

# Unixkurs 2012

**Eine Einführung für Erstsemester**

Patrick Uven  
Merlin Wasmann  
Universität Oldenburg

- Login und Passwort
- Was ist ein X-Terminal? Was ist eine Shell?
- Dateisystem und die wichtigsten Befehle
- Drucken
- Einige Anwendungsprogramme:
  - ▶ Editor *nedit*
  - ▶ Email-Programm *pine*
  - ▶ Internet-Browser *Firefox*
  - ▶ Acrobat Reader
- Hilfestellungen

- Department für Informatik
  - ▶ Eigenes Rechnernetz, eigene Rechnerräume und eigene **Abteilung Rechner- und Netzbetrieb Informatik** (ARBI)
  - ▶ Informatik-Studierende arbeiten auf **ARBI-Rechnern** mit **Informatik-Logins** und **Informatik-Email-Adresse** (vorname.nachname@informatik.uni-oldenburg.de)

- Muss bei der ARBI beantragt werden
- Gilt für alle Unix-Maschinen der ARBI
- Login-Name ist frei wählbar
  - ▶ Mehrere Alternativen angeben, da manche Namen bereits vergeben sind!

- Betriebssystem
  - ▶ Entwickelt von den Bell Laboratories (Start ca. 1969)
- 1989 Unix System Release 4.0 = heutiger 'Standard'
- BSD-Unix: Erweiterung von Unix, entwickelt von der Universität von Kalifornien in Berkley
- Viele Derivate von Unix: Linux, Solaris/Sun-OS, Ultrix, FreeBSD, Mac OS X, ...
- 'Shell' als Kommandoschnittstelle

- Multiuser-fähig:
  - ▶ Mehrere Personen können zugleich an demselben Rechner arbeiten
- Systemverwalter richtet neuen Benutzern die 'Arbeitserlaubnis', ein Login, auf dem Rechner ein.
- Arbeitsverzeichnisse liegen auf zentralem Server (**taifun**).

- Jeder Benutzer besitzt:
  - ▶ einen Login-Namen
  - ▶ ein Passwort *Diese müssen beim Anmelden am Rechner eingegeben werden.*
  - ▶ ein (home-)Verzeichnis (home directory), in dem seine Daten gespeichert werden. *Der Name des home-Verzeichnis ist gleich dem Login-Namen. Die home-Verzeichnisse aller Studierenden eines Semesters befinden sich im selben Unterverzeichnis: /user/stud/s12/*

- Das erste Passwort wird von der ARBI vergeben.
  - ▶ Automatisch generierte, beliebige Folge von Buchstaben und Ziffern
  
- Passwörter können geändert werden.



- Auswahl eines 'guten' Passworts
  - ▶ Mindestens 7 Zeichen
  - ▶ Enthalten nicht nur (Groß- und Klein-) Buchstaben, sondern auch Ziffern und Sonderzeichen
  
- Achtung: Groß- und Kleinbuchstaben werden unterschieden.
  
- Passwort vergessen:
  - ▶ Jörg Lehnert oder Olaf Wendt ansprechen

# Erste Schritte am Rechner

- 1 Anmelden (Einloggen) am Rechner
- 2 Die 'Erstsemesterbenutzungsoberfläche'
- 3 Ändern des Passwortes
- 4 Abmelden
- 5 Sperren des Arbeitsplatzes

# Anmelden (Einloggen)

- Einloggen auf beliebigem Rechner der ARBI
- Eingabefenster für Login-Namen und Passwort
  - ▶ Achtung: Der Cursor muss sich auf dem entsprechenden Feld befinden.
  - ▶ Eingabe des Passwortes:  
*Das Passwort muss eingetippt werden. Man sieht nicht, was für Zeichen man tippt!*
- Warten...
- Benutzungsoberfläche erscheint

# Die Ersti-Benutzungsoberfläche

- Benutzungsoberflächen durch Window-Manager gestaltet. (*Welche Buttons werden wo angezeigt.*)
- Unterschiedliche Window-Manager
- Erstsemester (Ersti)-Benutzungsoberfläche auf icewm aufbauend von der Fachschaft erstellt
- Themes um das Aussehen (Farbe u.ä.) zu variieren

