- 18a - X Server Konfigurieren (see "X-Server mit sax konfigurieren" seite unten)

  Administration des Systems -> XFree86[tm] konfigurieren
  SaX2, XF86Setup, xf86config (Textmodus)

  Maus Konfiguration:
  Treiber - Protokoll PS/2 oder IMPS/2 Gereätedatei /dev/psaux

  - für eine Logitech WheelMouse,
    - Datei **/etc/XF86Config** editieren:
      in Section »Pointer« folgende Eintragen zufügen:

```
Section "Pointer"
   Protocol              "IMPS/2"
   Device                "/dev/psaux"
   SampleRate            60
   BaudRate              1200
   Buttons               5
   ZAxisMapping          4 5
EndSection
```

    - Datei **~/.xinitrc** editieren:
      vor **exec $WINOWMANAGER** folgende Zeile zufüfen:
      imwheel @

  - für eine Microsoft WheelMouse mit Xserver 4.0,
    - Datei **/etc/X11/XF86Config** editieren:
      in Section »Pointer« folgende Eintragen zufügen:

```
Section "InputDevice"
  Driver        "mouse"
  Identifier    "Mouse[1]"
  Option        "Device"              "/dev/mouse"
  Option        "InputFashion"        "Mouse"
  Option        "Emulate3Buttons"     "on"
  Option        "Emulate3Timeout"     "50"
  OPtion        "ZAxisMapping"        "4 5"
  Option        "Name"                "WHEEL MOUSE"
  Option        "Protocol"            "IntelliMouse"
  Option        "Vendor"              "MICROSOFT"
EndSection
```

    - Datei ~/.xinitrc editieren:
      vor exec $WINOWMANAGER folgende Zeile zufüfen:
      imwheel @

  - für eine Maus mit nur zwei Tasten
    /etc/XF86Config editieren:
    in Section »Pointer« folgende Eintragen zufügen:

```
Emulate3Buttons
Emulate3Timeout   50
```

    oder in sax:
      Optionen - 3 Tasten Emulieren
      (Erweitert - Eigenschaften - 3 Tasten Emulationsgeschwindigkeit = 50)

- 18b - Das X Windows starten
  startx

X-Window Server

- **X-Server is Network based**
  - Draw Diagram of X-Server to X-Programs relation via internal Network

- **Difference between the X-Server, Windows managers, Desktops**

- **Seting-up the X-Server**
  - `xf86config` or `xf86config3` - Console only for x-servers version 3
  - `xf86config4` - Console only for x-servers version 4
  - `kvidtune` - X-Windows from packages on CD
  - `xcnf` - X-Windows from packages on CD
  - `sax2` (SuSE only) - Graphics oriented configurator
  - `export DISPLAY=:0 ; X &` - Fine adjusts screen size & position
    - `tixwish /var/X11R6/sax/bin/xfine.tcl`
    - `xterm -geometry 1x1+0+0 &`
    - `xterm -geometry 1x1-0+0 &`
    - `xterm -geometry 1x1+0-0 &`
    - `xterm -geometry 1x1-0-0 &`
  - `xset` : changes dynamically some of the settings of X :
    eg. `xset r rate 250 30` sets the keyboard rate
    `xset m 10/20` sets the mouse speed/acceleration

- **Exporting X-Windows programs output to remote Linux**
  - Run the `xhost +` on client machine

  Syntax: `xhost + [`*`Host1`*`] [`*`Host2`*`]..`
  and `xhost - [`*`Host1`*`] [`*`Host2`*`]..`

  Note: works only if the option `-nolisten tcp` is taken out of `/etc/X11/xdm/Xservers` (In SuSE 9.1 only)

  eg. `:0 local /usr/X11R6/bin/X` **`-nolisten tcp`** `-br vt7`
  - Use `telnet` or `ssh` from client to connect to server
  - In `telnet` or `ssh` run the command:
    `export DISPLAY=<`*`clients host address`*`>:0`
  - Many X-program accept the following `-display` option as well

  e.g. `xeyes -display <`*`clients host address`*`>:0`

- **Exporting local Mouse and Keyboard to remote X-Server**
  - Install the package **x2x** from CD
  - Run the `xhost +` on remote machine
  - Run the command:
    `x2x -east -to <`*`Remote IP`*`>:0 &`

- **Running X-Win programs output to remote Windows95/NT**
  - Install `OMNIX` software on a Windows Machine and run it
  - Start a telnet on Windows and logon to Linux Machine as a user
  - In telnet run the command:
    `export DISPLAY=<`*`clients host address`*`>:0`
  - Start any x program needed followed by the **`&`** e.g. `xeyes &`

- **Keystrokes to switch displays:**
  - To start extra X servers : `startx -- :`*`xx`* **(*`xx`* = `0` to `63`)**
  - eg. `startx -- :1`

    Ctrl-Alt-F1...F6 Go to ASCII virtual consoles

Ctrl-Alt-F7	Go to first X-Server Display
Ctrl-Alt-F8	Go to second X-Server Display
	(F9-third, F11-fourth, F12 - fifth X-Servers started)
Ctrl-Alt-Backspace	Kills current X-Server

- '`startx`' **command description.**
  - `startx` starts */usr/X11R6/bin/xinit* program
  - `xinit` uses */home/<user>/.xinitrc* script to setup X-Environment

- **Running multiple instances of X-Servers on the same PC:**

  Use the `startx` for the first instance of X-Server
  - Go to another virtual consolle (e.g. <Ctrl-Alt-F2>)
  - Login as another user if needed
  - Type the command
    ```
    startx -- :1
    ```
  - It will then be reachable through <Ctrl-Alt-F8>
  - The first X-Sertver is still reachable through <Ctrl-Alt-F7>

  To get yet another X-Server up:
  - Go to another virtual consolle (e.g. <Ctrl-Alt-F3>)
  - Login as another user if needed
  - Type the command
    ```
    startx -- :2
    ```
  - It will then be reachable through <Ctrl-Alt-F9> and so on

  The number given after the : is irrelevant to the Key combination to access it. The key combinations are consecutively assigned as new X-Serves are starting.

