Programmier- und Unix-Propädeutika

Eine Einführung für Erstsemester

Carl von Ossietzky Universität Oldenburg
21. September 2026

Leonard Langer

Meta

Beteiligung

1. Stellt gerne Fragen, auch während der Präsentation
2. Stellt gerne Fragen, auch während der Präsentation
3. Macht euch bemerkbar, wenn euch etwas zu schnell geht
   - Die Präsentation enthält viele potenziell neue Informationen.
   - Ihr sollt euch nicht überrannt fühlen!
   - Wenn euch etwas unklar ist, geht es anderen wahrscheinlich auch so
4. Reinrufen erlaubt, solange alle zu Wort kommen

Ablauf der Vorkurse

Ablauf der Vorkurse: Inhalte

Tag 1

Was ist UNIX?

Dateisysteme

Kommandozeile

Live Demo

UNIX

Tag 2

Scripte

Variablen & Datentypen

Kontrollfluss

Abstraktion

Komplexe Daten

Programmieren

Tag 3 & 4

Wir machen ein Videospiel!

Spieldesign

Grafiken darstellen

Spiel-Logik

Interaktivität

Sounds

Programmieren & UNIX

Ablauf der Vorkurse: Wer bin ich?

Leonard Langer

- 26 Jahre
- Studiere seit 2020
- FSR-Mitglied seit 2023

Kontakt

- `leonard.langer@uni-oldenburg.de` (Mail)
- `sntx.space` (Discord)

Events

- Video-Game-Night 2025-10-09   1800
- GameJam   2025-11-06~08 1700

Zeiten sind `YYYY-MM-DD~DD HHMM`; `~` def.: `von ~ bis`

Ablauf der Vorkurse: ARBI-Zugänge

Department für Informatik

- ARBI (**A**bteilung **R**echner- und Netz**b**etrieb **I**nformatik)
- Ansprechpartner: Jörg Lehners

Logins

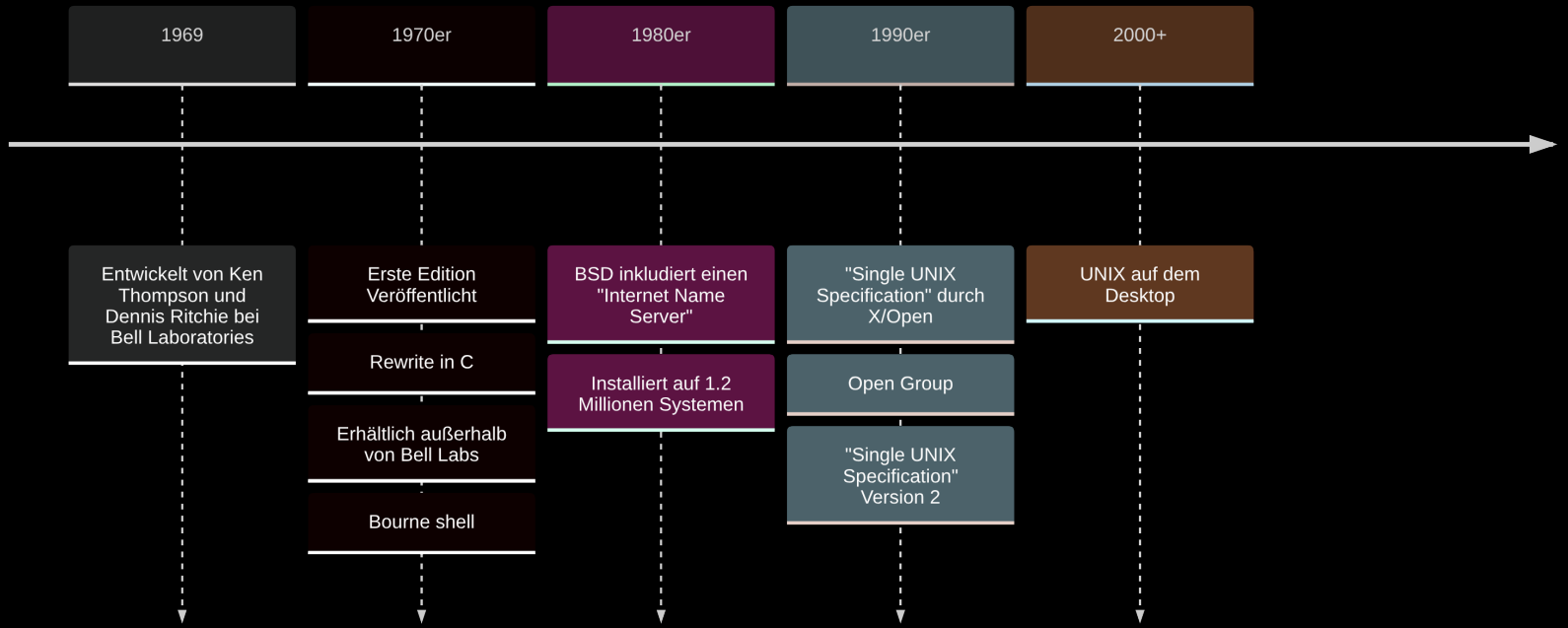
- Getrennt von anderen Uni-Logins
- `<accountname>@informatik.uni-oldenburg.de`

