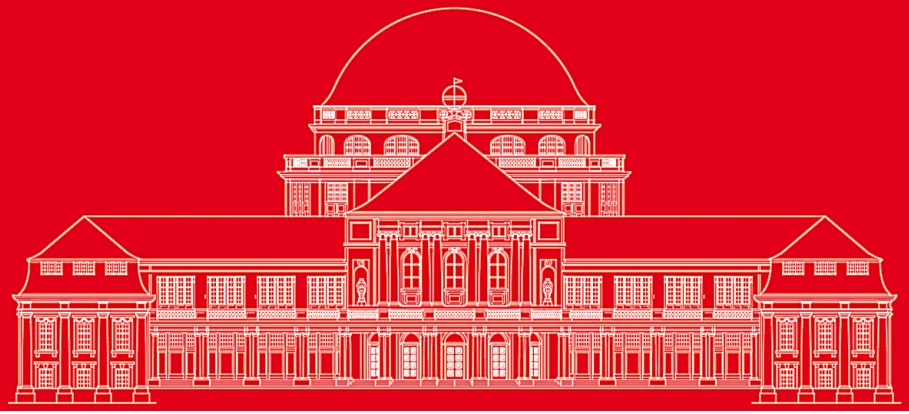
Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Praktikum: Paralleles Programmieren für Geowissenschaftler

Prof. Thomas Ludwig, Hermann Lenhart & Tim Jammer



Dr. Hermann-J. Lenhart
hermann.lenhart@zmaw.de

Einführung zum „Umfeld“ vom Paralleles Programmieren:

- Hardware Voraussetzung zur Parallelen Programmierung
- Softwareaspekte zum Parallelen Programmieren
- Modellstruktur zum Parallelen Programmieren
  (mit Blick auf MPI)

HPC Top500 Liste - Stand November 2016

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
		IBM				
30	Japan Aerospace eXploration Agency Japan	**SORA-MA** - Fujitsu PRIMEHPC FX100, SPARC64 XIfx 32C 1.98GHz, Tofu interconnect 2 Fujitsu	110,160	3,157.0	3,481.1	1,652
31	Government United States	Cray XC30, Intel Xeon E5-2697v2 12C 2.7GHz, Aries interconnect Cray Inc.	225,984	3,143.5	4,881.3	6,328
32	Air Force Research Laboratory United States	**Thunder** - SGI ICE X, Xeon E5-2699v3/E5-2697 v3, Infiniband FDR, NVIDIA Tesla K40, Intel Xeon Phi 7120P HPE/SGI	152,692	3,126.2	5,610.5	4,820
33	Academic Center for Computing and Media Studies (ACCMS), Kyoto University Japan	**Camphor 2** - Cray XC40, Intel Xeon Phi 7250 68C 1.4GHz, Aries interconnect Cray Inc.	122,400	3,057.3	5,483.5	748
34	DKRZ - Deutsches Klimarechenzentrum Germany	**Mistral** - bullx DLC 720, Xeon E5-2680v3 12C 2.5GHz/E5-2695V4 18C 2.1Ghz, Infiniband FDR Bull, Atos Group	99,072	3,010.7	3,962.9	1,276
35	Information Technology Center, Nagoya University Japan	Fujitsu PRIMEHPC FX100, SPARC64 XIfx 32C 2.2GHz, Tofu interconnect 2 Fujitsu	92,160	2,910.0	3,244.0	1,382
36	Leibniz-Rechenzentrum Germany	**SuperMUC** - iDataPlex DX360M4, Xeon E5-2680 8C 2.70GHz, Infiniband FDR IBM/Lenovo	147,456	2,897.0	3,185.1	3,423
37	Leibniz-Rechenzentrum Germany	**SuperMUC Phase 2** - NeXtScale nx360M5, Xeon	86,016	2,813.6	3,578.3	1,481

