



WESTFÄLISCHE  
WILHELMS-UNIVERSITÄT  
MÜNSTER

# > Implementierung paralleler Programmiermuster auf modernen many-core Systemen in OpenCL

Projektseminar im WiSe 2014 / 2015

Prof. Sergei Gorlatch, Ari Rasch, Michael Haidl



## Heutige Ziele:

- Besprechung organisatorischer Dinge
- Vorstellung der Themen für die Einführungsvorträge (1. VL-Woche)
- Fragen beantworten



## Was ist unser Ziel?

Umsetzung typischer **paralleler Programmiermuster** auf modernen **many-core Systemen**, wie Beispielsweise der Nvidia Tesla K20 oder der Intel Xeon Phi, mit **OpenCL**



- Theoretische und praxisorientierte Phasen
  - Theorie:  
Betrachtung wissenschaftlicher Literatur: analysieren, vergleichen, bewerten
  - Praxis:  
Anwendung und Umsetzung der erlernten Theorie
- Zusammenarbeit und Entwicklung in wechselnden Teams



- Projektraum
  - Bietet Möglichkeit sich dort zum gemeinsamen Arbeiten zu treffen
  - 24/7 zur freien Verfügung, gern selbst ausstatten (Kaffee!)
- Versionsverwaltung (Git)
- trac: Wiki (+ Git Anbindung) zur Projektplanung, Koordination
- Mailingliste
- Weitere Wünsche?



- (Teil-)Aufgaben werden von uns vorgegeben (→ trac)
- Bearbeitung und Verteilung organisiert ihr selbst
- Verantwortlich sind grundsätzlich alle
- Kommt bei Fragen und Problemen (rechtzeitig!) zu uns



- Bei den wöchentlichen Treffen (**Dienstags 10 Uhr ct. SR7**) gibt jede Gruppe einen (kurzen) Zwischenbericht ab:
  - Im Allgemeinen knapp, evtl. auf ein bis zwei Folien
  - Was wurde bearbeitet (abgeschlossen oder Fortschritt, evtl. Probleme)?
  - Was steht als Nächstes an?
  - Demonstrationen!

Das Projekt soll durch 2 Präsentationen und einer schriftlichen Ausarbeitung vorgestellt werden:

- **Zwischenpräsentation**

Vorstellung der Projekt-Motivation und der Projekt-Ziele in Form eines Vortrags (ca. 30min) vor unserer Arbeitsgruppe

- **Abschlusspräsentation**

Vorstellung der Ziel-Umsetzung in Form eines Vortrags (ca. 30min) vor unserer Arbeitsgruppe

- **Ausarbeitung**

Zusammenfassung des Projekts (Motivation, Ziele, Umsetzung Evaluation) in schriftlicher Form





In die Bewertung fließen ein:

- Schriftliche Ausarbeitung
- Vorträge und Referate
- Code (inkl. Dokumentation)
- Engagement und Teamarbeit
  - insbesondere **aktive (!)** Teilnahme an den wöchentlichen Treffen



- 23.08.2014: Vorbesprechung
- 07.10.2014: Einführungsreferate
- ... Zwischenpräsentation
- ... Fertigstellung Software,  
Abschlussvortrag
- ... Fertigstellung der Ausarbeitung



- Ziele:
  - Den anderen Teilnehmern soll ein Einblick ins Thema vermittelt werden
  - **Vortragende sollen Experten/Ansprechpartner auf ihrem Themengebiet werden**
  - Möglichst praxisbezogene Darstellung: Demo, Pseudo-Code, etc.
  - Dauer: ca. 10 Minuten (= ca. 10 Folien)

## 1. OpenCL Programmiermodell + API

- Vorstellung der wichtigsten Konzepte von OpenCL: Host, Device, Kernel, ...
- Erläuterung der Ausführung eines OpenCL Kernels auf einem Device
- Erstellung und Vorführung eines einfachen Beispielprogramms in OpenCL

## 2. C++(0x)

- Objektorientiertes Design (Klassen, Vererbung, ...)
- Templates
- C++0x-Features (initializer lists, type inference, smart pointers, ...)

### 3. GPU-Architektur und device-spezifische Optimierungen

- Wie sieht die GPU-Architektur aus?
- Wie schreibt man optimierten Device-Code im Kontext von OpenCL?

### 4. CPU-Architektur und device-spezifische Optimierungen

- Wie sieht die CPU-Architektur aus?
- Wie schreibt man optimierten Device-Code im Kontext von OpenCL?

### 5. Intel Xeon Phi-Architektur und device-spezifische Optimierungen

- Wie sieht die Intel Xeon Phi-Architektur aus?
- Wie schreibt man optimierten Device-Code im Kontext von OpenCL?

## 6. Algorithmische Skelette

- Was sind algorithmische Skelette? Warum wird dadurch die Programmierung vereinfacht?
- Überblick über übliche Skelette
- Wie werden in SkelCL algorithmische Skelette verwendet?

## 7. SkelCL

- Was ist SkelCL?
- Wie werden in SkelCL algorithmische Skelette verwendet?
- Warum wird durch SkelCL paralleles Programmieren einfach?
- Technisches Design von SkelCL?

- 1 Jetzt: Priorisierung der Themen in Liste eintragen (alle Themen berücksichtigen!)
  - Zuordnung hier im Anschluss
- 2 Einarbeitung in zugewiesenes Referatsthema und Erarbeiten eines Kurzvortrags
  - Einarbeitung heißt nicht nur lesen, sondern auch **Erprobung in der Praxis**
- 3 Besprechung des Vortrags ca. eine Woche vor dem Vortragstermin
  - Folien per Email an Betreuer
  - Besprechung mit dem Betreuer
- 4 Vortrag am 07. Oktober 2014, 10:00 Uhr (s.t.) im SR 7



Weitere Informationen folgen auf

[http://www.uni-muenster.de/PVS/lehre/ws14/  
projektseminar.html](http://www.uni-muenster.de/PVS/lehre/ws14/projektseminar.html)