Passwörter

- Euch wird eins zugewiesen
- Sobald ihr euren eigenen Account habt, solltet ihr das ändern!
- Man sieht nicht, was man tippt!

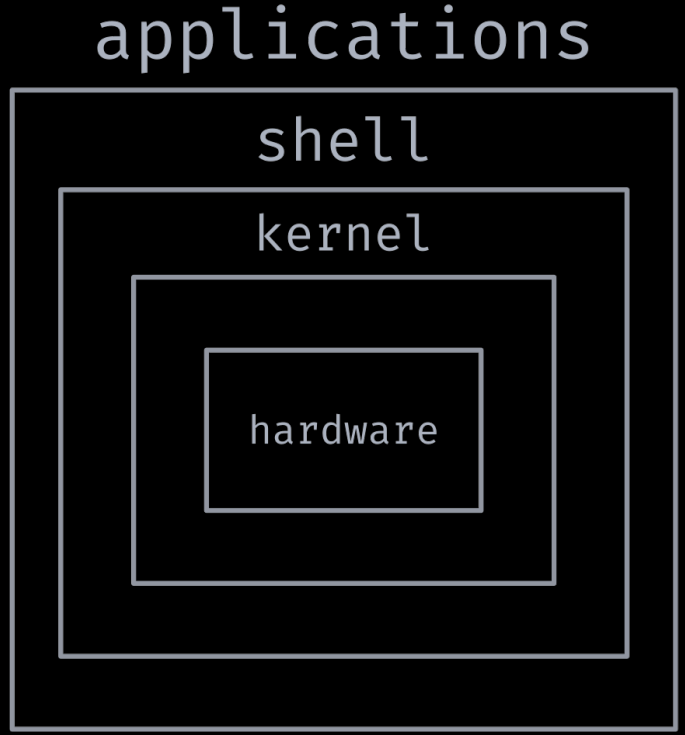
Geschichte

Geschichte: Timeline von Unix

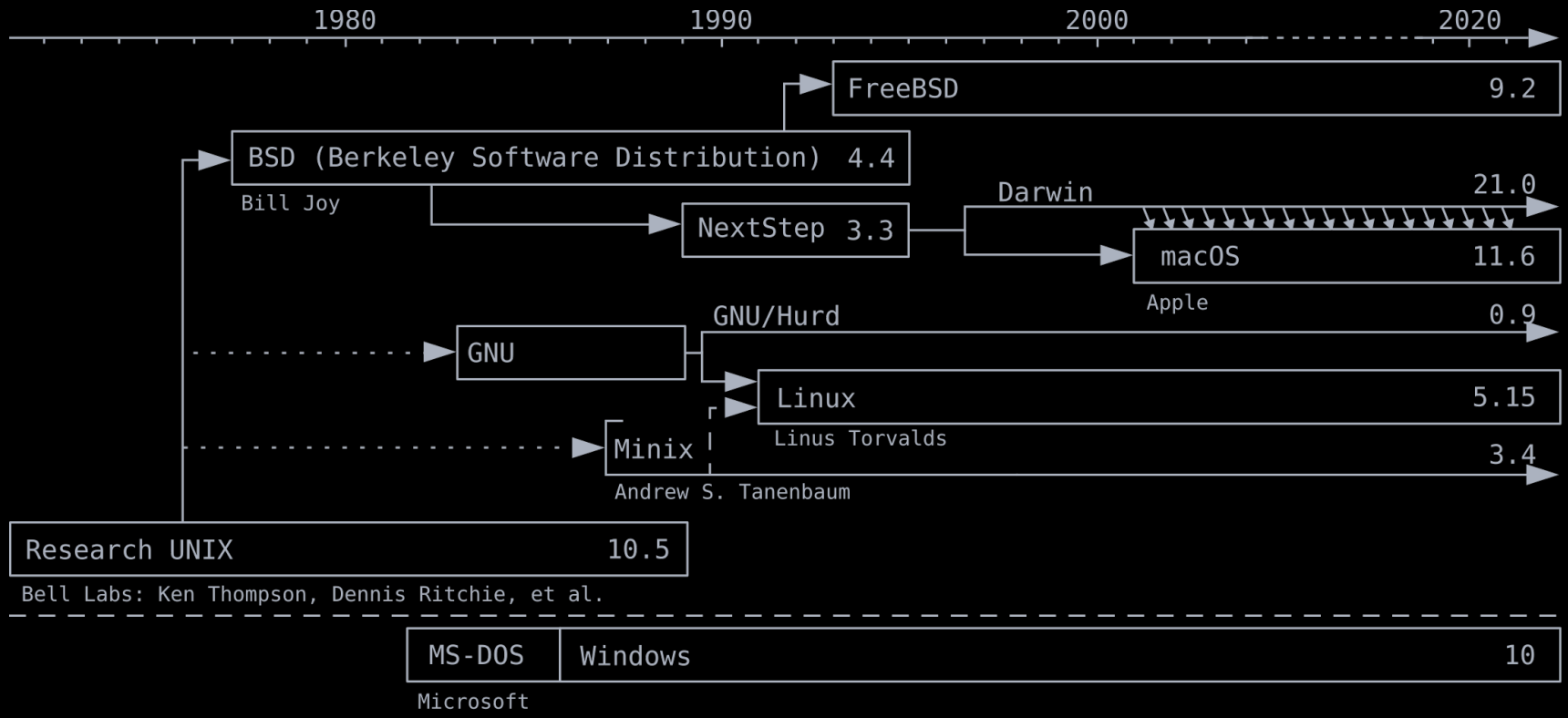


Was ist Unix?