(von rechts nach links)

- EXIT- Button: *Abmelden vom Rechner*
- Email-Button: *Aufruf des Programmes pine*
- Workspaces: *Virtuelle Monitore*
- Mozilla-Button: *Aufruf des Internet-Browsers Mozilla*
- Xterm-Button: *Öffnen eines Terminal-Fensters*
- WindowList: *Übersichten über geöffnete Fenster*
- Menü: *Übersicht über eine Auswahl von aufrufbaren Anwendungsprogrammen*

## Abmelden vom Rechner

- Abmelden, wenn mit der Arbeit fertig
- Abmelden durch:
  - ▶ EXIT-Button
  - ▶ Eintrag 'Logout' im Menü
- Kurzes Verlassen des Rechners (<15min) → Arbeitsplatz sperren
  - ▶ Eintrag 'Sperren' im Menü

*Bei Ihrer Rückkehr können Sie den Rechner durch Eingabe Ihres Passwortes wieder entsperren.*

- Ähneln den bekannten Fenstern unter Windows
  - ▶ Rollbalken  
*Achtung: linke Maustaste abwärts, rechte Maustaste aufwärts scrollen*
  - ▶ Icons zum Maximieren, Minimieren und Schließen des Fensters (rechts oben)
  
- Eingaben sind nur im aktiven Fenster möglich.


# Xterm – eine Terminal-Emulation

- Xterm: Fenster, in dem
  - ▶ eine Shell Unix-Befehle entgegen nimmt
  - ▶ Anwendungsprogramme gestartet werden können

*Beispielsweise können wir im xterm-Fenster den Befehl zum Ändern des Passwortes eingeben.*
- Aufruf (Öffnen) eines neuen xterm-Fensters:
  - ▶ Anklicken des xterm-Buttons oder
  - ▶ über das Menü oder
  - ▶ in ein bereits gestartetes xterm-Fenster den Befehl `xterm&` eingeben



# Xterm – eine Terminal-Emulation

- Beenden:
  - ▶ über Window-Icon 
  - ▶ oder Befehl **exit** im xterm-Fenster eingeben
  - ▶ oder mit Tastenkombinationen Strg-D bzw. Control-D

# Was ist eine Shell?

- Benutzer kommuniziert mit dem Betriebssystem über eine Shell
- Shell = Kommandointerpreter
  - ▶ Nimmt Kommandos des Benutzers entgegen und
  - ▶ setzt sie in Systemaufrufe um, die von Betriebssystemkern ausgeführt werden.

# Was ist eine Shell?

- Unterschiedliche Shells (tcsh, bash, zsh, . . . ) auf einem System einstellbar
- Ist die Shell zum Annehmen eines Befehls bereit, meldet sie sich mit einem so genannten Prompt, z.B. '>'
- Aufbau der Eingabeaufforderung der shell:  
<Benutzer>@<Rechnername>:<aktuelles Directory><Prompt> z.B. testv@lava: >

## Ändern des Passwortes

- NIS (Network Information System): Zentrale Datensammlung für alle Logins und Passwörter auf den ARBI-Rechnern.
- Geben Sie den Unix-Befehl ein: **yppasswd**
- Sie müssen dann
  - ▶ das alte Passwort
  - ▶ und zweimal das neue Passwort eingeben.
- Die Änderung gilt für das **gesamte** Rechnernetz der ARBI.

## Aufruf von Anwendungsprogrammen

- Beispiel: Bearbeiten von Texten
- Programm: **nedit**
- Aufruf: im xterm durch  
nedit <Dateiname>  
*Shell 'wartet' bis nedit beendet wird.*
- Besser: nedit <Dateiname> &  
*Shell meldet sich sofort mit dem Prompt und ist für den nächsten Befehl bereit.* '&' kann an jeden Befehl in der Shell angehängt werden, wenn die Shell schon neue Befehle annehmen können soll, bevor der letzte Befehl beendet ist.

