Karlsruher Institut für Technologie

Rechnernutzung in der Physik

Empfohlene Arbeitsumgebung

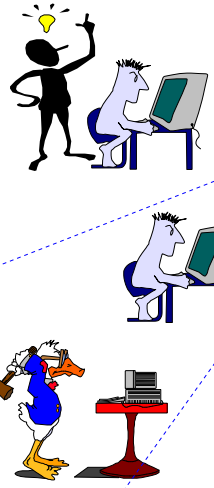
Prof. G. Quast

Fakultät für Physik
Institut für Experimentelle Teilchenphysik

WS 20212/23

Rechnernutzung in der Wissenschaft

Datenerfassung



Einzelplatz-Rechner als Schnittstelle zu PhysikerIn

Server-Rack

Moderne Wissenschaft ohne Computer nicht denkbar

CERN Computer Centre

Worldwide LHC computing Grid

Weltumspannende Netzwerke: Grid(s) und Clouds(s)

Rechnerhardware

Rechner in der Physik heute meist auf PC-Architektur *(PC=Personal Computer)*
d.h. **Mikrocomputersysteme** bestehend aus:

- Mikroprozessor(en)
- (flüchtigem) Halbleiter-Speicher
- permanentem Magnet-Festplattenspeicher oder „SolidStateDisk“ (SSD)
- Ein-Ausgabeschnittstellen
- Tastatur, Bildschirm, Maus, Wechsellaufwerke, Drucker, Netzwerk, ...

Verfügbar als

- Einzelplatz-Desktop-PC / Notebook
- vernetzter Cluster aus Desktop-PCs
- PC-Farm mit Fileservern
- Multi-Prozessor-Installationen für „High-Performance-Computing“
- vernetzte Rechenzentren („GRID“ & „Cloud“)

Für diesen Kurs (und Ihr gesamtes weiteres Studium) ist ein **eigenes *Notebook*** ideal:

- ≥2 GB Speicher
- ≥25 GB freier Plattenplatz
- ≥ core i3 oder Ryzen3

andernfalls: PCs im CIP-Pool

Empfohlene Software-Grundausstattung

Software-Pakete sind quelloffen (Open Source)
d.h. frei unter Linux, MS Windows, Mac OSX und auf dem CIP-Pool verfügbar

- Textdokumente erstellen mit LaTeX — siehe z.B. http://www.dante.de/
- (Vektor-)Grafik mit inkscape — https://inkscape.org/de/
- Script- und (Programmier-) Sprache python vers. 3
  unter Linux (meist) schon installiert
  für Windows siehe z.B. http://winpython.sourceforge.net/
  (*Komplettpaket, enthält schon die unten angegebenen Zusatzpakete*)

Interaktive Umgebung für *Python* — *jupyter Notebooks*

grafische Darstellung mit *Python*: matplotlib

Python-Bibliotheken für wissenschaftliches Arbeiten: numpy und scipy

- Funktionsanpassung mit *Python* *kafe2* https://github.com/PhiLFitters/kafe2
- Tools für Praktikua *PhyPraKit* https://gitlab.kit.edu/guenter.quast/phyprakit

Empfehlung: für diesen Kurs ***jupyter*** **Notebooks des SCC** nutzen:
https://uc2-jupyter.scc.kit.edu/

Anleitung und Tutorials http://ekpwww.etp.kit.edu/~quast/jupyter/

Details s. a. VL CgDA https://comp.physik.kit.edu/Lehre/CGDA/

Ihre Persönliche Arbeitsumgebung

einfache Lösung: Jupyter-Server des SCC im Browser nutzen
siehe Link https://uc2-jupyter.scc.kit.edu u.
Information auf den **Folien v. Robin Hofsaess**

für Linux-Nutzer: Jupyter auf eigenem System installieren,
s. Link https://jupyter.org/install