Quelle: www.top500.org

HPC Top500 Liste - Stand November 2016

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	National Supercomputing Center in Wuxi China	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC	10,649,600	93,014.6	125,435.9	15,371
2	National Super Computer Center in Guangzhou China	**Tianhe-2 (MilkyWay-2)** - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
3	DOE/SC/Oak Ridge National Laboratory United States	**Titan** - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
4	DOE/NNSA/LLNL United States	**Sequoia** - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890
5	DOE/SC/LBNL/NERSC United States	**Cori** - Cray XC40, Intel Xeon Phi 7250 68C 1.4GHz, Aries interconnect Cray Inc.	622,336	14,014.7	27,880.7	3,939
6	Joint Center for Advanced High Performance Computing Japan	**Oakforest-PACS** - PRIMERGY CX1640 M1, Intel Xeon Phi 7250 68C 1.4GHz, Intel Omni-Path Fujitsu	556,104	13,554.6	24,913.5	2,719
7	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
8	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint** - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect , NVIDIA Tesla P100 Cray Inc.	206,720	9,779.0	15,988.0	1,312
9	DOE/SC/Argonne National Laboratory United States	**Mira** - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786,432	8,586.6	10,066.3	3,945
10	DOE/NNSA/LANL/SNL United States	**Trinity** - Cray XC40, Xeon E5-2698v3 16C 2.3GHz, Aries interconnect	301,056	8,100.9	11,078.9	4,233

Quelle: www.top500.org

HPC Top500 Liste - Stand November 2016

Laptop heute

Quelle: www.top500.org

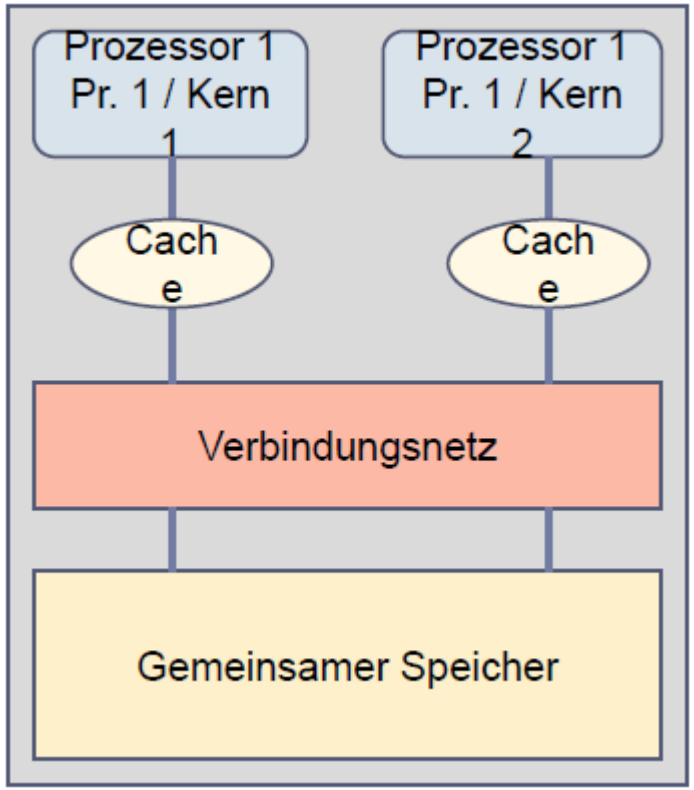
Möglichkeiten der Parallelen Programmierung :

- **OpenMP** - Möglich bei der Nutzung von gemeinsamem Speicher
  (shared memory directives)

- **MPI** (Message-Passing Interface)
  - bei Rechnerarchitektur mit verteiltem Speicher
  - derzeit einziger Standard mit Portabilität auf allen Plattformen

- **Hybride** Programmierung: Kombination von MPI und OpenMP

OpenMP - Gemeinsamer Speicherzugriff mittels SMP



SMP: Symmetrisches Multiprozessersystem (symmetric multiprocessing)

Multiprozessor-Architektur,
bei der zwei oder mehr identische Prozessoren
einen gemeinsamen Adressraum besitzen.