- Übersicht bewahren bei vielen offenen Fenstern
- Workspaces: 'Virtuelle' Monitore
  - ▶ Workspace ist vier Mal so groß wie Bildschirmfläche
  - ▶ Umschalten zwischen verschiedenen Bildschirmausschnitten
- Window List:
  - ▶ Liste aller geöffneten Fenster
  - ▶ geordnet nach Workspaces

# Copy and Paste

- Copy and Paste funktioniert unter X mit
- Markieren mit der Maus (Copy)
- Drücken der mittleren Maustaste (Paste)
- Das übliche STRG+C, STRG-V funktioniert natürlich auch.

## Anlegen und Bearbeiten von Textdateien mit **nedit**:

- Aufruf:
  - ▶ Im xterm durch `nedit [<Dateiname>]&`
  - ▶ oder entsprechende Icons oder Einträge im Startmenü anklicken
  
- Bearbeiten von Texten:
  - ▶ Cursor kann (wie gewohnt) verwendet werden
  - ▶ Kürzel zum Kopieren, Ausschneiden, Einfügen usw. wie üblich.



Anlegen und Bearbeiten von Textdateien mit **nedit**:

- Beenden
  - ▶ Close: schließt die Dateien im aktuellen Fenster
  - ▶ Exit: schließt alle mit nedit geöffneten Dateien und beendet das Programm

## Surfen im Internet

- Internet-Browser
  - ▶ Ähnlich zu den üblichen Internet-Browsern
- Aufruf durch
  - ▶ Startmenü, Icons oder xterm (firefox&)
- Alternativ sind auf den Rechnern auch die Internet-Browser Seamonkey, Opera, Dillo u.v.a. verfügbar.

- Anzeigen und Drucken von pdf-Dokumenten
  - ▶ pdf: Austauschformat für Textdokumente
  
- Aufruf durch
  - ▶ Startmenü oder Icons
  - ▶ Im xterm durch `acroread` oder `acroread <dateiname>`

## Emailprogramm

- Start
  - ▶ Im xterm: pine
  - ▶ Oder durch Button der Ersti-Oberfläche
  
- Nach dem Start: Fenster mit Menü bestehend u.a. aus
  - ▶ Help
  - ▶ Compose Message
  - ▶ Message Index
  - ▶ ...
  - ▶ Quit

- Eingabe von Befehlen
  - ▶ Auswahl durch Pfeiltasten
  - ▶ Befehle, die aus einem Buchstaben oder control-Sequenz bestehen
  - ▶ Eine Maus ist in diesem Programm nutzlos!

- Ein Unixbefehl hat die Form (Syntax)  
<Befehlsname> [-opt] [args]
  - ▶ Der Befehlsname besteht nur aus Kleinbuchstaben.
  - ▶ opt sind Optionen des Befehls.
  - ▶ args sind die Argumente des Befehls.
  - ▶ bei der Beschreibung eines Befehls bedeutet [...], dass der Teil in den Klammern auch weggelassen werden darf. Beim Aufruf des Befehls werden diese Klammern nicht eingegeben.
  
- Ein Befehl kann im xterm nach dem Prompt eingegeben werden.

## Auflisten der Dateien im aktuellen Verzeichnis

- Syntax des ls-Befehls:  
ls [-aAbcCdFFgilMnOpqrRstux1] [ file..]  
*Die meisten Optionen werden nur selten gebraucht.*

## Beispiele

- Alle Dateien (ohne versteckte Dateien): **ls**
- Alle Dateien mit zusätzlichen Informationen: **ls -l**
- Alle Dateien, die mit 'b' beginnen inkl. zusätzlicher Information: **ls -l b\***
- Anzeigen auch der versteckten Dateien: **ls -al**
- Farbige Unterscheidung zwischen Dateitypen: **ls -color**

- Datei- und Verzeichnisnamen müssen nicht immer vollständig ausgeschrieben werden.
- Die Shell (die im xterm läuft) vervollständigt Dateinamen, wenn die **Tabulatortaste** gedrückt wird.
- Mengen von Namen können mit Hilfe von Sonderzeichen beschrieben werden. Dabei bedeuten:
  - \* : beliebige viele Zeichen
  - ? : genau ein beliebiges Zeichen
  - [ ] : genau ein Zeichen aus der in den Klammern angegebenen Menge
- `??_*[a-z]` beschreibt somit alle Dateinamen, die an der dritten Stelle einen Unterstrichstrich besitzen und mit einem Kleinbuchstaben enden.



