

Programmierkonzepte in der Physikalischen Chemie

1. Motivation Inhalte und Struktur des Kurses

Albert-Ludwigs-Universität Freiburg

Dr. Till Biskup

Institut für Physikalische Chemie
Albert-Ludwigs-Universität Freiburg
Wintersemester 2013/14



**UNI
FREIBURG**

Motivation

- Wiederverwertbaren Code schreiben
- Möglichkeiten der Programmierung ausnutzen

Inhalte und Struktur des Kurses

- Grundlegende Konzepte an realen Beispielen
- Matlab als Sprache der Wahl
- Frontal statt „hands on“
- Freiwillige Beispielaufgaben

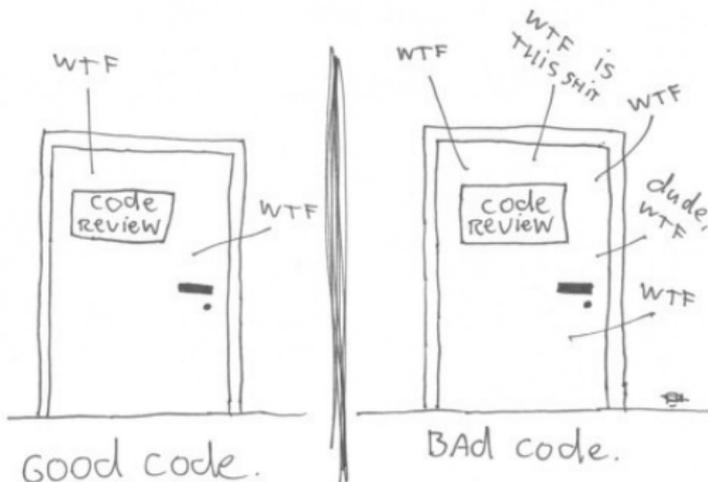
Zielgruppe

Motivation

Warum ein Methodenkurs „Programmierkonzepte“?



The ONLY valid measurement
OF code QUALITY: WTFs/minute



(c) 2008 Focus Shift/OSNews/Thom Holwerda - <http://www.osnews.com/comics>

Hilfe zur Selbsthilfe

- ▶ Gerade in der Physikalischen Chemie hat man in der Regel mit gemessenen Daten und deren Verarbeitung zu tun.
- ▶ Da die Verarbeitung meist im Computer erfolgt, ist Programmierung ein wesentlicher Bestandteil.
- ▶ Oft fehlt den Beteiligten aber das Wissen über allgemeine Programmierkonzepte, die das Leben mitunter sehr vereinfachen können.
- ☞ Vermittlung allgemeiner Programmierkonzepte, die sich universell einsetzen lassen.

Minimale Qualitätsstandards sicherstellen

- ▶ Diejenigen, die Programme schreiben, sind meist nicht langfristig in Arbeitskreisen (in der Regel maximal für die Dauer einer Doktorarbeit).
- ▶ Der jeweilige Arbeitskreis hat ein Interesse, ein Minimum an **Qualitätsstandard** bei den Programmen sicherzustellen.
- ▶ Nur so ist es möglich, daß andere die Programme übernehmen und weiterpflegen.
- ☞ Code wird in der Praxis viel zu häufig neu geschrieben – aufgrund mangelnder Wiederverwertbarkeit.

Qualitätsstandards

- ▶ Qualität der Dokumentation
 - ▶ Robustheit des Codes
 - ▶ Übersichtlichkeit des Codes
- ☞ Jeder dieser Aspekte wird in den folgenden Stunden ausführlicher beleuchtet werden.

Grundlegend zwei Konzepte:

- 1 ein Skript für jeden Datensatz
- 2 eine Toolbox aus Funktionen, die generisch jeden Datensatz verarbeiten kann

☞ Beide Konzepte haben Vor- und Nachteile.

Skripte – Vorteile

- ▶ Ein Skript enthält (idealerweise) alle Befehle für die Verarbeitung bis zur fertigen Abbildung.
- ▶ Das Skript liegt bei den Rohdaten.
- ▶ Auch nach langer Zeit kann die identische Abbildung durch einen einzigen Befehl neu generiert werden.
- ▶ Es ist jederzeit vollständig nachvollziehbar, wie die Rohdaten verarbeitet wurden.
- ▶ Ideal für eine einzelne, schnelle Auswertung

Skripte – Nachteile

- ▶ viele Doppelungen
 - gleicher Code wird viele Male geschrieben/kopiert
 - Änderungen in einem Skript müssen manuell in andere Skripte eingepflegt werden, die die gleiche Aufgabe durchführen.
- ▶ unübersichtlich
 - Welches Skript war das neueste?