Was ist Unix: Architektur



Was ist Unix: Unix-like

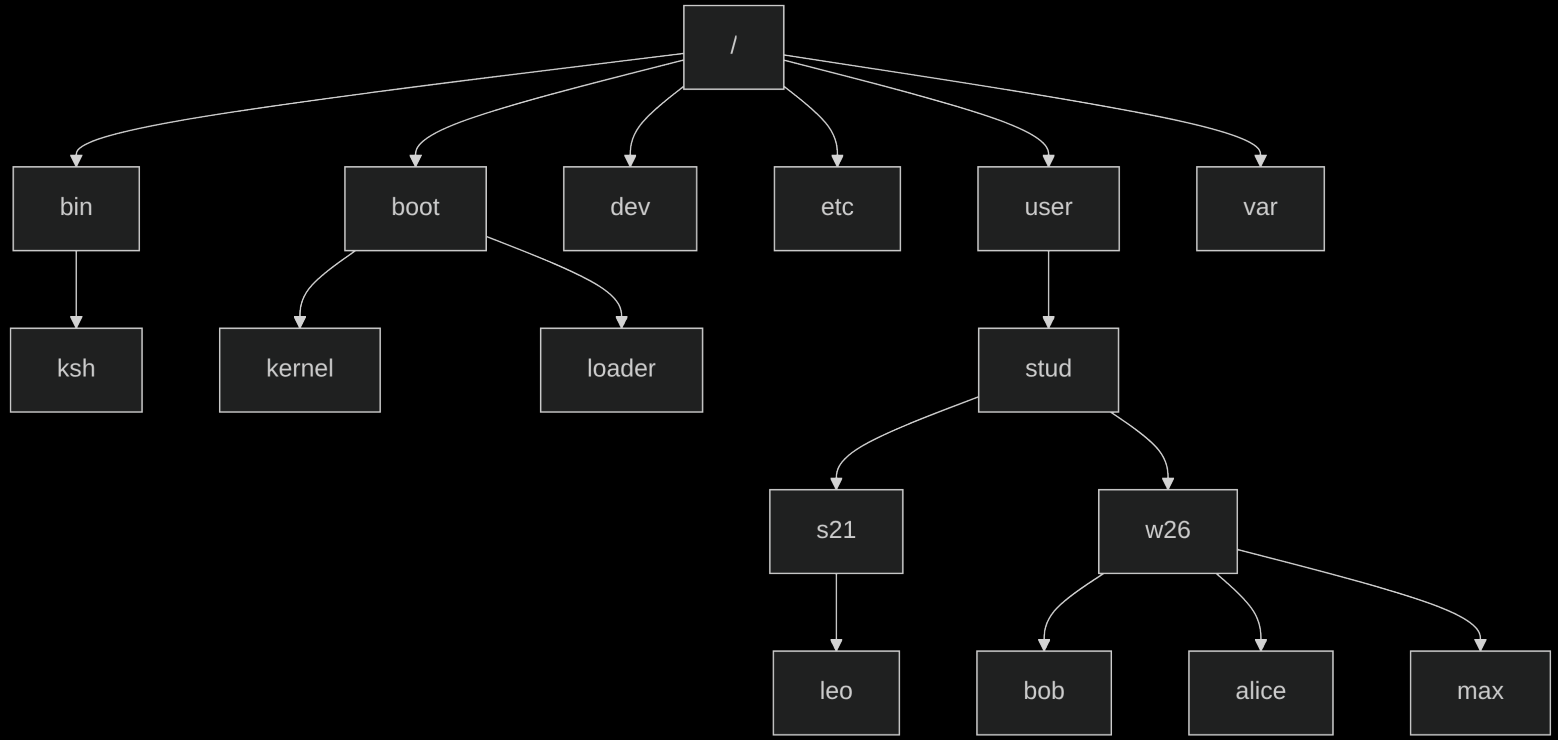


vgl. https://commons.wikimedia.org/w/index.php?curid=1667764

Was ist Unix: Dateisystem I

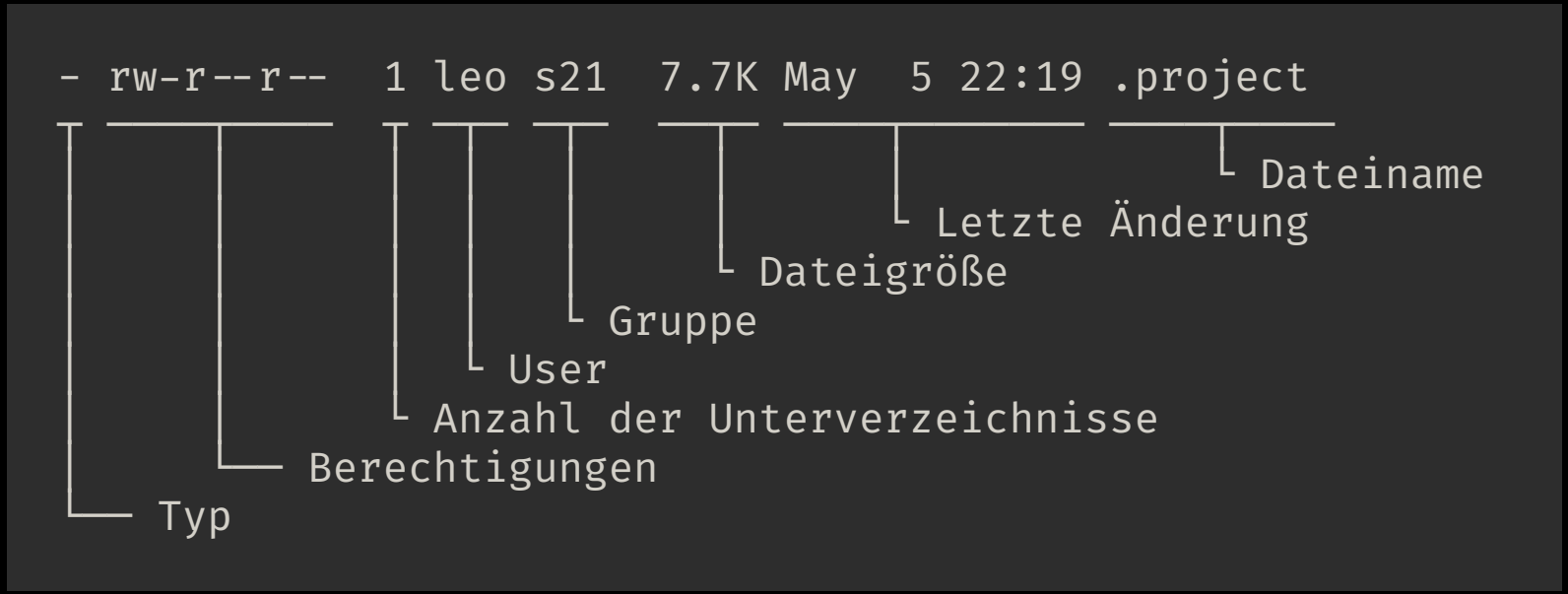
```
├── bin  # Essential user command binaries
├── boot # Static files of the boot loader
├── dev  # Device files
├── etc  # Host-specific system configuration
├── home # User home directories
├── lib  # Essential shared libraries and kernel modules
├── mnt  # Mount point for a temporarily mounted filesystem
