

# Mehrprozessorenarchitekturen

(SMP, UMA/NUMA, Cluster)

Arian Bär

12.07.2004



# Gliederung

---

- 1. Einleitung
- 2. Symmetrische Multiprozessoren (SMP)
  - Allgemeines
  - Architektur
- 3. Speicherarchitekturen
  - Gemeinsamer Speicher (SM)
  - Verteilter Speicher (DSM)
- 4. Cluster Computer
- 5. Zusammenfassung



# Einleitung

---

- Ohne gehts nicht!
  - Google
  - Wettervorhersage
  - Große Webauftritte
  
- Deswegen:
  - Ein System aus vielen Prozessoren
  - Am besten: Anzahl der Prozessoren proportional zur Leistung



# SMP – Allgemeines

---

- Threads (Aktivitätsträger):
  - Mehrere Programme „gleichzeitig“ in Ausführung
  - Betriebssystem schaltet hin und her
  - Kernel-level Threads
  - User-level Threads
  - Ein Prozessor kann nur einen Thread bearbeiten
  - Umschalten kostet Zeit



# SMP – Allgemeines Fortsetzung 1

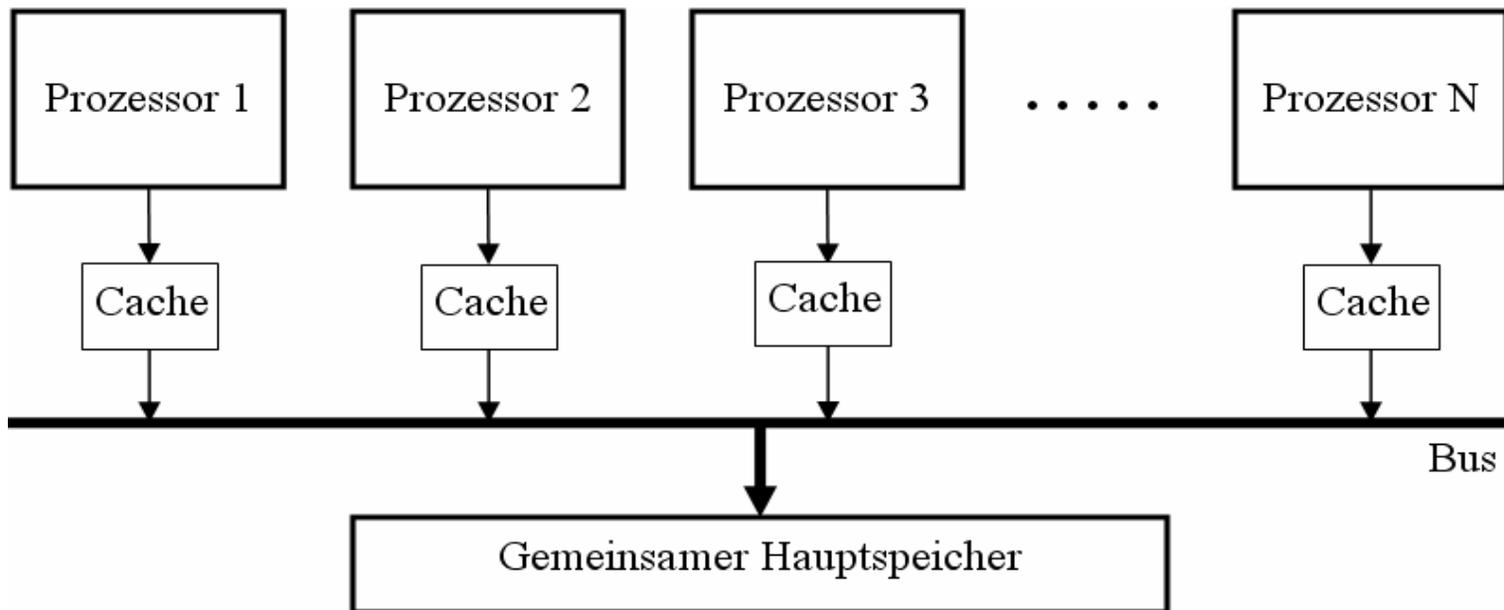
---

- Vorteile durch SMPs:
  - Weniger Contextwechsel
  - Mehrere Threads wirklich gleichzeitig in Ausführung
  - Leistungssteigerung
- Nachteile:
  - Performanceverlust durch Verwaltung:
    - Gemeinsame Variablen
    - Verteilung der Aufgaben
  - Aufwendigere Programmierung