Unter Linux ist Python immer dabei.
Mit Hilfe des Paket-Managers sollte eine Version 3.7 oder höher installiert werden (aktuell: Version 3.11).
Jupyter installiert man dann einfach mittels

```
> pip install jupyterlab
```

für Windows-Nutzer: Python und Jupyter auf eigenem System installieren

- Python 3.10 aus dem Windows App Store
- oder WinPython (läuft auch ohne Installation, z.B. von einem externen Speicher).
- oder anaconda, eine auf allen Plattformen verfügbare Umgebung für Datenanalysen

docker-Container zum Kurs lokal laufen lassen,

s. Link http://etp.kit.edu/~quast/Docker/DockerContainers.html

Zugang zu Jupyter auf bwUniCluster2

Für diesen Kurs haben wir Ressourcen auf dem bwUniCluster2 des KIT reserviert

Registrierung nach Anmeldung für diesen Kurs in ILIAS durch das Tutoren-Team !

Wichtige Punkte:

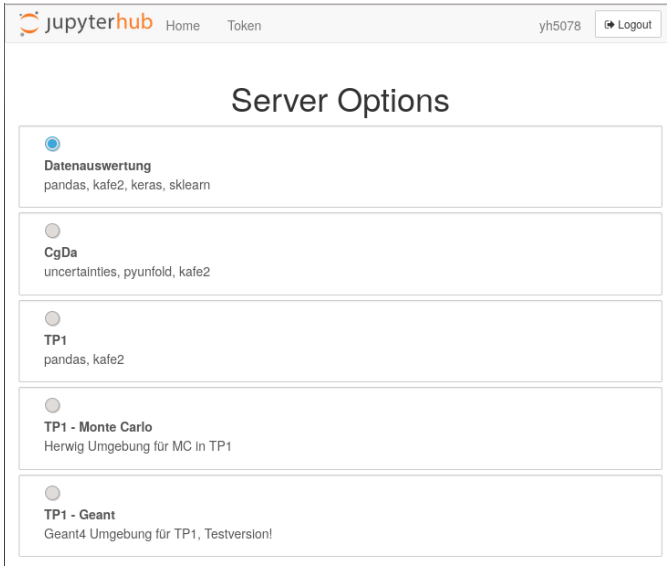
- Zugang nur aus dem KIT Netz bzw. über eine **VPN-Verbindung** (wie schon für Zugang zum Campus-System notwendig)
- und **Zwei-Faktor-Authentifizierung** („2FA“):
- erzeugt für jeden Login ein Einmal-Passwort („Token“), das von einem Endgerät in Ihrem Besitz (z.B. Handy) erzeugt wird und und das Sie jedes Mal zusätzlich zur Nutzername und Passwort angeben müssen

Detail s. Folien von Robin Hofsaess

Im Fall von Problemen gibt es ein **spezielles Beratungstutorium** am 7. Nov. , 15:00 im CIP Pool

Arbeiten mit *jupyter*-Notebooks

https://jupytermachine.etp.kit.edu



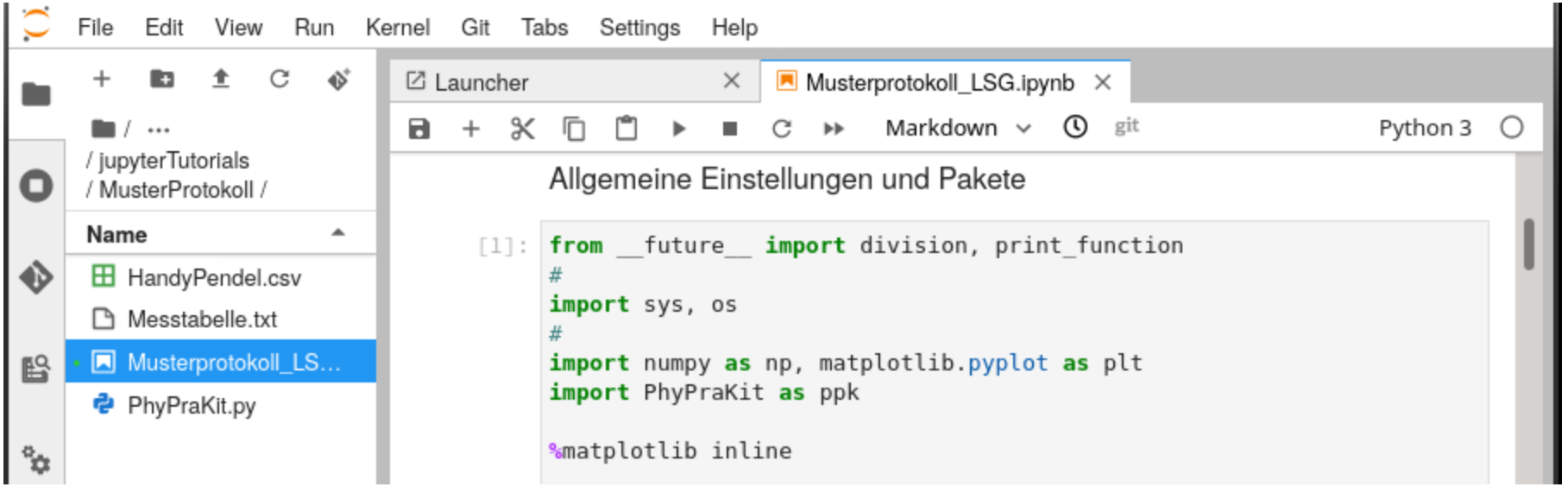
Voraussetzung zur **Nutzung des *jupyter*-Servers**:

- KIT-Account und CIP-Pool-Zugang
  s. https://comp.physik.kit.edu/Account/

Zahlreiche Tutorials:

http://etp.kit.edu/~quast/jupyter/jupyterTutorial.html

- Spickzettel für *jupyter* : JupyterCheatsheet.ipynb
- Einführung in *Python* : PythonIntro.ipynb
- Spickzettel für *Python* und *matplotlib*
  – PythonCheatsheet.ipynb
  – matplotlibTutorial.ipynb
- Grundlagen der Statistik: IntroStatistik.ipynb
- Fehlerrechnung im Physikalischen Praktikum:
  Fehlerrechnung.ipynb



Betriebssystem für Wissenschaftliches Arbeiten

Spezielle Empfehlung: Nutzen Sie **Linux** für wissenschaftliches Arbeiten !

Linux ist eine Re-Implementierung (Open Source) von Unix
(begonnen Anfang der 90er von Linus Torvalds)

LINUX ist (wie UNIX) ein portables (in C programmiertes), recht einfach aufgebautes Betriebssystem, mit folgenden wesentlichen Eigenschaften:

- Multitasking (Prozess-Scheduling, Prozess-Kommunikation)
- Multiuser (Zugangskontrolle und Abrechnung)
- dialogorientiert
- ein Werkzeugkasten mit viele hundert Dienstprogrammen
- (mehrere) Shell(s) mit mächtiger Script-Sprache;
  Scripte laufen als Unterprozesse des startenden Prozesses
- Dateisystem mit Baum-artiger Struktur, Rechtekontrolle für Datei-Zugriff
- graphische Oberfläche X-Windows (X11)
- volle Netzwerkunterstützung, incl. Netzwerk-Laufwerken, remote shells etc.

Linux bietet viele Vorteile – auch unter Windows !

Mit WSL (Windows Subsystem for Linux) ist der Einstieg leicht *Probieren Sie es aus !*

Für Windows-Nutzer

Unter Windows tut sich gerade viel,
und Sie haben einige Möglichkeiten,
von openSource-Entwicklungen zu profitieren:

Windows 10

Windows 11

- *Python* und *Jupyter* unter Windows installieren
  aus dem Microsoft Store

oder WinPython

- jupyter-Server in Docker-Container laufen lassen

- Windows Subsystem for Linux (wsl) nutzen -
  ist bei neueren Windows-Versionen dabei !

Material zur nächsten Vorlesung

- Arbeitsumgebung vorbereiten (s. Präsentation von R.Hofsaess)
- Stoff der folgenden Jupyter-Tutorials kennen

  – JupyterCheatsheet.ipynb

  – jupyterTutorials.ipynb

  – PythonCheatsheet.ipynb

  – PythonIntro.ipynb

  – matplotlibTutorial.ipynb

- Soff der nächsten Vorlesung:

  V02a_ZusFasCgDA.pdf