├── opt  # Add-on application software packages
├── run  # Run-time variable data
├── tmp  # Temporary files
├── usr  # Shareable, read-only data
└── var  # Variable data files
```

Was ist Unix: Dateisystem II



Euer Home-Verzeichnis wird in `/user/stud/w26` liegen

Was ist Unix: Dateisystem III



Was ist Unix: Philosophie

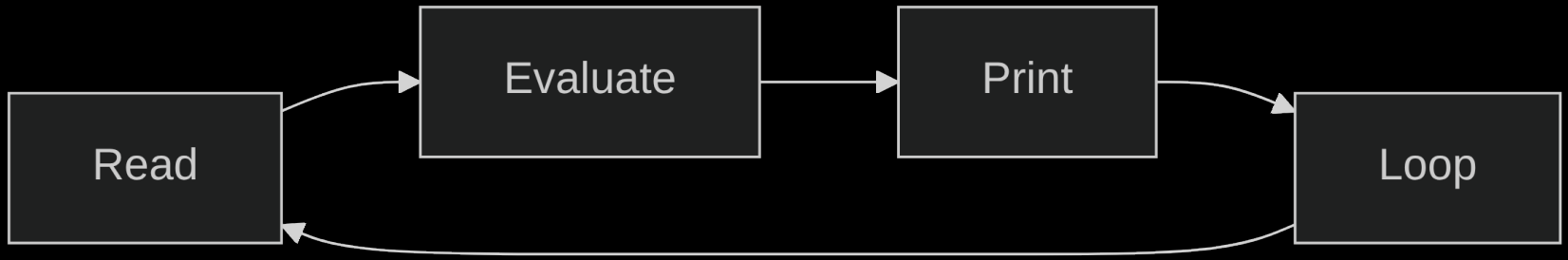
- Write programs that do one thing and do it well.
- Write programs to work together.
- Write programs to handle text streams, because that is a universal interface.