  Note: The <Ctrl-Alt-F10> is (in SuSE distribution) reserved for system messages

So for example:

Ctrl-Alt-F7 - First X-Server	(`startx`)
Ctrl-Alt-F8 - Second X-Server	(`startx -- :2`)
Ctrl-Alt-F9 - Third X-Server	(`startx -- :6`)
Ctrl-Alt-F11 - Fourth X-Server	(`startx -- :8`)
Ctrl-Alt-F12 - Fifth X-Server	(`startx -- :9`)

- To start another X-Server only (no KDE or Gnome)
  `X :1` Starts a new X-Server only at display port 1

- **Extra Info and commands:**

  - To get the **Geometry of a window**, run the program **`xwininfo`** from `xterm`. Point and click on the window desired. The listing of *x-win* parameters for this program will be displayed in the `xterm` that started the **`xwininfo`**. The geometry values are at the very end of the listing.
  - To get Settings info on X-Server: `xset -q`
  - To set the **`Standby`** , **`Suspend`** and **`ComputerOff`** values

`xset dpms` *`StandbyValue SuspendValue ComputerOffValue`*
`eg. xset dpms 540 900 900`
*(**standby** after 9 minutes, **suspend** and **Off** after 15 miutes)*

- To get a visual listing of different colors and their names use the program:
  `xcolors`

- To fix the **ugly *Helvetica font size: 11* in StarOffice 5.2** do the following:
  - Edit the file `/etc/X11/XF86Config`
  - Delete all the parameters `:unscaled` from the list of fonts
    (Find and replace all `:unscaled` for *nothing*)

- **To get the full Hardware name of a Graphic Card:**
  `SuperProbe` (Not always present on distribution)

- **Files Involved with X-Server:**

`/etc/XF86Config`	- X Windows server config file for Version 3.3.x
`/etc/X11/XF86Config`	- X Windows server config file for version 4.0.x
`xf86config`	- X Windows server configuration program
`XF86Setup`	- Graphic X Windows server configuration program for Version 3.3.x
`sax`	- Graphic X Windows server configuration program for Version 3.3.x
`sax2`	- Graphic X Windows server configuration program for Version 4.0.x
`/usr/X11R6/lib/X11/app-defaults`	- System wide X-Windows defaults param.
`/home/~/.Xresources`	- Link to .Xdefaults for user specific Xwindows parameters. They override the system wide ones.
`/usr/X11R6/bin/X`	- Xserver itself

- **Extra commands and features of X-Server**
  - To send a message to another X-server (local or remote):
    `xmessage -display` *`IPNr`*`:0 -center "This is my message"`
    `kdialog -display` *`IPNr`*`:0 --msgbox "hallo"`

  - To control the mouse and keyboard on remote X-Server:

    `x2x -to` *`servername`*`:0 -`*`Direction`* `(west, east, north, south)`

    (moving the mouse out of local host in direction as above controls the remote mouse and keyboard)

  - **Control and info of X-Server**
    - To get Settings info on X-Server: `xset -q`

Some Components of KDE 3

`X :1 &`	starts the X Server only! on Port `:1`
`Ctrl-Alt-F1`	to go back to the virtual console 1
`export DISPLAY=:1`	Tells following programs where the X-Server is.
`xeyes &`	eyes appear on grey screen
`xteddy &`	teddy appears. He is movable
`xterm -e mc &`	MC works but no window title or movable also hiding `xeyes`
`twm &`	One type of window Manager (Primitive!!!)
`killall twm`	Back without frame and titlebar
`kwin &`	KDE Main window manager
`kdesktop &`	Background and icons
`kicker &`	KDE Panel with climb-up menus

Start of the whole KDE.

`startkde`	Start the whole KDE but doesn't start the XServer
`startx`	Starts the whole KDE including the XServer

Extras:

`panel &`	Starts the GNOME panel
`tasklist_applet`	Starts the GNOME tasklist in panel

To end X session:

<Ctrl> <Alt> Backspace - Kills x-server and all x-programs

Windows X-Servers

OMNIX and plink.exe

The combination of OMNIX X-Server for Windows and `plink.exe` from Putty set can be quite good for executing X-programs on a Linux server and showing them on the Windows Interface with a simple click of the mouse.

1) Install Omnix as Multiple windows mode.
2) Copy `plink.exe` in `C:\`
3) Create an shortcut as icon on the desktop or in a window using the right mouse and NEW Shortcut.
4) Enter the command in the field (all on one line):

```
C:\plink.exe -ssh -P 22 -pw user1 user1@192.168.70.40 \
      "/usr/X11R6/bin/xterm -display 192.168.70.60:0"
```

   Here the Linux host is at `192.168.70.40`
   it has a user called `user1`
   password `user1`
   The local windows host is at `192.168.70.60`

5) Start the Omnix server (it will only be shown in the task bar)
6) Double-click on the new shortcut icon to get the DOS window with a question
7) Press **y** (that's the only time you'll ever need to do that for this connection)
8) The `xterm` should appear on the desktop.
9) You can now close the DOS window.