# Kleine Pause

- Arten von Dateien
- Struktur des Dateisystems
- Arbeiten mit Dateiverzeichnissen
- Arbeiten mit Dateien

- Verzeichnisse
- Normale Dateien (files)
  - ▶ Textdateien  
Beliebiger Text, Programmtexte, ...  
Dot-Files (versteckte Dateien) enthalten Unix-Kommandos,  
Systemeinstellungen, ...  
Die Datei `.forward` enthält die Email-Adresse, an die eingehende  
emails weitergeleitet werden sollen.

- Normale Dateien (files)
  - ▶ Binärdateien  
Ausführbare Programme  
Dateien in speziellen Formaten, die nur mit speziellen  
Programmen angezeigt werden können  
(pdf-Dateien, Postscript-Dateien, Bilder,...)

⋮

- Verzeichnisse:
  - ▶ bin: ausführbare Programme
  - ▶ Mail: Dokumente zu ein- und ausgehenden Emails
  - ▶ viele weitere
  
- 'Versteckte' Dateien: Dot-Dateien (Punktdateien)
  - ▶ Dateiname beginnt mit einem Punkt
  - ▶ Beinhalten System- und Programmeinstellungen
  - ▶ Beispiele:
    - .forward
    - .signature

Der Befehl: 'ls -l b\*' liefert die Ausgabe

```
drwxr-xr-x 2 testv ss02 512 Sep 16 02:28 bin/
```

# Zugriffsrechte: `rw-r-xr-x`

- Für jede Datei kann festgelegt werden, wer das Leserecht (r), Schreibrecht (w) und Ausführungsrecht (x) besitzt:
- Die ersten 3 Stellen → Besitzer
- Die zweiten 3 Stellen → Gruppe
- Die letzten 3 Stellen → Andere
- Verändern mit → **chmod**  
z.B. `chmod o-rx bin/`

# Unix-Befehle für Dateien

<file> sei der Name einer Datei (ggf. inkl. Pfad)

- `cp <file> <Kopie>` oder `cp <file> <dir>`
  - ▶ kopiert (copy) den Inhalt der Datei <file> in die Datei <Kopie> bzw. in das Verzeichnis <dir>
- `cp -r <dir1> <dir2>` kopiert das Verzeichnis <dir1> samt Inhalt
- `rm <file>`
  - ▶ löscht (remove) die Datei <file>



# Unix-Befehle für Dateien

<file> sei der Name einer Datei (ggf. inkl. Pfad)

- mv <file> <neu>
  - ▶ benennt die Datei <file> in die Datei <neu> um
  
- Für Textdateien:
  - ▶ Anzeigen des Inhalts: cat <Dateiname>
  - ▶ Erzeugen einer neuen Datei und Änderungen ihres Inhaltes : Editor

# Struktur des Dateisystems

- Baumartig (hierarchisch) organisiert: Verzeichnisse (directories) können Dateien und auch wieder Verzeichnisse enthalten
- Das **root**-Verzeichnis ist das Verzeichnis, das alle anderen enthält
- Das **home**-Verzeichnis (home-directory) ist das Verzeichnis, das alle Dateien und Verzeichnisse des entsprechenden Benutzers enthält.
  - ▶ `~testv` ist das home-Verzeichnis des Benutzers `testv`.
- Das Arbeitsverzeichnis (working directory) `'.'` ist das Verzeichnis, in dem man sich 'gerade befindet'.