Peter H. Salus, A Quarter-Century of Unix, 1994

CLI

(Command Line Interface)

CLI: Shell



CLI: Shells

POSIX Shells

- Bourne Shell (sh)
- C Shell (csh)
- Korn Shell (ksh)
- Bourne Again Shell (bash) → wenn `-sh` übergeben wird

Non-POSIX Shells

- fish (fish)
- Nushell (nu)
- PowerShell (ps)
- Z shell (zsh)
- …

CLI: Eingabeaufforderung

```
<User>@<Host>:<aktuelles_Verzeichnis><Promptindikator><Prompt>
```

```
leo@duemmer:~>
```

CLI: Befehle

```
[ arch base32  base64 basename  basenc cat chgrp  chmod chown
chroot  cksum comm  coreutils cp  csplit cut  date dd  df dir
dircolors  dirname du echo  env expand expr factor false  fmt
fold groups hashsum head hostid hostname id install join kill
link ln logname ls mkdir  mkfifo mknod mktemp more mv nice nl
nohup nproc numfmt od paste  pathchk pinky pr printenv printf
ptx pwd readlink  realpath rm rmdir seq shred shuf sleep sort
split stat  stdbuf stty  sum sync tac  tail tee  test timeout
touch tr  true truncate tsort  tty uname unexpand uniq unlink
uptime     users      vdir     wc      who     whoami     yes
```

CLI: Befehle verwenden

man [man options] [[section] page ...] ...

- **bold text** genau so eingeben wie dargestellt
- italic text durch entsprechendes Argument ersetzen
- [**-abc**] beliebige Argumente in [ ] sind optional
- **-a**|**-b** die Optionen können nicht zusammen genutzt werden
- argument ... das Argument kann wiederholt werden
- [expression] ... komplette Ausdrücke in [ ] sind wiederholbar

Siehe: `man man`

CLI: Interaktives Projekt

Wir organisieren unsere Studienunterlagen!

CLI: Befehle zum Einstieg

- Lernen
  - man
- Dateisystem navigieren
  - ls, pwd, mkdir, cd, mv, cp, rm
- Dateien schreiben
  - touch, nano
- Dateien lesen
  - cat, less, file
- Dateien finden
  - find
- Streams verarbeiten
  - grep, head, tail

- nano: Es gibt viele weitere Editoren wie z. B. Vim
- finger: `.profile`, `.projects`

CLI: Spezielle Verzeichnisse

- Es gibt spezielle Verzeichnisse, die euch das Leben einfacher machen können

Verzeichnisse

- `.` aktuelles Verzeichnis
- `..` übergeordnetes Verzeichnis
- `~` home-Verzeichnis
- `-` Verzeichnis vor letztem cd
- `/` Wurzelverzeichnis der Festplatte

CLI: Ein- und Ausgabeströme

- Programmaufrufe produzieren i. d. R. Ausgaben
  - Oft ist es nützlich diese weiterzuverarbeiten

Operatoren

- **>** schreibt den Output in eine Datei
- **>>** hängt den Output einer Datei an
- **|** gibt den Output an einen weiteren Befehl weiter
- **<** liest den Inhalt einer Datei als Input für einen Befehl

Beispiele

```
tail -n 1 < Studienverlaufsplan >> Aktuelles-Semester
```

```
cat Studienverlaufsplan | tail -n 3 > Uni-Info
```

CLI: Umgebungsvariablen

- Werden von Programmen gelesen
  - Beeinflussen das Verhalten von Programmen
- Können auch direkt von Befehlen angegeben werden
  - Existieren dann nur für diesen Befehl

Befehle

- `export VARIABLE=VALUE` weist der Variable einen Wert zu
- `echo $VARIABLE` gibt den aktuellen Wert der Variable aus
- `env` gibt alle Variablen und deren Werte aus

Beispiel

```
export EDITOR=vim
$EDITOR text.txt
```

CLI: Drucken in der ARBI

- Druckquota wird jeden Tag um 50 Seiten erhöht
  - Maximal 500 Seiten
- Dateien müssen im PDF- oder Postscript-Format sein

Befehle

- `lpr [-P DRUCKERNAME] DATEINAME` erstellt Druckauftrag
- `lpq [-P DRUCKERNAME]` zeigt Druckaufträge an
- `lprm [-P DRUCKERNAME] [JOB_ID|all]` entfernt Druckaufträge