XDM/KDM Configuration

- Some extra X-server info:

`/usr/X11R6/bin/X`	X-Server Binary Program
`X -ac`	Starting X-Server with NO Access controls
`xhost +`	Disable the Access Control of already started X-Server
`xhost + remotehost`	Enables the remotehost to access the local X-Server
UDP port 177	Listening Protocol and port of KDM(listed in `kdmrc`)

- **Methods of communication with XDM from remote machines(Application Server)**:

Local X-Server Command Remote XDM/KDM

1) XTerminal ------ Direct Query (`X -query remoteXDMserver`)---------->| XDM
<---------- X-Login Prompt ----------|

2) XTerminal ----- Broadcast Query (`X -broadcast`)---------->XDM--XDM--XDM
<---------- X-Login Prompt----------|

3) XTerminal -------Indirect Query (`X -indirect`) ---------- -------->| Remote Host
<---------- List of Network XDM available ----------|
---------- Section of remote XDM---------->|
<---------- X-Login Prompt ---------- Chosen XDM

4) XTerminal <---------- X-Login Prompt ----------| XDM + Clients list
XTerminal <---------- X-Login Prompt ----------| in `Xservers`
XTerminal <---------- X-Login Prompt ----------|

- **Setting-up XDM/KDM**
  Location of XDM config files(SuSE & Debian): `/etc/X11/xdm/`
  Location of KDM config files(debian): `/etc/kde3/kdm/`
  Location of KDM config files(SuSE 9.2 & up):`/etc/opt/kde3/share/config/kdm/`
  and `/opt/kde3/share/config/kdm/`

- Configuration files (same format, purpose and filenames for both XDM and KDM:
  `xdm-config` or `kdmrc`
  `Xaccess` `Xservers`
  `Xresources`

- `xdm-config` or `kdmrc`

  - List of other configuration files
    (Note-SuSE: `kdm` uses all of `xdm` config files except `Xservers`)
  - Access permissions
  - Name of Start scripts to run for x-sessions (different than `xinit` or `startx`)
  - For XDM (`xdm-config`) to enable the listening of XDMCP on the network:
    Remote quieries Port Nr:
    `DisplayManager.requestPort: 0` = Blocked
    `! DisplayManager.requestPort: 0` = Remote queries Allowed
  - For KDM (`kdmrc`) to enable the listening of XDMCP on the network:
    (In SuSE 9.2/9.3 Both `kdmrc` files must be changed)
    `/etc/opt/kde3/share/config/kdm/kdmrc`
    and `/opt/kde3/share/config/kdm/kdmrc`

```
[Xdmcp]
  Enable=true
```

  - To make sure user must give root password to shutdown or reboot the system.

```
[X-*-Core]
  AllowShutdown=Root
```

- Xaccess
  List of machines which can connect to local XDM. Also where we turn the CHOOSER ON for indirect requests.

- Xservers
  List of machines (running X-Server) which should receive the X-Login Prompt automatically. Minimum setting is `localhost`, if this auto-connecting method is not used for remote X-Servers.
  Example:(`vt7` is given to avoid conflicting with the local `mingettys`)

```
# First the local host
:0 local /usr/bin/X11/X vt7
# To disable the tcp listening add the -nolisten tcp to the line above
# eg. :0 local /usr/bin/X11/X -nolisten tcp vt7
# Then the remote hosts
emma:0 foreign
alex:0 foreign
```

  In this example the XDM will present an X-Login prompt on to the local X-server port `:0` and send an X-Login to the 2 hosts `emma` and `alex` port `:0`. The access permissions must be given on both hosts `emma` and `alex` to alow XDM to connect to their X-servers. (`X -ca` or `xhost +` *`xdmservername`*)

- Set-up method 1: Host(`mike`) queries the KDM server(`appserv`) for an X-Login:

  - On KDM server(`appserv`):
    - In `/etc/opt/kde3/share/config/kdm/kdmrc`:

```
[Xdmcp]
  Enable=true
[X-*-Core]
  AllowShutdown=Root
```

    - Content of `/etc/X11/xdm/Xaccess`

```
*           #any host can get a login window
```

  - On Client host(`mike`):
    - Start X-server as follows:

```
X -query appserv :0
```

- Set-up method 2: Host(`mike`) queries all network KDM servers for an X-Login: (Broadcast) In this case the only Application Server existing (`appserv`) will respond.

  - On KDM server(`appserv`):
    - In `/etc/opt/kde3/share/config/kdm/kdmrc`:

```
[Xdmcp]
  Enable=true
[X-*-Core]
  AllowShutdown=Root
```

    - In `/etc/X11/xdm/Xaccess`

```
*           #any host can get a login window
```

  - On Client host(`mike`):

- Start X-server as follows:
```
X -broadcast :0
```

- Set-up method 3: Host(mike) queries Chooser-Host (appserv) Host for KDM List:
  Note: All KDM servers must be ready to receive queries([Xdmcp] Enable=true)

  - On KDM server(appserv):
    - In /etc/opt/kde3/share/config/kdm/kdmrc:
```
[Xdmcp]
  Enable=true
[X-*-Core]
  AllowShutdown=Root
```
    - In /etc/X11/xdm/Xaccess
```
*  CHOOSER BROADCAST    #any host can get a chooser
```

  - On Client host(mike):
    - Start X-server as follows:
```
X -indirect appserv :0
```

- Set-up method 4: XDM/KDM sends X-Logins automaticaly to hosts: andre & marie.

  - In KDM Server Host (Central application server)
    - In /etc/opt/kde3/share/config/kdm/kdmrc:
```
[Xdmcp]
  Enable=true
[X-*-Core]
  AllowShutdown=Root
```

    - In /etc/opt/kde3/share/config/kdm/Xservers:
```
# First the local host
:0 local /usr/bin/X11/X vt7
#
# Then the remote hosts
andre:0 foreign
marie:0 foreign
```

  - In hosts: andre & marie
    - Start the X-server with the command:
```
X -ac :0
```
    - Wait for the X-Login and login.

  Note: The Application Server Clients (andre & marie) can start and stop their X-server as they want and the KDM will send its X-Login when client is ready.