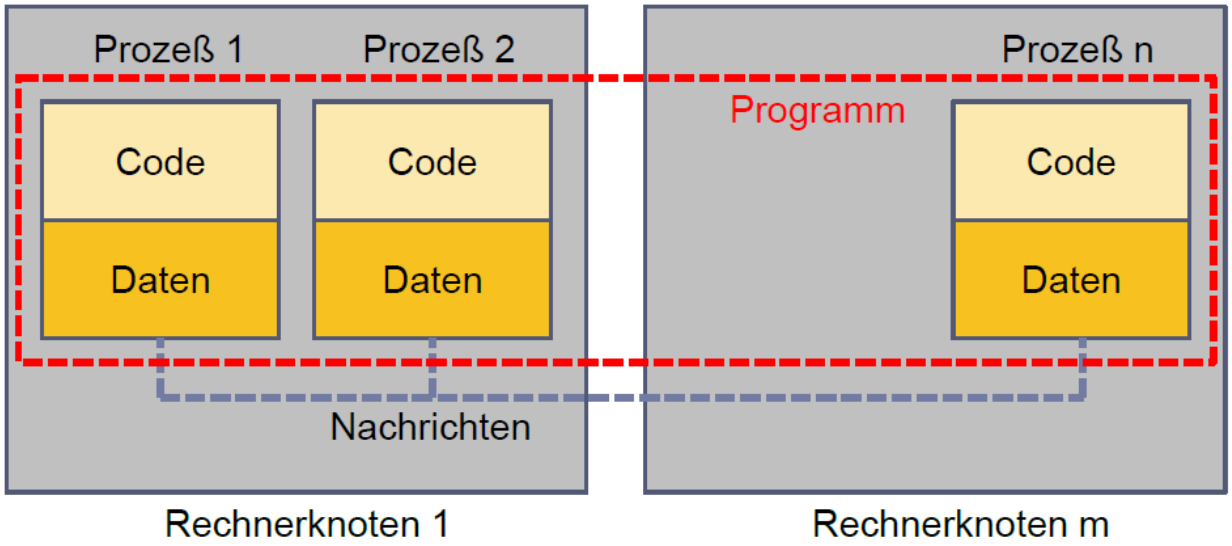
Eine SMP-Architektur erlaubt es,
die laufenden Prozesse dynamisch auf alle
verfügbaren Prozessoren zu verteilen.

Dagegen muss beim asymmetrischen Multiprocessing
jeder CPU eine Aufgabe fest zugewiesen werden
(z. B. führt CPU0 Betriebssystemaufrufe und CPU1 Benutzerprozesse aus).

Source: Grafik Ludwig WS12/13, Text Wikipedia

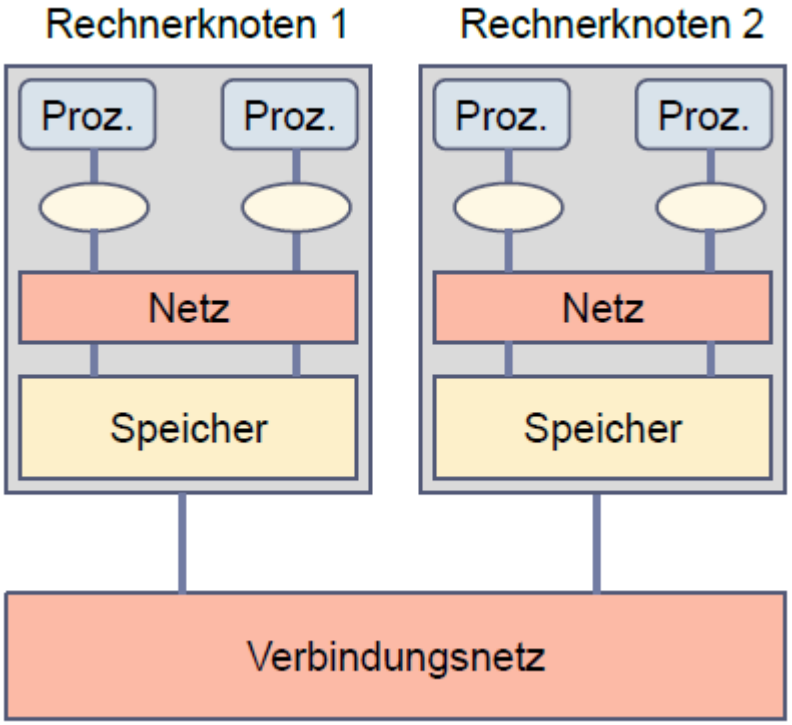
MPI – Hardware Voraussetzung

Quelle: Ludwig WS12/13



- Keinen direkten Zugriff auf Memory (Daten) von anderen Prozessen.
- Datenverfügbarkeit über expliziten Datenaustausch (Senden/Empfangen) mit anderen Prozessen!