Beispiele

```
PRINTER=lwrot lpr paper.pdf
```

```
lpr -P lwrot paper.pdf
```

CLI: Fernzugriff per SSH

- Eignet sich wunderbar um …
  - von zuhause aus auf ARBI-Systemen zu arbeiten
  - schnell vom Laptop zu drucken, ohne sich vor Ort anzumelden

Befehle

- `ssh [ssh options] ... [user@]hostname` — Shell zum anderen Rechner
- `sftp [sftp options] ... [user@]hostname` — Dateien übertragen

Beispiele

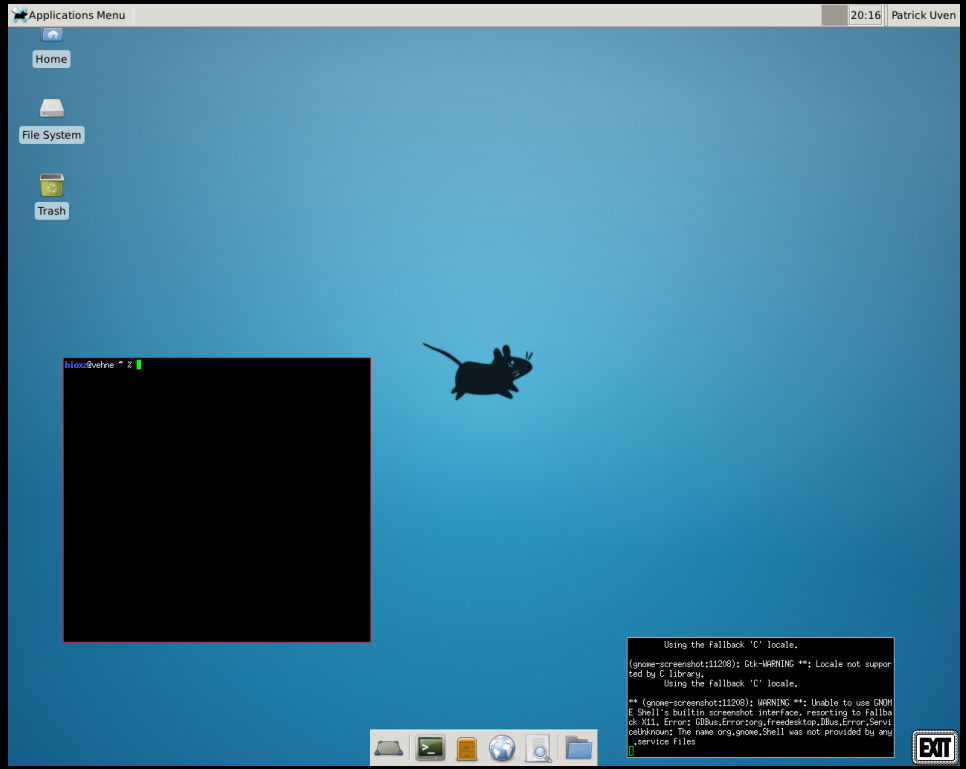
```
ssh leo@duemmer.informatik.uni-oldenburg.de
```

```
sftp duemmer.informatik.uni-oldenburg.de
```

GUI

(Graphical User Interface)

GUI: Komposition



HILFE!!!!1!!11!! Help!

Was macht der Computer da?!

Bei Unklarheiten …

• man-pages	konsultieren
• KommilitonInnen	fragen
• TutorInnen	ausquetschen
• MitarbeiterInnen der ARBI	um Hilfe bitten
• Internet	durchsuchen

⚠ Warning

Befehle und Programme aus dem Internet und von KI-Sprachmodellen sind oft sehr hilfreich. Doch seid euch sicher, dass ihr alles versteht, bevor ihr diese ausführt, da auch kleine Details oft große Auswirkungen haben können!

Danke

Diese Präsentation wurde …

- präsentiert mit
  - **presenterm** (https://mfontanini.github.io/presenterm/)
- erstellt mit
  - **helix** (https://helix-editor.com/)
- versioniert mit
  - **git** (https://git-scm.com/)
- dargestellt in
  - **kitty** (https://sw.kovidgoyal.net/kitty/)
- zur Verfügung gestellt von
  - **nixos** (https://nixos.org/)

… und mit viel Liebe gemacht <3