- Der **Pfad** ist die Wegbeschreibung zu einer Datei
  - ▶ `/home/stud/s12/testv` ist der Pfad aller Dateien, die im Verzeichnis `testv` gespeichert sind.
  - ▶ Ein **Punkt** (`.`) bezeichnet stets das **aktuelle** Verzeichnis, i.e. Arbeitsverzeichnis.
  - ▶ Eine **Tilde** (`~`) bezeichnet stets das home-Verzeichnis.
  - ▶ `~<login-Name>` ist das home-Verzeichnis des Benutzers mit `<login-Name>`
  - ▶ **Zwei Punkte** (`..`) bezeichnen das übergeordnete Verzeichnis.

- enthalten ausführbare Programme
- Beispiel:
  - ▶ `/usr/local/X11/bin`  
enthält Befehle zum Aufruf von Anwendungsprogrammen, z.B. `acroread`
- `acroread` kann ohne Pfadangabe aufgerufen werden, da sein Pfad im `PATH` eingetragen ist.
- Alle Verzeichnisse, die in der `PATH`-Variablen angegeben sind, werden automatisch durchsucht.

- Problem:
  - ▶ Firefox kennt den Pfad zu acroread nicht automatisch!  
In Firefox werden pdf-Dateien nicht automatisch angezeigt, da hierzu der Acrobat Reader benötigt wird. Firefox fragt nach einem Programm zum Anzeigen von pdf-Dateien.
  - ▶ In Firefox muss der auszuführende Befehl **mit korrektem Pfad** eingegeben werden.
  
- Finden des Pfades für einen Befehl:  
Befehl `which <befehl>`  
`which acroread` liefert `/usr/local/X11/bin/acroread`

- Aktuelles Arbeitsverzeichnis ausgeben:
  - ▶ `pwd` (*print working directory*)
  
- Anlegen eines neuen Verzeichnisses
  - ▶ `mkdir <Verzeichnisname>`  
*legt ein Verzeichnis mit dem angegebenen Namen an.*
  
- Löschen eines Verzeichnisses
  - ▶ `rmdir <Verzeichnisname>`  
*löscht das Verzeichnis mit dem angegebenen Namen. Das zu löschende Verzeichnis darf keine Datei enthalten.*

Wechseln des Verzeichnisses: **cd**

Sei JavaPK das aktuelle Arbeitsverzeichnis

- Mit *cd Aufgabe1* wechselt man ins Verzeichnis Aufgabe1
- Mit *cd ..* wechselt man ins übergeordnete Verzeichnis, d.h. in das Verzeichnis testv
- Mit *cd ../AD1* wechselt man von Verzeichnis JavaPK in das Verzeichnis AD1

- Programm um Dateiinhalte zu durchsuchen.
- Die wichtigsten Switches sind:
  - ▶ -i für caseinsensitiv
  - ▶ -R für rekursiv
  - ▶ -n für die Zeilennummer
- Beispiel: `grep -i icewm .xsession`  
liefert: *37:icewm*



# ssh (secure shell)

- SSH ist ein Protokoll zur Fernadministration. Im Gegensatz zu z.B. Telnet wird die Verbindung verschlüsselt.
- Es eignet sich wunderbar um in der Arbi von zuhause aus zu Arbeiten.
- GUI Anwendungen können per X-Forwarding von anderen Rechnern aus gestartet werden. (switch -X)
- Beispiel: `ssh -X ersti@duemmer.informatik.uni-oldenburg.de`

# scp (secure copy)

- SCP ist ein Tool um per ssh Dateien zu übertragen.
  
- Beispiel: `scp testfile duemmer:/tmp`

- Drucker in der ARBI
- Druckquotas
- Die Druckerwarteschlange
- Starten eines Druckauftrags
- Zustand des Druckers
- Abbrechen/Löschen eines Druckauftrags

- Alle Drucker der ARBI haben einen Namen, der mit lw beginnt: lwblau, lwrot, lwlila, lworange, . . .
- Welcher Drucker wird verwendet?
  - ▶ Befehl `lpq` zeigt Informationen über den voreingestellten Drucker (Default-Drucker) an, d.h.  
in welchem Raum der Drucker steht,  
ob es Probleme mit dem Drucker gibt und  
welche Druckaufträge in der Drucker-Queue stehen und  
welche ID-Nummern diese Druckaufträge haben.
- Der Befehl `lpq -P<Druckername>` zeigt diese Informationen über den Drucker mit dem entsprechenden Namen an.