# SMP – Architektur

- Mehrere gleiche Prozessoren
- Keine Verarbeitungshierarchie
- **Problem:** Cachekonsistenz



# SMP – Architektur Fortsetzung 1

---

- Einfache Lösung
  - Zuschreibende Variablen sind ‚*noncacheable*‘
  - Nur ‚*const*‘ Variablen im Cache
- **Problem:**
  - Weniger Cachetreffer
  - Muss vom Programmierer beachtet werden



# SMP – Architektur Fortsetzung 2

---

- ‚*snooping bus*‘
  - Prozessor A benötigt Variable
  - Sendet Anfrage an alle Prozessoren
  - Prozessoren senden Antworten zurück
  - Prozessor A wertet Antworten aus
  - Variable wird entweder:
    - Aus dem Cache des anderen Prozessors
    - oder aus dem Hauptspeicher geladen



# Gliederung

---

1. Einleitung
2. Symmetrische Multiprozessoren (SMP)
  - Allgemeines
  - Architektur
- **3. Speicherarchitekturen**
  - Gemeinsamer Speicher (shared memory - SM)
  - Verteilter Speicher (distributed SM - DSM)
4. Cluster Computer
5. Zusammenfassung



# Gemeinsamer Speicher

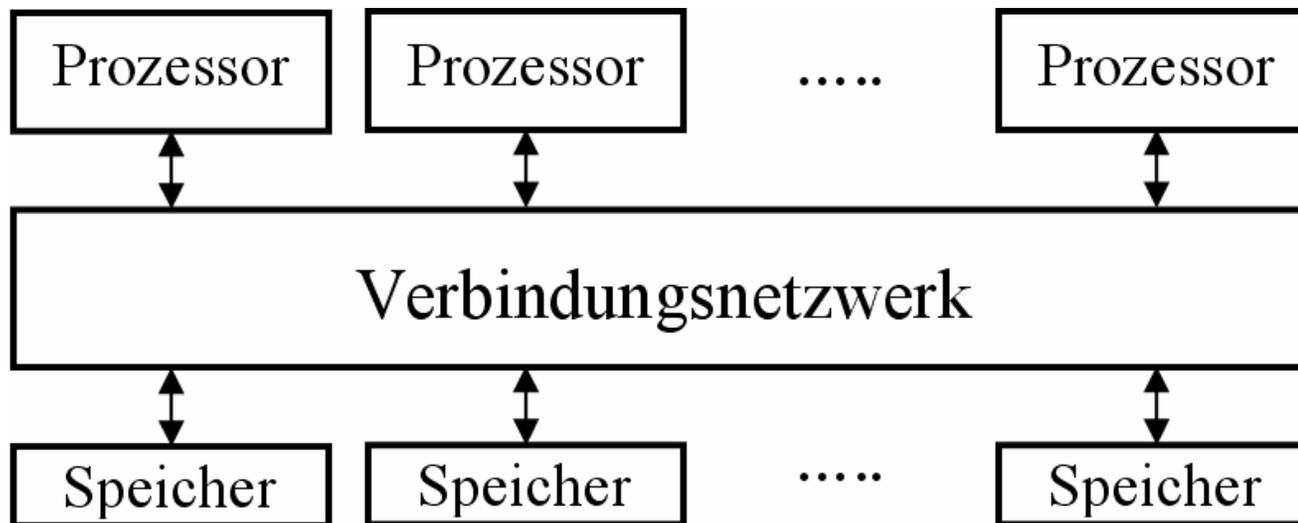
---

- Ein globaler Adressraum
- Zugriff über Verbindungsnetzwerk
- Drei unterschiedliche Klassen:
  - UMA – uniform memory access
  - NUMA – non uniform memory access
  - COMA – cache only memory architecture