XTerminals using the xdm/kdm Display Managers:

1) Activate the XDMCP (XDM Control Protocol):
   - Edit the file /etc/X11/xdm/xdm-config
     add a '!' at the beginning of the following line (normally the last line):
```
!DisplayManager.requestPort:    0
```

Note 1: The display port number (:2 etc) can be chosen at will from the client as long as

the same port is not chosen multiple time sin the same client host. This number can also be eg. `:2.0` which means the first graphic card used (`0`). Since it's mostly the case we only use eg. `:2` and it's enough.

Note 2: For these configuration files changes to take effect `kdm`/`xdm` needs to be restarted.

- **X Server**
  X-Server offers an empty display where programs the support the X-Protocol will be displayed and controlled via the mouse and keyboard where the display runs.
  The X-Server takes control of the local Graphic card,monitor, mouse and keyboard and possibly other devices like joystick, graphic tablet etc.
  The X-Server is a network service for local or remote clients(X-Programs).
  The X-Server is been develloped for many hardware environments. Most of them are proprietary and one of them is Free: X-Free-86(XF86). It is the one explained below.

`/etc/X11/XF86Config`	Main XF86 Configuration file. It is the first looked for. Before FHS it was often at `/etc/XF86Config` If the user is root and he starts the XF86 manually then the `~/XF86Config` file will be first searched for.

  Note: Because XFree86 organization has changed it's licence, XORG.org is been created to continue to provide a GPL version of the X-server. Since then most distributions have changed from XFREE86 X-server to XORG X-server.
  Therefore the configuration filename is changed from XF86Config to xorg.conf

  - **Search Path of XF86Config file.**

    -When X is started as a normal user:
    ```
    /etc/X11/$XF86CONFIG
    /usr/X11R6/etc/X11/$XF86CONFIG
    ```
    Then Common search path

    -When X is started as the `root` user.
    ```
    $XF86CONFIG
    /etc/X11/$XF86CONFIG
    /usr/X11R6/etc/X11/$XF86CONFIG
    $HOME/XF86Config
    ```
    Then Common search path

    Common search path:
    ```
    /etc/X11/XF86Config-4
    /etc/X11/XF86Config
    /etc/XF86Config
    /usr/X11R6/etc/X11/XF86Config.<hostname>
    /usr/X11R6/etc/X11/XF86Config-4
    /usr/X11R6/etc/X11/XF86Config
    /usr/X11R6/lib/X11/XF86Config.<hostname>
    /usr/X11R6/lib/X11/XF86Config-4
    /usr/X11R6/lib/X11/XF86Config
    ```

- **Note:** In the above 2 paths ***/X-Rootdir*** is normally `/usr/X11R6/`
  Depending on distributions the configuration files of X-Server Version 3 and Version 4 are located in different locations. Often used locations and names:

`/etc/XF86Config`	Version 3
`/etc/X11/XF86Config`	Verison 4

  or

`/etc/X11/XF86Config`	Version 3
`/etc/X11/XF86Config-4`	Verison 4

- **XF86 Configuration programs** (Helpers to write the `XF86Config` file)

`xf86config` First text-based configuration program.
Provided and suported by the XF86 development team.
Belongs to standard X-Server packages.

`XF86Setup` Graphic-based (640x480-VGA 16 colors) configuration program.
Also provided and supported by XF86 development team.

`xf86cfg` Graphic-based configuration program. More complex and more for advanced administrators. Provides the possibility of dynamically try some of the settings by pressing on 'Apply' button.
Provides auto-detecting of graphic cards.
Also provided and supported by XF86 development team.

SAX & SAX2 SuSE graphic-based configuration programs.
SAX is for XF86 Version 3 and SAX2 for XF86 Version 4.
Provides auto-detecting of graphic cards.

`Xconfigurator`
RedHat text-based configuration programs. Similar but improved version of the `xf86config`. It does auto-detection of graphic cards. Works in inteactive mode or in automatic-install mode.

`dexconf` Background program run by Debian system installation program.
No user startable program. To reconfigure the X-Server execute:
`dpkg-reconfigure xserver-xfree86`

All of the above configuration programs do 2 things:
- Configuration of the `XF86Config` file.
- Creation of a symbolic link to the configured X-Server

- **Configuration of the `XF86Config` file** :
  Sections of XF86Config file:

`ServerFlags`	Server flags
`Module`	Dynamic module loading
`InputDevice`	Input device description
`Device`	Graphics device description
`VideoAdaptor`	Xv video adapter description

`Monitor`	Monitor description
`Modes`	Video modes descriptions
`Screen`	Screen configuration
`ServerLayout`	Overall layout
`DRI`	DRI-specific configuration
`Vendor`	Vendor-specific configuration

Note: Although the next 2 sections are recognized by version 4 it is recommended to use InputDevice section instead.

`Keyboard`	Keyboard configuration(Version 3)
`Pointer`	Mouse configuration(Version 3)

- **Creation of a symbolic link to the configured X-Server**

  For Version 3

  `/usr/X11R6/bin/X ==> /var/X11R6/bin/X ==> /usr/X11R6/bin/XF86_`*`Servertype`*

  For Version 4

  `/usr/X11R6/bin/X ==> /var/X11R6/bin/XFree86`

- **Starting the Xserver and Windowmanager:**

  `startx` (script)

- **Fine Tuning the monitor settings:**

  - Manual with monitor's buttons
  - Via the `xvidtune` program

- **X11 Fonts and Fonts server:**

  Fonts are listed in XF86Config through the Keyword: `FontPath`
  eg.

```
FontPath "/usr/X11R6/lib/X11/fonts/TrueType"
FontPath "/usr/X11R6/lib/X11/fonts/75dpi:unscaled"
```

  Font servers can also be listed but MUST be listed as first in the list:

`FontPath "unix/:7100"`	Font server on local Unix socket
`FontPath "unix/:-1"`	Font server on local Unix socket

- **`xset`: Temporarily Changing the X-Server's `FontPath` settings as it runs:**

  Example of adding and deleting FontPaths(non permanent).

```
  xset  +fp /usr/X11R6/lib/X11/fonts/TrueType   Adds a FontPath
or xset  fp+/usr/X11R6/lib/X11/fonts/TrueType

  xset  -fp /usr/X11R6/lib/X11/fonts/TrueType   Deletes a FontPath
or xset  fp-/usr/X11R6/lib/X11/fonts/TrueType
```

- **Setting-up a Font server:XFS**

  XFS is the standard Font Server which listens for requests on port `7100`(Standard).