- Alle Drucker sind Postscript-Drucker nur Dateien im Postscript-Format können gedruckt werden.
- Druckbefehle für Postscript-Dateien
  - ▶ `lpr -P <Druckername> <Dateiname>`
- Unterschiedliche Dateiformate in unterschiedlichen Anwendungsprogrammen:
  - ▶ Text-Dateien (ASCII-Dateien) nedit, pine, ...
  - ▶ pdf-Dateien Acrobat-Reader
  - ▶ HTML-Dateien Mozilla

- Oft ist es nicht nötig Dateien 'manuell' in das Postscript-Format umzuwandeln
- Man muss für diese Umwandlung nicht unbedingt eine grafische Oberfläche bemühen
- Druckbefehle für (fast) alle Dateien
  - ▶ `a2ps -P <Druckername> <Dateiname>`
- a2ps wandelte eigentlich alle textbasierten Dateiformate (und PDF) in das Postscript-Format um und druckt es
- Formate wie DOC, DOCX, ODT usw. können so leider nicht gedruckt werden. In diesen Fällen muss erst in PDF umgewandelt werden.

- Nach dem Eingeben des Druckbefehls:
  - ▶ Der Druckauftrag wird in die **Warteschlange (queue)** des Drucker <Druckername> eingetragen.
  - ▶ Erst nach Abarbeitung aller früheren Druckaufträge der Warteschlange wird der Druckauftrag ausgeführt.
  
- Abfragen der Druckerwarteschlange:
  - ▶ `lpq -P <Druckername>`

# Wieviel darf ich drucken?

- Für jeden Student wird ein **Druckquota** von maximal 50 Seiten verwaltet, d.h. es dürfen maximal 50 Seiten pro Tag gedruckt werden.
- Bei jedem Druckauftrag wird das Druckquota entsprechend erniedrigt.
- Das Druckquote wird jeden Tag um 5 Seiten erhöht, bleibt aber stets unter 50.
- Man erhält per EMail eine Benachrichtigung über sein Druckquota-Konto.



- Solange ein Druckjob noch in der Queue steht, kann man ihn löschen durch

*lprm -P<Druckername> <jobid>*

- ▶ *<jobid>* ist die ID-Nummer des zu löschenden Druckauftrags.
- ▶ Weitere Möglichkeiten, den zu löschenden Auftrag zu beschreiben, sind in den man-pages erklärt.

# Wieviel Platz hab ich eigentlich?

- Jeder Benutzer hat nur begrenzten Speicherplatz zur Verfügung.
- Wieviel Platz dies ist ändert sich regelmässig.
- Je mehr Backup-Kapazität zur Verfügung steht desto mehr Speicher steht einem zur Verfügung.
- *quota -v* gibt Auskunft zum momentanen Stand.

# Ein- und Ausgabeströme

- Programmaufrufe produzieren in der Regel Ausgaben
- Oft ist es nützlich diese weiterzuverarbeiten oder zu speichern.
- 3 wichtige Operatoren: `>`, `<`, `|`
- `>`: ist die sog. Outputpipe: `cat file > file2`
- `<`: die Inputpipe: `mysql -p < table.sql`
- `|`: Weiterleitung von Ausgaben:  
`cat file | grep pattern`

# HILFE!!! Help!

Viele Möglichkeiten, Hilfe zu erhalten:

- man-pages aufrufen → Erläuterungen zu Unix-Befehlen
- Kommilitoninnen und Kommilitonen fragen
- TutorInnen löchern, z.B. die Tutoren des Programmierkurses
- In Internet und Newsgroups informieren
- Mitarbeiter der ARBI fragen

# Unix zum Ausprobieren. . .

- Wer das hier Gelernte zu Hause ausprobieren möchte, sich aber kein BSD- oder GNU/Linux-System installieren möchte, findet unter *<http://www.ubuntuusers.de>* eine Zusammenstellung von GNU/Linux-Software, komplett von CD lauffähig.