# UMA – uniform memory access

- Alle Speicher Zugriffe gleich
- Tanzsaalarchitektur (dancehall architecture)
- Zugriffslatenz identisch
- **Flaschenhals:** Verbindungsnetzwerk



# UMA – Fortsetzung

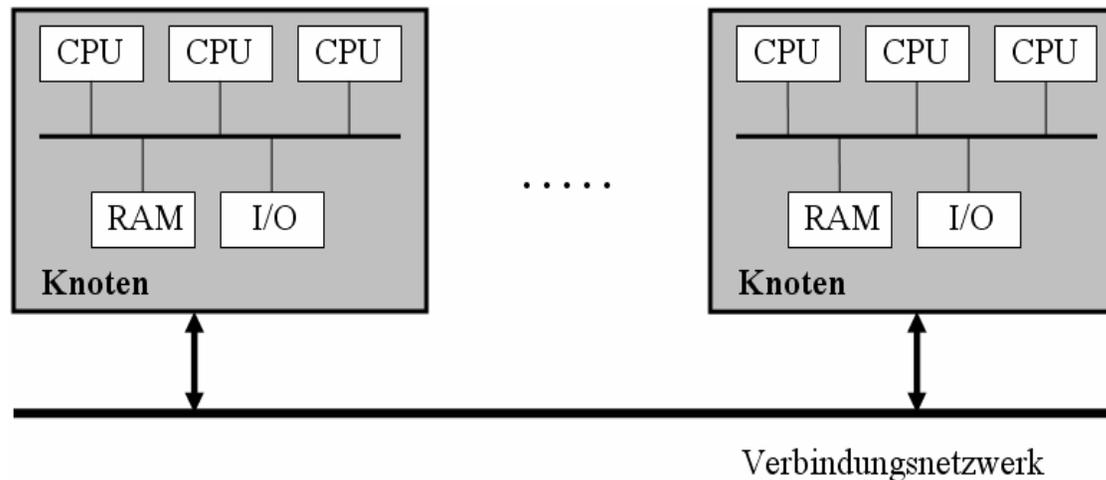
---

- Mehrprozessorboards setzen meist UMA ein
- Beispiel:
  - Intel® Server Board SE7525GP2
  - Bis zu zwei Prozessoren
  - Greifen auf einen Speicher-Controller zu
- UMA oft teil einer NUMA (später mehr)