  - Settings of client's `XF86Config` configuration file:

```
FontPath "unix/:7100"                Local Font server on Unix socket
FontPath "unix/:-1"                  Local Font server on Unix socket
FontPath "tcp/myserver.fd.com:7100"  Remote font server
```

- XFS Configuration file:`/etc/X11/fs/config` or `/etc/X11/xfs.conf`
- Starting the font server as Daemon:
  `xfs -config /etc/X11/fs/config -daemon`

Example of XFS configuration file:

```
no-listen = tcp
port = 7100
client-limit = 10
clone-self = on
use-syslog = on
deferglyphs = 16

catalogue = /usr/X11R6/lib/X11/fonts/misc:unscaled,
            /usr/X11R6/lib/X11/fonts/75dpi:unscaled,
            /usr/X11R6/lib/X11/fonts/100dpi:unscaled,
            /usr/X11R6/lib/X11/fonts/japanese:unscaled,
            /usr/X11R6/lib/X11/fonts/baekmuk:unscaled,
            /usr/X11R6/lib/X11/fonts/Type1,
            /usr/X11R6/lib/X11/fonts/URW,
            /usr/X11R6/lib/X11/fonts/Speedo,
            /usr/X11R6/lib/X11/fonts/CID,
            /usr/X11R6/lib/X11/fonts/PEX,
            /usr/X11R6/lib/X11/fonts/cyrillic,
            /usr/X11R6/lib/X11/fonts/latin2/misc,
            /usr/X11R6/lib/X11/fonts/latin2/75dpi,
            /usr/X11R6/lib/X11/fonts/latin2/100dpi,
            /usr/X11R6/lib/X11/fonts/latin2/Type1,
            /usr/X11R6/lib/X11/fonts/latin7/75dpi,
            /usr/X11R6/lib/X11/fonts/kwintv,
            /usr/X11R6/lib/X11/fonts/truetype,
            /usr/X11R6/lib/X11/fonts/uni,
            /usr/X11R6/lib/X11/fonts/ucs/misc,
            /usr/X11R6/lib/X11/fonts/ucs/75dpi,
            /usr/X11R6/lib/X11/fonts/ucs/100dpi,
            /usr/X11R6/lib/X11/fonts/hellas/misc,
            /usr/X11R6/lib/X11/fonts/hellas/75dpi,
            /usr/X11R6/lib/X11/fonts/hellas/100dpi,
            /usr/X11R6/lib/X11/fonts/hellas/Type1

# in decipoints
default-point-size = 120
default-resolutions = 75,75,100,100
# font cache control, specified in KB
cache-hi-mark = 2048
cache-low-mark = 1433
cache-balance = 70
```

- **Fonts names Format.**

```
Author        Weight    Width      Pixels  XRes    Spacing    ISO-Standard
   |             |         |          |       |       |          |
  -b&h-lucida-medium-r-normal-sans-18-180-75-75-p-106-iso8859-1
        |             |           |       |      |      |            |
    Fontname      Attribute     Style   Points YRes Average Options
                  (i or * =Italic)              (1/72 in)  Width
                   (r =roman)
```

- **Installing new fonts:**
  New fonts needs some preparation before they can be used.
  Besides the font files(with extentions `.bdf .snf .pcf`)
  located in the font directories, some extra files needs attention:

  `fonts.dir` Contains the number of fonts available in this dirctory (on first line) and one line per font description. The Format is:
  First line: Number of fonts listed in this file.(eg. 439)
  Rest of file: *`FontFilename Font_Description`*
  eg.

```
439
putbi.pfa -adobe-Utopia-bold-i-normal--0-0-0-0-p-0-adobe-standard
putbi.pfa -adobe-Utopia-bold-i-normal--0-0-0-0-p-0-iso10646-1
putbi.pfa -adobe-Utopia-bold-i-normal--0-0-0-0-p-0-iso8859-1
.........................
```

To create this file the program `mkfontdir` must be run:
Syntax:

`mkfontdir` *`/path/to/font/directory`*

Valid font types: PCF (`.pcf`), SNF (`.snf`) and BDF (`.bdf`)