Hybride Programmierung



Existierende HPC Rechner sind heute meist eine Kombination aus **Rechnerknoten mit gemeinsamem Speicher**, von den man viele verwendet und **über ein Verbindungsnetz** verbindet.

Hybride Programmierung ermöglicht die Kombination von MPI und OpenMP, aber bedarf mehr Struktur der Zugriffsrechte.

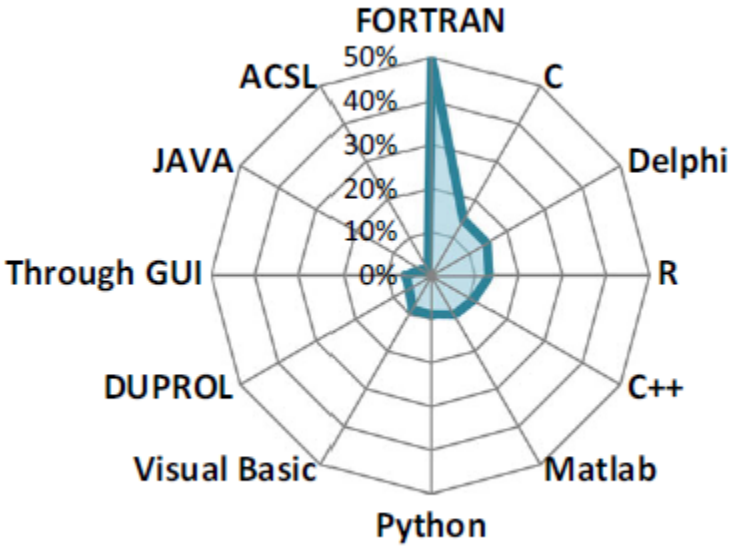
Vortrag von Dr. Panagiotis Adamidis als DKRZ Beitrag am 5. Juli 2017

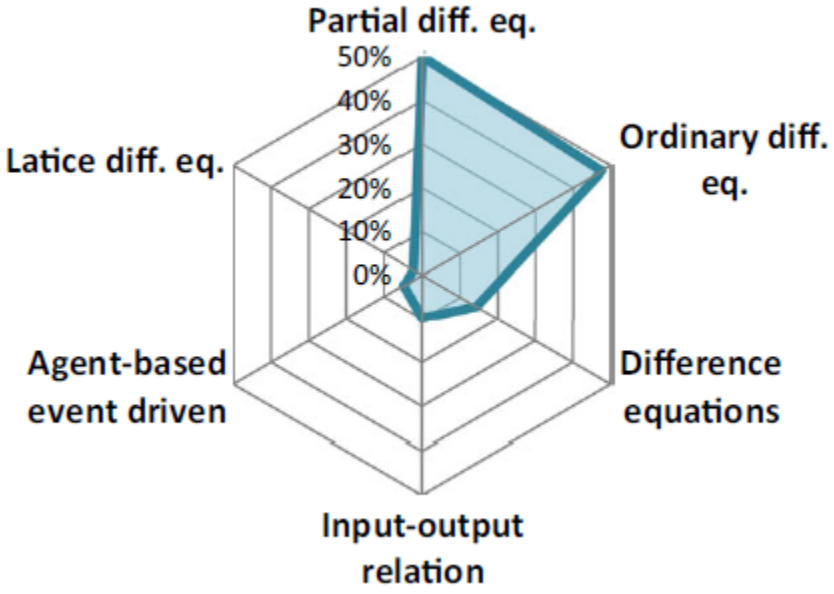
Quelle: Ludwig WS12/13

Software-Anwendungen in Erdsystemmodellen

Paper: Exploring, exploiting and evolving diversity of aquatic ecosystem models: a community perspective.

Janssen et al., 2015 in Aquatic Ecology





Software-Aspekte für Parallele Programmierung I

Neben Hardware gibt es auch einige Anforderungen an Softwarekomponenten, z.B. für die Einbindung von Programm-Bibliotheken.

Thread-Sicherheit (Thread safety)

Softwarekomponente können gleichzeitig von verschiedenen Programmbereichen mehrfach ausgeführt werden, ohne dass diese sich gegenseitig behindern.