Toolbox – Vorteile

- ▶ Code wird nur einmal geschrieben.
DRY — *Don't Repeat Yourself*
- ▶ Bei modularem Aufbau sehr flexibel
- ▶ Bietet die Möglichkeit einfach zu bedienender (graphischer) Nutzerschnittstellen
- ▶ Sehr gut geeignet für sich häufig wiederholende gleiche oder ähnliche Auswertungen

Toolbox – Nachteile

- ▶ Verarbeitungsschritte der Rohdaten aus den verarbeiteten Daten nicht ohne Weiteres ersichtlich
 - ▶ Verlust der Reproduzierbarkeit:
Durch Veränderungen einzelner Funktionen läßt sich eine Abbildung ggf. nach Jahren nicht mehr reproduzieren.
 - ▶ Trennung von Toolbox und Rohdaten
Zur nochmaligen Auswertung muß immer auch die Toolbox installiert werden.
- ☞ (Fast) alle Nachteile lassen sich beheben.

Grundlegend zwei Konzepte:

- 1 ein Skript für jeden Datensatz
 - 2 eine Toolbox aus Funktionen, die generisch jeden Datensatz verarbeiten kann
- ☞ Im Rahmen dieses Methodenkurses wird das Konzept der **Toolbox** vorgestellt und erläutert werden.

Wiederverwertbar heißt auch:

- ▶ verständlich, gut dokumentiert, übersichtlich
 - ▶ Eine andere Person kann die Pflege des Quellcodes übernehmen.
-
- ☞ Die Begriffe „verständlich“, „gut dokumentiert“ und „übersichtlich“ werden im Rahmen dieses Methodenkurses detaillierter behandelt werden.
 - ☞ Code wird in der Praxis viel zu häufig neu geschrieben – aufgrund mangelnder Wiederverwertbarkeit.

Programmierung bietet viele Möglichkeiten, u.a.:

- ▶ Automatisierung
- ▶ Modularisierung

☛ Automatisierung und Modularisierung sprechen beide für das Konzept der Toolbox.

Möglichkeiten der Programmierung ausnutzen

Automatisierung

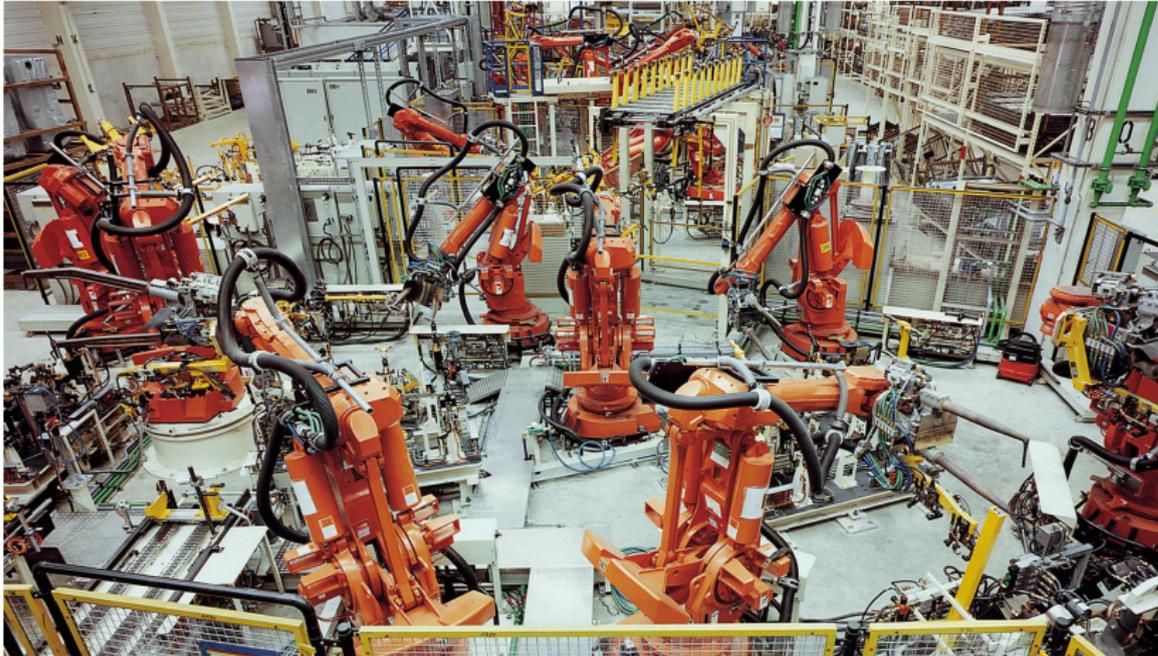


ABB via <http://www.ingenieur.de/>

Satz

Computer sind dumm, aber fleißig.

Satz

Wissenschaftler sind intelligent, aber faul.

☞ **Automatisierung** ist das Gebot der Stunde.

Beispiele, was sich gut automatisieren läßt:

- ▶ regelmäßig wiederkehrende Aufgaben
 - Einlesen von Daten
 - immer gleiche Prozessierungsschritte

- ▶ Dokumentation der Prozessierung von Daten
 - Jeder einzelne Verarbeitungsschritt der Daten wird automatisch mitgeschrieben.
 - Ein gut geführtes Laborbuch enthält genau das...
aber: Menschen tendieren zu Faulheit. (s.o.)