`fonts.alias`

List entered by hand assigning alias font names to an existing ones
Format: *`fictive_name existing_name`*
eg.

```
fixed        -misc-fixed-medium-r-semicondensed--13-120-75-75-c-60-iso8859-1
variable     -*-helvetica-bold-r-normal-*-*-120-*-*-*-*-iso8859-1
5x7          -misc-fixed-medium-r-normal--7-70-75-75-c-50-iso8859-1
5x8          -misc-fixed-medium-r-normal--8-80-75-75-c-50-iso8859-1
6x9          -misc-fixed-medium-r-normal--9-90-75-75-c-60-iso8859-1
6x10         -misc-fixed-medium-r-normal--10-100-75-75-c-60-iso8859-1
6x12         -misc-fixed-medium-r-semicondensed--12-110-75-75-c-60-iso8859-1
6x13         -misc-fixed-medium-r-semicondensed--13-120-75-75-c-60-iso8859-1
6x13bold     -misc-fixed-bold-r-semicondensed--13-120-75-75-c-60-iso8859-1
7x13         -misc-fixed-medium-r-normal--13-120-75-75-c-70-iso8859-1
7x13bold     -misc-fixed-bold-r-normal--13-120-75-75-c-70-iso8859-1
7x13euro     -misc-fixed-medium-r-normal--13-120-75-75-c-70-iso8859-15
7x13eurobold -misc-fixed-bold-r-normal--13-120-75-75-c-70-iso8859-15
7x14         -misc-fixed-medium-r-normal--14-130-75-75-c-70-iso8859-1
7x14bold     -misc-fixed-bold-r-normal--14-130-75-75-c-70-iso8859-1
8x13         -misc-fixed-medium-r-normal--13-120-75-75-c-80-iso8859-1
8x13bold     -misc-fixed-bold-r-normal--13-120-75-75-c-80-iso8859-1
8x16         -sony-fixed-medium-r-normal--16-120-100-100-c-80-iso8859-1
9x15         -misc-fixed-medium-r-normal--15-140-75-75-c-90-iso8859-1
9x15bold     -misc-fixed-bold-r-normal--15-140-75-75-c-90-iso8859-1
10x20        -misc-fixed-medium-r-normal--20-200-75-75-c-100-iso8859-1
12x24        -sony-fixed-medium-r-normal--24-170-100-100-c-120-iso8859-1
```

`fonts.scale`

List of fonts that are scalable. The format is:
First line: Number of fonts listed in this file.(eg. 439)
Rest of file: *`FontFilename Font_Description`*
eg.

```
439
putbi.pfa -adobe-Utopia-bold-i-normal--0-0-0-0-p-0-adobe-standard
putbi.pfa -adobe-Utopia-bold-i-normal--0-0-0-0-p-0-iso10646-1
putbi.pfa -adobe-Utopia-bold-i-normal--0-0-0-0-p-0-iso8859-1
.................
```

- **Changing configuration of an active X Server:**

  Dynamic settings:

  The X-Server can be dynamically (non permanently) controlled by `xset` command

  Display the xserver settings values

  ```
  xset q
  ```

  Changing the keyboard rate:

  ```
  xset r 30 250
  ```
  (30 char/sec. Delay=250ms)

  Changing the mouse speed parameters

  ```
  xset m 10/20
  ```
  (10 Speed/20 Accelleration)

  Changing the screen saver parameters

Deactivating the screen saver
`xset s off`

Activating the screen sreen saver

`xset s on`

Start the screen sreen saver NOW
`xset s activate` (better put as command in a desktop icon)

- X-Server can also be controlled to provide certain configurations when X-Programs are started using the `.Xresources` file. It is normally in the user's home directory.
  Note: In SuSE `.Xresources` is a symbolic link to `~/.Xdefaults`
  The file format is:

  `X-ProgramName*attribute`: *`value`*
  eg.(commented lines start with a '!')

```
xterm*background:       LightYellow2
xterm.eightBitInput:    true
xterm*multiScroll:      on
xterm*jumpScroll:       on
! xterm*font:   -adobe-courier-bold-r-normal--14-140-75-75-m-90-iso8859-1
xterm*ScrollBar: on
xterm*SaveLines: 2000
! xterm*VisualBell: true
xterm.eightBitOutput:   true
```

These parameters can be overriden by starting the X-Program (`xterm`) with arguments.
eg. `xterm -fn 9x15bold -geometry 100x40+30+40 -bg LightYellow2 \`
`-T "Test_Xterm" -sb -rightbar`

Setup a display manager

- **Key files, terms, and utilities:**
  `/etc/inittab`
  `/etc/X11/xdm/*`
  `/etc/X11/kdm/*`
  `/etc/X11/gdm/*`

- **Methods of starting an X session**
  An X session can be started in 2 ways:
  - **Loging from a virtual terminal**(text based) and then run the script `startx`.
    `startx` in turns starts `xinit`.
    `xinit` configuration file:
    `$HOME/.xinitrc` if found otherwise,
    `/var/X11R6/lib/xinit/xinitrc`

  - **Via an X-Display-Manager(XDM):** The user loging in is done graphically.
    The display manager is started at boot time(runlevel 5) in the background as daemon and provides graphical logins to users.
    Note: For this we need to make sure that the /etc/inittab has 5 as default runlevel:
    `id:5:initdefault:`

  - **Popular display managers:**

XDM	Config files Directory	Description
`xdm`	`/etc/X11/xdm/`	Provided by XFree86
`kdm`	***`kde_rootdir`***`/share/config/kdm/`	Provided by KDE

gdm ***gnome_rootdir***/gdm/ Provided by Gnome

kde_rootdir = Main root directory for kde desktop system
for kde3: /etc/opt/kde3

gnome_rootdir= Main root directory for Gnome desktop system
for Gnome 2: /etc/opt/gnome

- **Properties of the Display Managers:**
  kdm is based on xdm and uses many of its configuration files.
  gdm is totally new devellopped and is therefore more independant from xdm.

- **xdm configuration:**
  xdm is a typical X11 program and offers only a logo, a background and login fields.
  The parameters to change its behaviour are in :
  /etc/X11/xdm/Xresources

```
eg.
xlogin*greeting:   Welcome at CLIENTHOST  <---(replaced automatically by $HOSTNAME)
xlogin*namePrompt:      \040\040\040\040\040\040\040Login:
xlogin*fail:            Login incorrect
xlogin*login.greetFont: *-FAMILY-bold-SLANT-normal--*-140-*-*-*-*-iso8859-1
xlogin*login.promptFont:    *-FAMILY-bold-r-normal--*-120-*-*-*-*-iso8859-1
xlogin*login.Font:        *-FAMILY-medium-r-normal--*-120-*-*-*-*-iso8859-1
xlogin*logoFilename       /xxxxx.xpm
xlogin*borderWidth
xlogin*useShape:               true
xlogin*greetColor:             CadetBlue
xlogin*failColor:              red
xlogin*borderWidth:            0
xlogin*frameWidth:             5
xlogin*innerFramesWidth:       2
xlogin*Foreground:             black
xlogin*Background:             #c0c0c0
xlogin*shdColor:               #828282
xlogin*hiColor:                #e0e0e0
```

  xdm runs a script called /etc/X11/xdm/Xsetup each time it presents a login window. There we can run programs that change the background etc.
  Some examples of programs: xpmroot , xsetbg etc
  eg. /usr/sbin/xpmroot /etc/X11/xdm/background.xpm