Zum Zweck der Mehrfachausführung bieten Betriebssysteme das Konzept von sogenannte Threads, die als „leichtgewichtige Prozesse“ gelten.

Quelle: Wikipedia

Software-Aspekte für Parallele Programmierung II

Jeder Thread arbeitet dabei unabhängig von den anderen einen Programmteil ab.

Häufig muss das Programm dabei gleichzeitig auf einen gemeinsamen Speicherbereich des Computers zugreifen.

Änderungen im Speicher durch verschiedene Threads müssen koordiniert werden, um einen chaotischen Zustand des Speichers zu verhindern.

Quelle: Wikipedia

Software-Aspekte für Parallele Programmierung III

Problem beim Parallelen Programmieren:

Reihenfolge der Ausführung nicht steuerbar, bzw. nicht deterministisch.

Ein Beispiel für einen unkontrollierten Zustand des Speichers sind sogenannte

Wettlaufsituation (Race Condition)

Wettlaufsituationen bezeichnen in der Programmierung eine Konstellation,
in der das Ergebnis einer Operation vom zeitlichen Verhalten
bestimmter Einzeloperationen abhängt.
D.h. die Reihenfolge in der die Berechnungen und Speicherung ablaufen
ist nicht vertauschbar!

Quelle: Wikipedia

MPI (Message Passing Interface) - Nachrichtenaustausch

MPI Nachrichten sind Datenpakete die zwischen Prozessen ausgetauscht werden.

Hardware Rahmenbedingungen:

- Keinen direkten Zugriff auf Speicher (bzw. Daten) von anderen Prozessen.
- Datenverfügbarkeit **nur** über expliziten Datenaustausch (Senden/Empfangen) mit anderen Prozessen!

!! Vorteil: MPI Prozesse lassen sich skalieren !!

Programmstruktur in der Modellierung

Der Ablauf eines Modelles wir üblicherweise in 3 Phasen eingeteilt:

1) Initialisierung

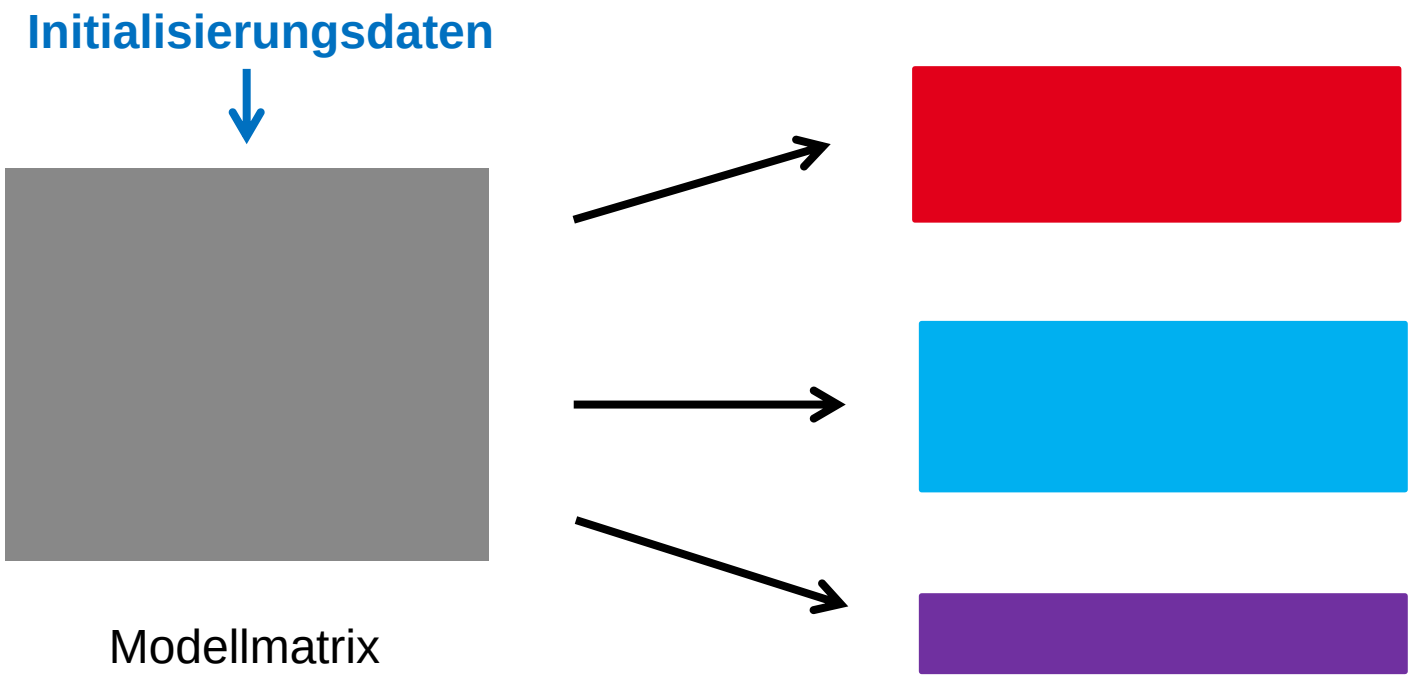
2) Berechnung des Modells

3) Finalisierung

Diese Einteilung wird z.B. von Kopplern wie ESMF gefordert.

MPI Nachrichtenaustausch in der Modellierung I

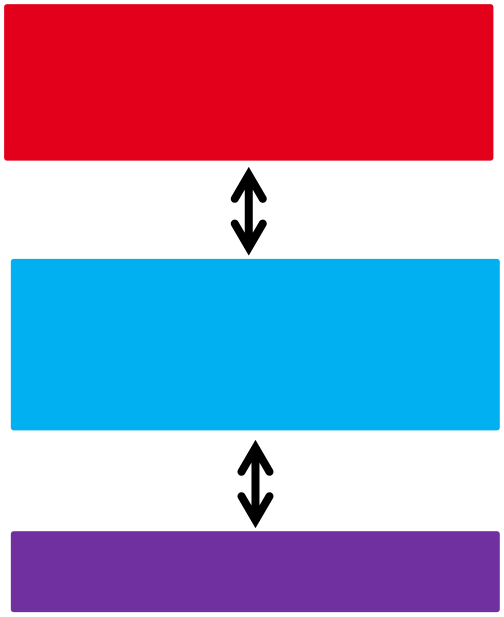
1) Initialisierung: Aufteilung der Rechengebiete und Initialisierung



