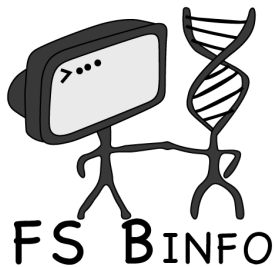


# Einführung in Linux und die Shell



Sebastian Pölsterl 2009 und  
Florian Seifert 2010 / 2011

Fachschaft Bioinformatik

12. Mai 2011

# Zeitplan

---

12.05.	<b>Einführung in Linux und die Shell</b>
19.05.	Arbeiten mit Dateien auf der Shell
26.05.	Kommandozeileneditor: vim
09.06.	Einführung in Java und Eclipse
16.06.	Konditionen, Arrays und Schleifen
30.06.	Rekursion und Iteration
07.07.	Objektorientierte Programmierung
14.07.	File-I/O und Wiederholung
21.07.	Vererbung, Tipps und Fragen
25.01.	Quiz, Evaluation, Fragen

# Kontakt

---

Webseite: `http://www.bioinformatik-muenchen.com/bioinfocom/informatik-tutorium`

Fuer organisatorische Fragen und Probleme: `F.Seifert@campus.lmu.de`  
oder 'flopsbox' auf Facebook.

## Worte zum Geleit oder das Wichtigste zuerst!

---

### Definition

Lernziel Info 1: Handwerliche Faehigkeit, auf einem neutralen Rechner mit Editor und Compiler das banalste Programm schreiben. ->"Hello, World!" (23.10.2011 :-))

### Definition

Dringende Empfehlung fuer "normale" Studierende: Erledige deine Uniaufgaben von Anfang an und soweit wie moeglich unter einer Linuxumgebung.

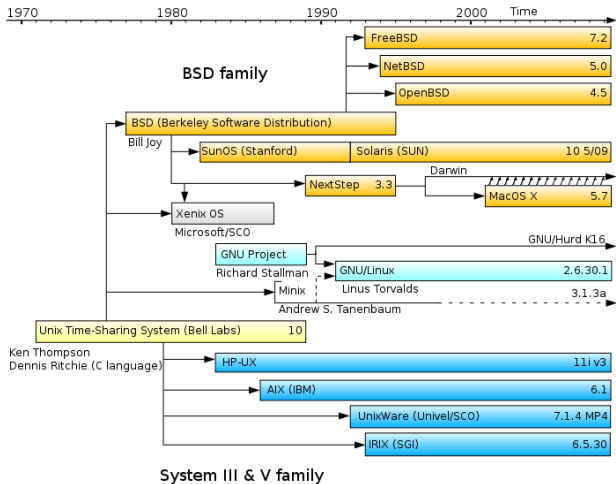
# Was ist Linux?

---



- Freies und meist kostenloses Betriebssystem, das auf dem Linux-Kernel basiert
- Distributionen: Ubuntu, Fedora, OpenSuse, Mandriva, Knoppix, ...
- Graphische Oberflächen: GNOME, KDE, ...
- (Primär) Multiusersystem

# Ein Bild zur Geschichte von "Linux"



## Die wichtigsten Programme

---

Browser	Firefox, Chrome
Textverarbeitung	OpenOffice.org Writer
Tabellenkalkulation	OpenOffice.org Calc
Präsentation	OpenOffice.org Impress
Bildbearbeitung	GIMP
Taschenrechner	KDE: speedcrunch, GNOME: gcalctool
Texteditor	KDE: kate, GNOME: gedit
Dateien entpacken	KDE: ark, GNOME: file-roller

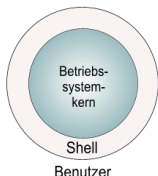
Mehr Äquivalente zu Programmen aus der Windows-Welt findet man unter

<http://blogs.sun.com/richb/resource/freeware-list.html>

# Das wichtigste Programm

---

## Die Shell



Konsole unter KDE,  
gnome-terminal unter GNOME

Tip: Die Tastenkombination Alt + F2 oeffnet auf vielen graphischen Oberflaechen eine "Kommandozeile" zum starten von Programmen :)

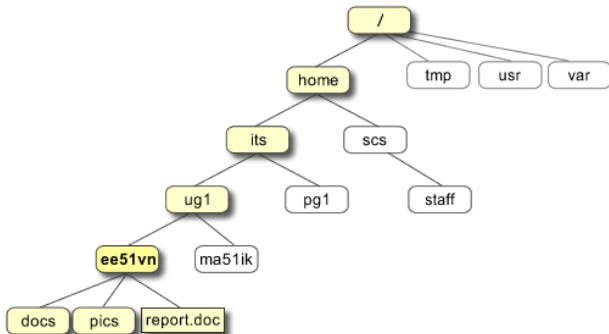
# Was ist die Shell?

---

- Graphische Oberflächen (GUI) dienen dazu die Bedienung einfacher zu machen
- Funktionen auch ohne graphische Oberfläche von der sog. Shell verfügbar
- Shell nimmt Kommandos entgegen und “bittet” den Kernel sie auszuführen
- Die Shell stellt viele nützliche Werkzeuge zur Verfügung
- Mit Hilfe der Shell lassen sich Prozesse automatisieren => Effizienz

# Das Dateisystem

---



# Programmkategorien

---

- `/usr/share/gnome`
- `/usr` = Gültigkeitsbereich, `share` = Kategorie, `gnome` = Anwendung
- `bin` = Programme (meist `/bin`, `/usr/bin`, `/usr/sbin`)
- `lib` = von Programmen verwendete Bibliotheken (libraries)
- `doc` = Dokumentation
- `etc` = Konfigurationsdateien
- `src` = Quellcode für Programme
- `/home` = persönliche Dateien von Benutzern