- 👉 Der Methodenkurs wird viele konkrete Beispiele für die Automatisierung liefern.

Möglichkeiten der Programmierung ausnutzen

Modularisierung



- ▶ Unix-Prinzip: eine Aufgabe, eine Funktion
- ▶ Strikte Trennung zwischen Datenverarbeitung und Nutzerschnittstelle
- ▶ Code wird nur einmal geschrieben.
DRY — *Don't Repeat Yourself*
- ☞ Toolboxen erhalten ihre Flexibilität und Qualität durch Modularisierung.
- ☞ Das Thema „Modularität“ wird nächste Woche ausführlich behandelt werden.

Methodenkurs „Programmierkonzepte in der Physikalischen Chemie“

Einführung grundlegender Programmierkonzepte
am Beispiel von Matlab



Inhalte und Struktur

- ▶ Grundlegende Programmierkonzepte an realen Beispielen aus der Praxis einführen
- ▶ **Matlab** als Programmiersprache der Wahl (für diesen Kurs)
- ▶ Frontal statt „hands on“
- ▶ Zu jedem Thema gibt es vertiefende Aufgaben, deren Bearbeitung freiwillig ist.

Voraussetzungen

- ▶ Grundkenntnisse im Programmieren
- ▶ Grundkenntnisse in der Bedienung von Matlab

- ▶ Modularität
- ▶ Dokumentation (im Code)
- ▶ Infrastruktur rund um die Entwicklung von Code (Versionierung, Bug-Verwaltung, Dokumentation)
- ▶ Robuster Code
- ▶ Schneller Code
- ▶ Verarbeitung von Daten
- ▶ Nutzerschnittstellen (CLI, GUI)
- ▶ Toolboxen
- ▶ Programmentwicklung (Versionsnummern, Meilensteine, Projektdokumentation)

Gründe für die Wahl von Matlab

- ▶ Matlab ist ein *de-facto*-Standard zur Datenverarbeitung in den Natur- und Ingenieurwissenschaften.
- ▶ Matlab ist über eine Landeslizenz (in Ba-Wü) verfügbar.
- ▶ Matlab eignet sich gut zur Verarbeitung und Auswertung von Daten in der Physikalischen Chemie.
- ▶ Matlab bietet einfache und schnelle Möglichkeiten zur Darstellung von Daten.
- ▶ Die u.a. an C angelehnte Programmiersprache ist einfach zu erlernen und sehr mächtig.

Anmerkungen

- ▶ Matlab ist nicht allein glücklich machend.
- ▶ Matlab ist ein kommerzielles Produkt.
 - Eine nichtakademische Lizenz ist sehr teuer.
 - Die Landeslizenz ist zeitlich befristet.
 - Sparsamer Umgang mit kommerziellen Toolboxes ist deshalb das Gebot der Stunde!
- ▶ Programmierkonzepte lassen sich nur sinnvoll anhand einer konkreten Programmiersprache vorstellen.

Frontal statt „hands on“

Vorstellung von Konzepten zur eigenen konkreten Anwendung

- ▶ Viele Konzepte sind zunächst einmal abstrakt.
- ▶ Reale Beispiele aus eigener Erfahrung werden die Konzepte erläutern und vertiefen.
- ▶ Die Umsetzung der Konzepte ist in vieler Hinsicht eine Frage der persönlichen Disziplin.
- ▶ Freiwillige Beispielaufgaben dienen der Vertiefung.
- ☛ Rückfragen und die Diskussion an konkreten Beispielen aus dem Publikum sind natürlich erwünscht.



- ▶ Zu jedem Thema gibt es vertiefende Aufgaben.
- ▶ Die Bearbeitung ist rein freiwillig.
- ▶ Viele Wege führen nach Rom:
Es gibt meist nicht *die* eine Lösung.
- ▶ Rückfragen sind natürlich jederzeit möglich.
- ▶ Auf Wunsch kurze Besprechung der Aufgaben
anhand der Musterlösungen jeweils in der Folgewoche.

Homepage zur Veranstaltung

<http://till-biskup.de/de/lehre/programmierkonzepte/>



- ▶ Doktoranden und Studenten in höheren Semestern mit Grundkenntnissen im Programmieren
- ▶ Grundkenntnisse im Programmieren sind Voraussetzung, Interesse an konzeptionellen Überlegungen erwünscht.
- ☛ Jeder, der größere Auswertungen mit dem Computer durchführen möchte/muss und dabei keine „black boxes“ verwenden möchte.



Einführender Methodenkurs

„Anwendung von Mathematica und Matlab
in der Physikalischen Chemie“

Datum 19.–20.02.2014

Dozenten Stefan Weber, Till Biskup

Inhalte Einführung in die beiden Programmpakete,
Verwendung in der Physikalischen Chemie

So long, and thanks for all the fish.

Vorschau: [Allgemeines](#)

- ▶ „Kenne Deine Programmiersprache“
- ▶ richtiger/sinnvoller Editor
- ▶ klare und sinnvolle Verzeichnisstrukturen
- ▶ „Matlab in fünf Minuten“