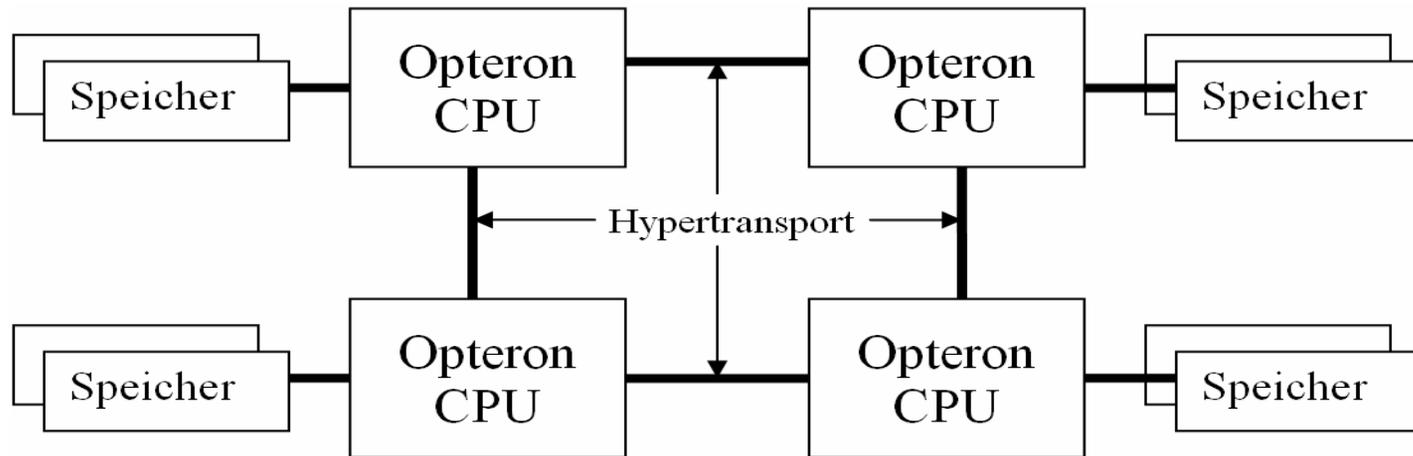
# NUMA – non uniform memory access

- Speicher wird Prozessoren lokal zugeordnet
- Knoten besteht aus CPU(s), I/O, RAM
- Latenzzeiten unterschiedlich
- Keine Vorhersage über Latenz möglich



# NUMA – Beispiel

- AMD® Opteron Prozessoren verwenden NUMA
- Jeder Prozessor hat eigenen Speicher
- Prozessoren verknüpft über Hypertransport
- Hypertransport: schnelle Verbindung (12.8 GB/s)



# COMA – cache only memory architecture

---

- Alle Speicher sind Cachespeicher
- Lokaler Speicher ist Cache
- Speicher anderer CPU's ist Hauptspeicher
- Zugriffe auf auf lokalen Speicher direkt
- Daten aus „Hauptspeicher“ werden erst gecachet
- **Problem:** auffinden von Daten im Speicher



# Verteilter Speicher

---

- Speichermodule sind Prozessoren zugeordnet
- Nur lokaler Speicher direkt adressierbar
- Zugriff auf andere Speicher über Nachrichten
- NORMA – no remote memory access
- Schnelle Anbindung an lokalen Speicher
- Verbindung untereinander langsamer als bei SM



# Gliederung

---

1. Einleitung
2. Symmetrische Multiprozessoren (SMP)
  - Allgemeines
  - Architektur
3. Speicherarchitekturen
  - Gemeinsamer Speicher (SM)
  - Verteilter Speicher (DSM)
- 4. Cluster Computer
5. Zusammenfassung