# Pfade

---

- Versteckte Dateien beginnen immer mit “.” (zB. “.local”)
- “.” bezeichnet das aktuelle Arbeitsverzeichnis
- “..” bezeichnet das Verzeichnis eine Ebene höher
- “~” bezeichnet das HOME-Verzeichnis
- Zwischen Groß- und Kleinschreibung wird unterschieden
- Absolute Pfade beginnen immer mit /
- Relative Pfade sind relativ zum aktuellen Arbeitsverzeichnis

# Einfache Dateioperationen I

---

- Verzeichnis Inhalt auflisten: `ls [OPTIONEN] [VERZEICHNIS]`
  - Wenn kein Verzeichnis angegeben ist wird Inhalt des aktuellen Arbeitsverzeichnisses ausgegeben
  - Option `-l` gibt Inhalt in Listenformat aus
  - Option `-a` zeigt auch versteckte Dateien an
- Zu Verzeichnis wechseln: `cd VERZEICHNIS`
- Verzeichnis erstellen: `mkdir VERZEICHNIS...`

## Übung:

- 1. Liste alle Dateien und Verzeichnisse in `/`
- 2. Erstelle Verz. "Meine Spielwiese"
- 3. Erstelle darin das Verz. "doc"



# Einfache Dateioperationen II

---

- Datei kopieren: `cp QUELLE ZIEL`
- Verzeichnis kopieren: `cp -r QUELLE ZIEL`
- Datei/Verzeichnis verschieben: `mv QUELLE ZIEL`
- Es können auch mehrere Dateien als QUELLE angegeben werden, dann muss ZIEL ein Verzeichnis sein

## Übung:

- 1. Kopiere Verz. `/usr/share/doc/bash` nach "doc"
- 2. Erstelle Ordner "bash neu" in "doc"
- 3. Verschiebe die Datei FAQ und alle Dateien mit der Endung `.gz` aus "bash" nach "bash neu"
- 4. Nenne die Datei FAQ in `.faq` um
- 5. Liste den Inhalt des Verz "bash neu" auf so dass `.faq` angezeigt wird

## Einfache Dateioperationen III

---

- Datei entfernen: `rm DATEI...`
- Verzeichnis entfernen: `rm -r VERZEICHNIS` oder `rmdir VERZEICHNIS`
- Mit dem Platzhalter `*` können mehrere Dateien mit ähnlichem Namen angesprochen werden (`a*` = alle Dateien die mit "a" anfangen)

### Fact

**ACHTUNG:** *Es wird in der Regel nicht nachgefragt => Weg ist weg!*

### Übung:

1. Lösche das Verz. "bash"
2. Nenne "bash neu" in "bash" um
3. Liste alle Dateien mit der Endung `.so` in `"/usr/lib"` auf

## Keyboard Shortcuts

---

- Pfeil hoch: vorheriger ausgeführter Befehl
- Pfeil runter: anschließend ausgeführter Befehl
- Tab: Vervollständigung
- Strg + R: Chronik aller ausgeführten Befehle durchsuchen
- Strg + A / Pos 1: An Zeilenanfang springen
- Strg + E / Ende: An Zeilenende springen
- Strg + C: Befehl abbrechen
- Strg + L: Shell leeren
- Mehr unter <http://linuxhelp.blogspot.com/2005/08/bash-shell-shortcuts.html>

**Übung:** Probiere alle Befehle aus

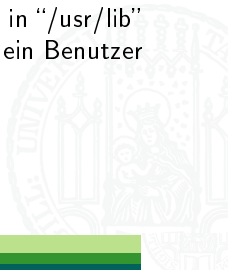
# Man pages

---

- Dokumentation von Befehlen
- `man BEFEHL`
- Alternativen: `info BEFEHL` und `BEFEHL --help`

## Übung:

- 1. Man page von `man` lesen
- 2. Finde Optionen von `ls` um nach Größe bzw. Änderungsdatum zu sortieren
- 3. Finde die größte und neuste Datei mit der Endung `.so` in `"/usr/lib"`
- 4. Finde ein Programm, das ausgibt in welchen Gruppen ein Benutzer Mitglied ist
- 5. Geb die Gruppen des Benutzers `seifert` aus
- 6. Finde heraus was der Befehl `less` tut



# less

---

- Programm zum lesen von Dateien (sog. Pager)
- Wird auch verwendet um Man pages darzustellen

## **Übung:**

- 1. Finde heraus wie man mit less in Dateien sucht

# history

---

- Chronik aller eingegeben Befehle
- Chronik leeren: `history -c`
- Letzten  $N$  Befehle ausgeben: `history N`
- Den  $N$ -ten Befehl ausführen: `!N`

## Übung:

1. Gib die letzten 10 eingegeben Befehle aus



# find

---

- Befehl um nach Dateien zu suchen
- `find VERZEICHNIS -name MUSTER`
- 
- **Übung:**
  1. Finde alle README Dateien in “/usr/share/doc”
  2. Finde alle versteckten Verzeichnisse in deinem HOME-Verzeichnis
  3. Finde alle leeren Verzeichnisse in deinem HOME-Verzeichnis

# Übungen für Daheim

---

1. Finde heraus was die Befehle `kill` und `cat` tun
2. Finde heraus was genau der Befehl `ps aux` ausgibt
3. Finde einen Befehl, der das aktuelle Datum und die Uhrzeit ausgibt
4. Finde heraus was die Datei `/etc/passwd` beinhaltet
5. Lese das Kapitel "Benutzer- und Berechtigungskonzepte unter Linux" unter [http://www.selflinux.org/selflinux/html/nutzer\\_unter\\_linux.html](http://www.selflinux.org/selflinux/html/nutzer_unter_linux.html)