  - **kdm configuration**
    kdm works quite similar to xdm and uses a many of its configuration files in:
    /etc/X11/xdm/
    The main kdm configuration file is:
    kde_rootdir/share/config/kdm/**kdmrc**
    kde_rootdir = Main root directory for kde desktop system for kde3:
    normally: /etc/opt/kde3

    The pictures of the users shown in kdm login are(valid formats: .xpm or .png):
    kde_rootdir/share/apps/kdm/pics/users/*username*.png
    The default is default.png

  - **gdm configuration**
    gdm has its own configuration files separate from the xdm/kdm.
    - Main configuration file:

gnome_rootdir/gdm/gdm.conf
gnome_rootdir= Main root directory for Gnome desktop system
for Gnome 2: /etc/opt/gnome

- Method of configuring gdm.conf:
  manual(editor) and (much better) through the config program: gdmconfig

- Other tool for configuring individual user's pictures in gdm login:
  gdmphotosetup

- Note 1: For these configuration files changes to take effect kdm/xdm needs to be restarted.
- Note 2: The display port number (:2 etc) can be chosen at will from the client as long as the same port is not chosen multiple time sin the same client host. This number can also be eg. :2.0 which means the first graphic card used (0). Since it's mostly the case we only use eg. :2 and it's enough.
  - **gdm XDMCP configuration:**
    Use the program gdmconfig ---> Expert sub-menu ---> Activate XDMCP
    or
    - Edit the file ***gnome_rootdir***/gdm/gdm.conf
      gnome_rootdir= Main root directory for Gnome desktop system
      for Gnome 2: /etc/opt/gnome/
    - Enable the Xdmcp:
      [xdmcp]
      Enable=true

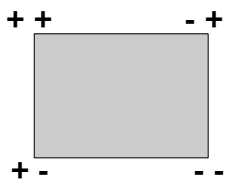
- **Window manager:**
  The window managers allow the applications to be moved, made bigger or smaller, to iconize, presents a window top bar , some of them also a menu system or allow drag-&-drop between applications.

  Window managers names:
  twm, mwm, olwm, fvwm, kwin, windowmaker etc:

  Configuration files of window managers:
  Different for each one but they seem almost all to have the .*xxxx*rc format.
  They are normally in the $HOME directory.
  eg. .mwmrc, .fvwm2rc, .olwmrc etc.

- **Configuration of X Cilents(X programs):**
  Many X Clients will accept many of the following X11 standard parameters: eg.
  xterm -T "Title" -fn 9x15 -display :0 -geometry 100x40+30+40
  - **Positioning and size of window when starting an X Client**: -geometry

    Syntax:
    -geometry <*Hsize*>**x**<*Vsize*><*Hpos*><*Vpos*>
    <*Hsize*> and <*Vsize*> are numbered in characters
    <*Hpos*> '+' is down, '-' is up (in screen pixels)
    eg. **+10** is down 10 pixels
    <*Vpos*> '+' is right, '-' is left (in screen pixels)
    eg. **+10** is right 10 pixels

    E**xamples:**

`-geometry 1x1+0+0`
1 char Horiz, 1char Vert, top left corner (offset 0,0):
`-geometry 5x20-10+30`
5 char Horiz,
20char vert,
top right corner
10 Pixels Horiz.to the left
30 Pixels vertical down

Note: The geometry can also be set for x clients by editing the `~/.Xresources`
eg. `Xterm*geometry: 90x30`

- **Selecting a font for X Clients:** `-fn` *`fontname`*
  List of fixed some short named fonts:
  `7x14 6x10 6x13 8x13 9x15 10x20`
  eg.
  `xterm -fn 10x20`
  or
  `xterm -fn -misc-fixed-medium-r-normal--20-200-75-75-c-100-iso8859-1`

  Note: The fonts can also be set for x clients by editing the `~/.Xresources`
  eg. `Xterm*font: 90x30`

- **`~/.Xresources` or `~/.Xdefaults` file:**
  Note: in SuSE `~/.Xresources` is a symbolic link to `~/.Xdefaults`

  File syntax:
  `ProgramName*Resource`**`: Value`**

  Examples of Xterm settings in `~/.Xresources`

```
xterm*background:       LightYellow2
xterm*Foreground:       Blue
xterm.eightBitInput:    true
xterm*multiScroll:      on
xterm*jumpScroll:       on
xterm*font:   -adobe-courier-bold-r-normal--14-140-75-75-m-90-iso8859-1
xterm*ScrollBar: on
xterm*SaveLines: 2000
xterm*VisualBell: true
xterm.eightBitOutput:   true
Xterm*geometry: 90x30
```

  Actualizing changes made in the `~/.Xresources` file
  without restarting the X Server:
  `xrdb -merge .Xresources`

- **Starting an X session with `startx`:**
  Loging from a virtual terminal(text based) and then run the script `startx`.
  `startx` in turns starts `xinit`.
  `xinit` startx the X Server then starts the `xinitrc` script file:
  (`$HOME/.xinitrc` if found otherwise `/var/X11R6/lib/xinit/xinitrc`)

  Content of `xinitrc` script:
  - System wide and configured key definitions are loaded:

Definitions are in:
`/etc/X11/Xmodmap` and `~/.Xmodmap`
- System wide and configured Resources definitions are loaded:
Definitions are in:
`/etc/X11/Xresources, ~/.Xresources, & ~/.Xfefaults`
- Some user's manually entered programs may start here
- The selected window manager is started.