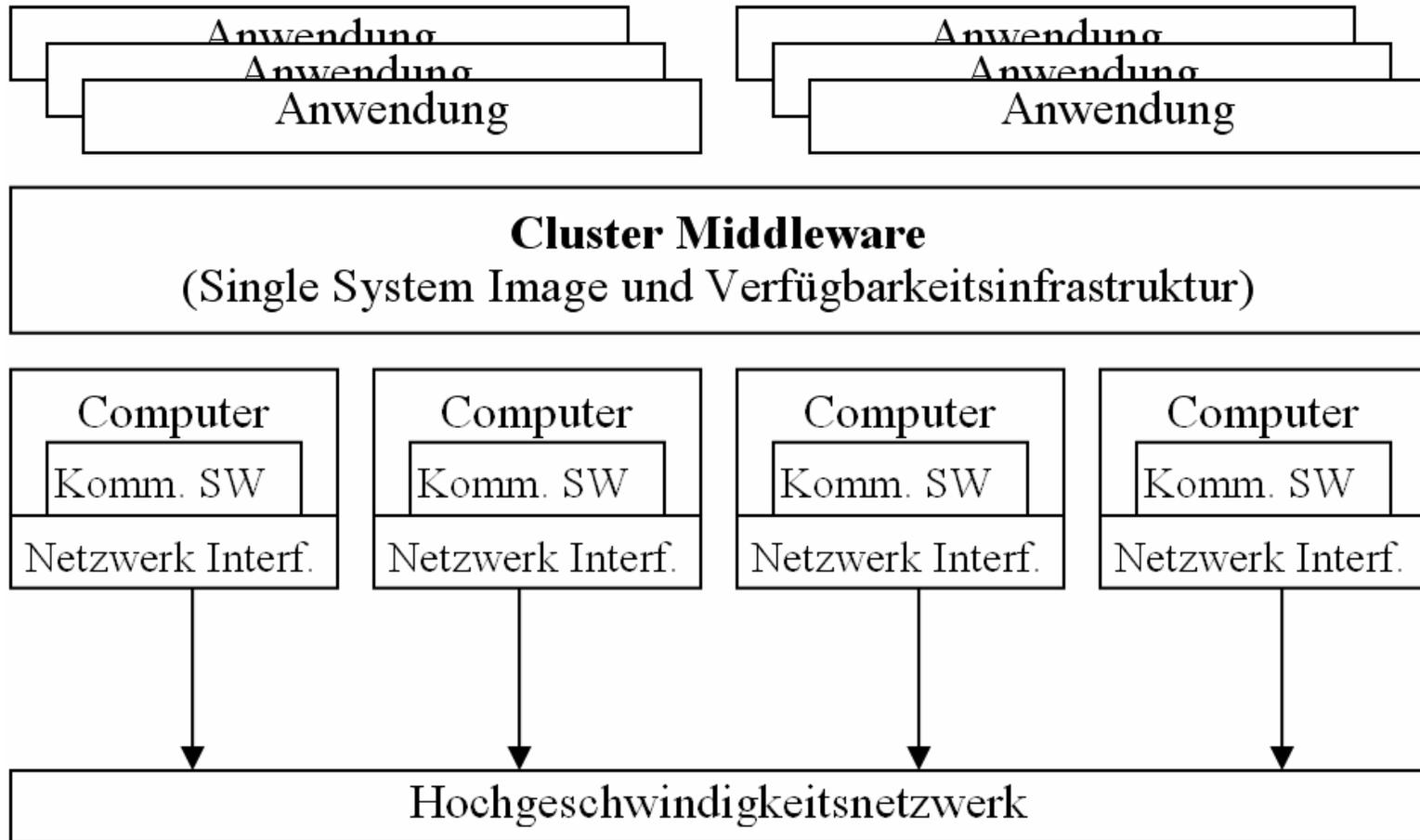
# Cluster Computer

---

- Netzwerk aus Knoten
- Knoten sind komplette PC's
- meist Gigabit Ethernet als Netzwerktechnologie
- SSI - *single system image*
- Einsatz z. B. Berechnung von 3D Filmen
- Kostengünstig



# Cluster Computer Fortsetzung



# Zusammenfassung

---

- SMPs
  - Jeder Prozessor erledigt jede Aufgabe
  - Ansätze auch in Heim-PC's (Hyper Threading)
- Speicherarchitekturen
  - Wege der Kommunikation
- Cluster Computer
  - Günstige Hochleistungsrechner



# Quellen

---

- Klaus Waldschmidt (Hrsg.): Parallelrechner Architekturen – Systeme – Werkzeuge, Stuttgart: Teubner, 1995.
- Rajkumar Buyya: High Performance Cluster Computing Architectures and Systems, Volume 1, Prentice Hall PTR Upper Saddle River, New Jersey 1999.
- George Coulouris, Jean Dollimore, Tim Kindberg: Verteilte Systeme Konzepte und Design, 3., überarbeitete Auflage, Pearson Education Limited, 2002 .
- Sequent Computer Systems, Ltd., [http://parallel.ru/ftp/computers/sequent/NUMA\\_SMP\\_REV.pdf](http://parallel.ru/ftp/computers/sequent/NUMA_SMP_REV.pdf), 16.05.2004
- Jörg Wirtgen/c't: AMD will NUMA-Architektur forcieren, <http://www.heise.de/newsticker/meldung/14776>, 16.05.2004
- Stefan Schumacher: Parallelrechner: Multiprozessorsysteme, <http://www.uni-magdeburg.de/steschum/multiprozessorsysteme.pdf>, 16.05.2004
- Intel® E7525 Memory Controller Hub (MCH) Chipset Datasheet, <ftp://download.intel.com/design/chipsets/datashts/30240501.pdf>, 07.07.2004
- AMD: Porting to AMD64 FAQ, [http://www.amd.com/us-en/assets/content\\_type/Additional/AMD64\\_Porting\\_FAQ.pdf](http://www.amd.com/us-en/assets/content_type/Additional/AMD64_Porting_FAQ.pdf), 07.07.2004
- Aad van der Steen: ccNUMA machines, <http://www.top500.org/ORSC/2003/ccNUMA.html>, 07.07.2004

