

Institut für Informatik  
Henning Wenke M.Sc.  
Sascha Kolodzey B.Sc.  
Nils Vollmer B.Sc.

Universität Osnabrück  
<http://www-lehre.inf.uos.de/~pa/>

## Übung: Parallele Algorithmen mit OpenCL

*Sommersemester 2013*

Blatt 6

### **Aufgabe 6.1 Trailvisualization: Datenlayout und Algorithmus (20 Punkte)**

Erklären Sie Ihrem Tutor das Datenlayout, Zugriffe auf bestimmte Daten und den Aufbau des in der Vorlesung vorgestellten Algorithmus zur Erzeugung eines Trails für jedes Partikel des N-Body Systems.

### **Aufgabe 6.2 N-Body System: Trailvisualization (80 Punkte)**

Implementieren Sie den in der Vorlesung vorgestellten Algorithmus zur Generierung eines Trails für jedes Partikel des N-Body Systems. Laden Sie sich hierzu das Framework [http://www-lehre.inf.uos.de/~pa/Uebungen/Blatt6/PA\\_Blatt6.zip](http://www-lehre.inf.uos.de/~pa/Uebungen/Blatt6/PA_Blatt6.zip) herunter und installieren Sie es in einer Entwicklungsumgebung.

Ihre Aufgabe besteht darin, die Kernel `passPositionOn` und `setTrailParticle` in der Klasse `NBodySimCurve` anzulegen und den dazugehörigen Sourcecode in der Datei `nbody.cl` zu vervollständigen. Benutzen Sie zur Implementierung die Informationen aus den Folien der Vorlesung. Vergessen Sie nicht die bereits zur Verfügung gestellten Daten korrekt als Argumente der beiden Kernel zu setzen, bevor Sie diese im dafür vorgesehenen Abschnitt der `run` Methode ausführen.

**Hinweis:** Die Positionen des N-Body Systems liegen nicht mehr doppelt im Speicher. Dies ist insoweit nicht mehr von Nöten, da die Simulation des N-Body Systems sich über zwei Kernel erstreckt. Im ersten Kernel `nBody_CalcNewV` wird lediglich die neue Geschwindigkeit berechnet und in einem zweiten, darauffolgenden Kernel `nBody_CalcNewPos` die neue Position. Somit kommt es zu keiner Inkonsistenz der neu berechneten Positionen.