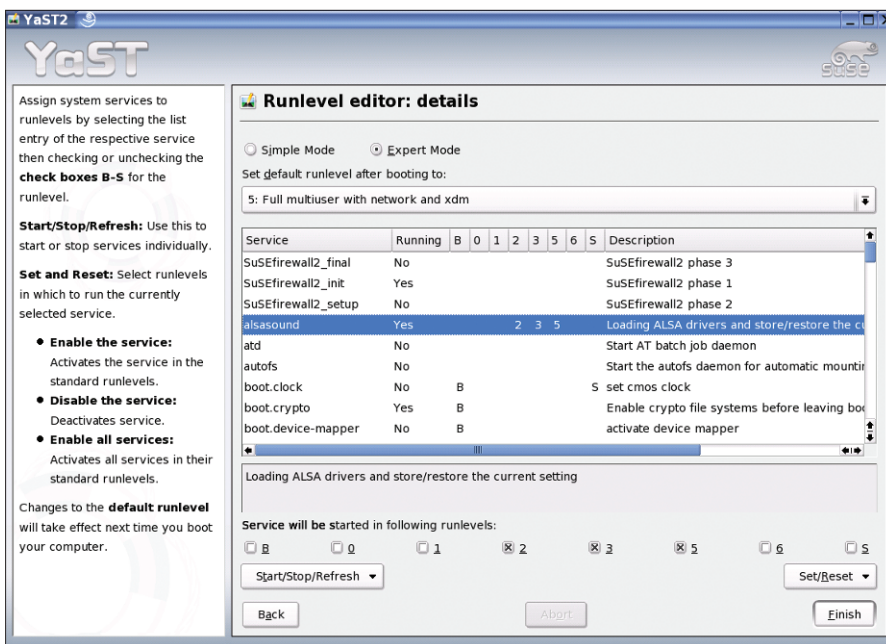


Figure 9: Check the "Expert Mode" checkbox at the top of the screen for detailed modifications to the services running at each runlevel.

THE AUTHOR

Hagen Höpfner has a higher degree in Computer Science and is a member of scientific staff at the Otto-von-Guericke University of Magdeburg, Germany. Between time spent tinkering with his collection of Linux computers, Hagen likes to direct his creative energy into song-writing, composing lyrics, and playing the guitar for a rock band called "Gute Frage!" (translates "Good Question").