- **Starting an X session with `xdm/kdm/gdm`:**
When a user does login via a display manager, a similar process as with `startx` will occur, the difference is the script that will be run is:
`/etc/X11/xdm/Xsession` and `~/.Xsession` if it exists.

  Note: Some distributions are running the `~/.xinitrc` from `Xsession` to keep the same environment consistance.

- **X11 in Network**

Preparing the X-Server for access through network

For reasons of security, the X-Server by default will allow only the local user's programs to be displayed. For other users locally or hosts to be allowed to display their X-Programs programs on it, the X-Server needs to open it's security restrictions.

Two security authentication systems are available to X:
- Host Address based authentication : controlled by the command `xhost`
- Token authentification based. Controlled by the command `xauth`.

Host Address based authentication

The X-Server does not need any special options to have the Hosts Address based authentication.

Syntax:

```
xhost [+|-] ClientHostName/IP
```

Examples:

`xhost + localhost`

Allows other users X-programs on the local host to connect to this X-server.

`xhost +`

Allows everybody from anywhere to connect to this X-Server. Dangerous!!!

`xhost + myfriend`

Allows the host `myfriend` to connect to this local X-Server.

`xhost – bugger`

Take the host “bugger” out of the list of allowed hosts.

`xhost -`

Activates the access control mechanism and only the already listed hosts can connect.

Note 1: Only the owner of the X-Server process is allowed to issue the `xhost` command.

Permanently allowing access to an X-Server

There are 2 regular methods to permanently a list of hosts access to the local X-Server:

1. Edit the file enter the command `xhost` for all the hosts allowed in `~/.xinitrc` script, or
2. Create a file called `/etc/Xn.hosts` and enter all the hosts allowed to use the local X-Server. (n=X-Server display port number.)

Token Based Authentication

This authentication method works as a complement to the Host Address based Authentication method. This means that only the hosts that are not in the list of allowed hosts of the Host Address based Authentication method will be affected by this Authentication Method. We can see it like this: The Host Address based Authentication method sets the general rules of which hosts are allowed to connect to the X-Server and the Token based Authentication method sets the rules of who else from other hosts are also allowed. Normally the X-server will compare its tokens stored in the local user's `~/.Xauthority` file with the the incoming client's tokens. The incoming client's tokens are also taken from it's user's `~/.Xauthority` file. If one of them matches, the X-program of the user from the foreign host will be allowed to connect.

For the X-Server to use the Token Based Authentication, an option needs to be given to it when it gets started.

eg. `X -auth` *`AuthenticationFileName`*
or `startx -- -auth` *`AuthenticationFileName`*

Fortunately the popular Display Managers like KDE or GNOME do take care of this.

To list the hosts and keys stored in the user's `~/.Xauthority` on his host:

`xauth [-n] list`
The option `-n` allows to see the real address of hosts written in the file.

Managing the Token Based Authentication

The incoming user's token file is also `~/.Xauthority` on his host. This way the local user's X-programs will always be allowed to connect since the local user's file used by the X-program is the same as the X-server's authentication file.

The program `xauth` is normally used to manage the Authentication token files.
It is better than editing the file by hand.
`xauth` alows to:
add, remove, merge, generate and extract tokens to/from that file.
Syntax:

```
xauth add Host:Display Protocol TokenKeyValue
xauth remove Host:Display
xauth merge ExternalAuthFile
xauth merge -                              ('-' means the STDIN)
xauth extract Outputfile Host:Display
xauth extract - Host:Display       ('-' means the STDOUT)
xauth -niv generate servername:0 .
```

In the `add` function, the TokenKeyValue should be a large hexadecimal value of an even

number of digits: eg. `xauth add marty:0 . 6e7ac1d17814ca478fcf68236d2fb4cb`
Note: the dot ('.') as protocol means the standard MIT-MAGIC-COOKIE-1.

Tip: The 'generate' command is the best way to get the remote's X-server's key and store it in the user's `~/.Xauthority` in the client's host. For this to function, the remote X-server should allow temporary access of the client via the command:

`xhost + ClientHostName`

So here is how it goes best:

- The user on the X-server side issues the command:
  `xhost + ClientHostName ServerHostName`
  (allows the client host to connect to the X-server to generate the key.)
  `xauth -niv generate ServerHostName:0 .` (done only once)
  (The user generates it's own X-Server's key and stores it in `~/.Xauthority`)
- The remote client user the issues the command:
  `xauth -ni generate ServerHostName:0 .`
  (Connects to the remote X-server, gets the X-server's key and stores it in local user's `~/.Xauthority` file.
  Options: n=Do not resolve the X-server's address and store it as address
  `i`=Ignore the lock on the `~/.Xauthority` made from `kdm`
  the '.' at the end(as protocol) means the standard MIT-MAGIC-COOKIE-1.
- The user on the X-server side issues the command:
  `xhost - ClientHostName`
  (Access is again not granted to client unless he has the right key)
  - The user on remote client host(and only this user) can now connect to the X-server. eg. `xterm -display ServerHostName:0`

Running VNC and KDE as startup VNC session.

To make your vnc "sessions" run anything other than the default `twm` or `xterm`, edit the `$HOME/.vnc/xstartup` and change it to run whatever session you'd like...
If you comment out all the lines and add a line like

```
exec /etc/X11/xinit/Xclients
```

you'd run whatever session is defined in `/etc/sysconfig/desktop` (system default), or if you'd like you could use the sessions defined for your display manager(s) like this:

```
exec /etc/X11/xdm/Xsession kde
```

where kde can be replaced with any session defined on the system... do `ls -l /usr/share/apps/switchdesk/`
... removing the Xclient. part, and you have the different sessions possible:-).