MPI Nachrichtenaustausch in der Modellierung II

2) Berechnung des Modells auf Teilmatrizen: Austausch zwischen Teilmatrizen

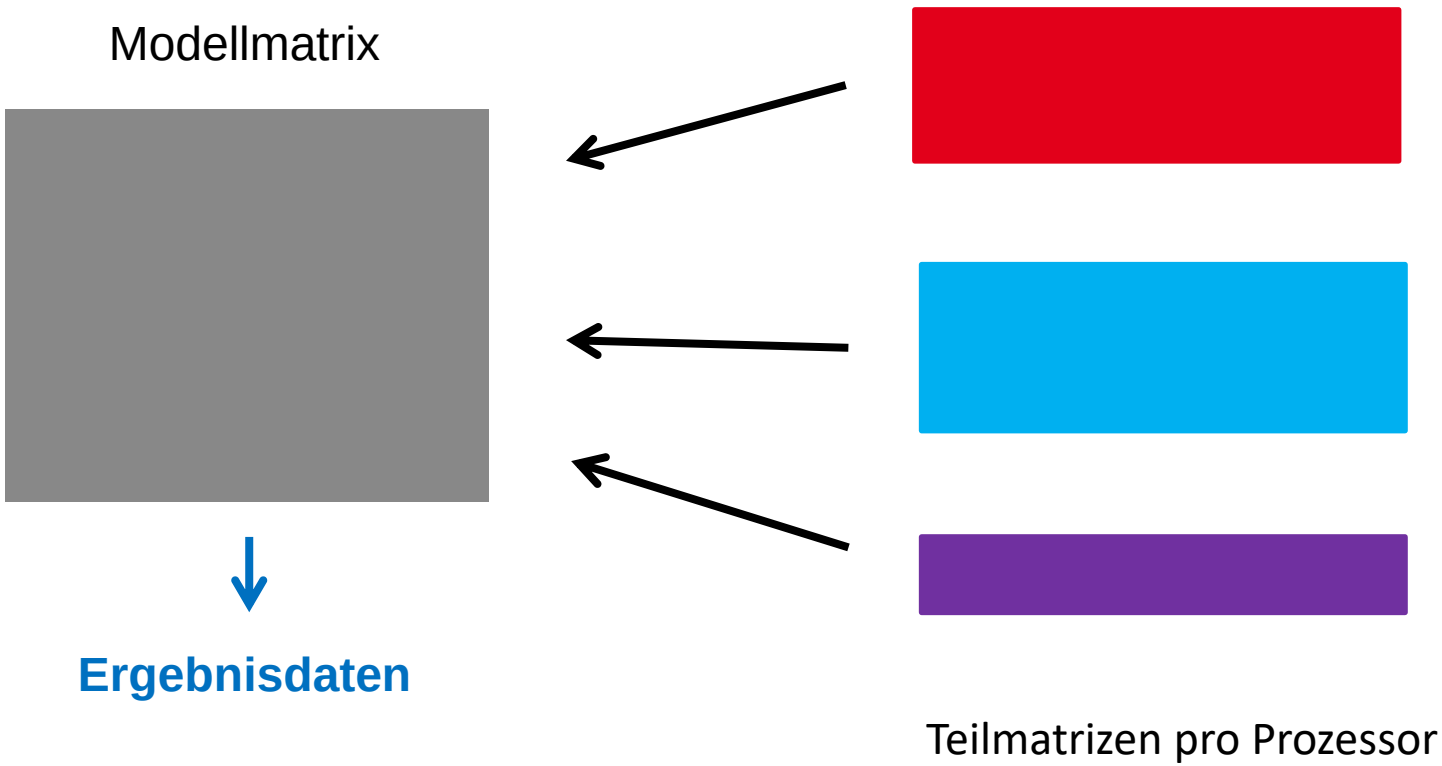
Modellmatrix



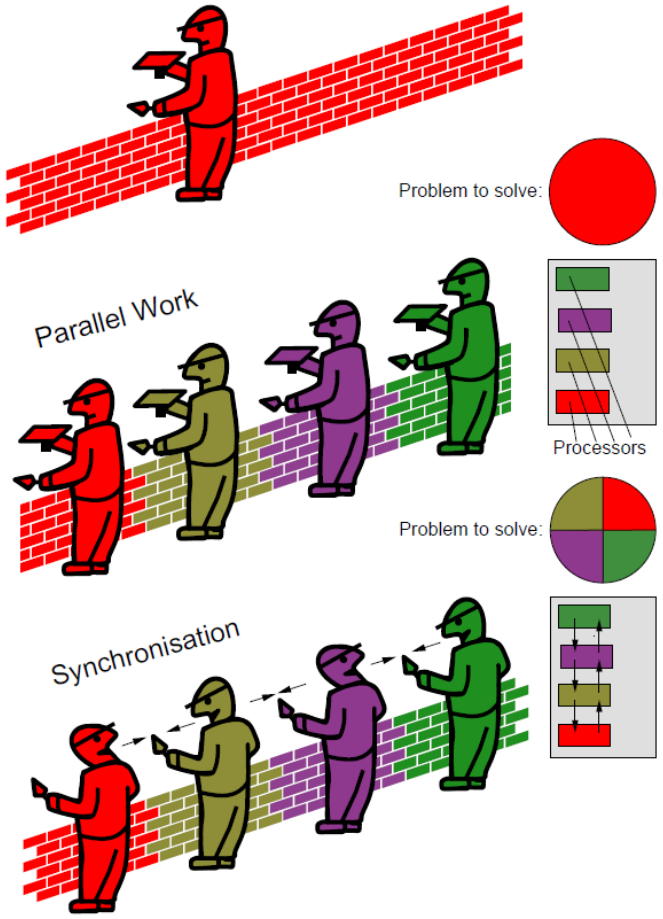
Teilmatrizen pro Prozessor

MPI Nachrichtenaustausch in der Modellierung III

3) Finalisierung: Zusammenfügen der Rechenergebnisse der Teilmatrizen



MPI Nachrichtenaustausch: Zu Beachten!!



[picture by W. Baumann]

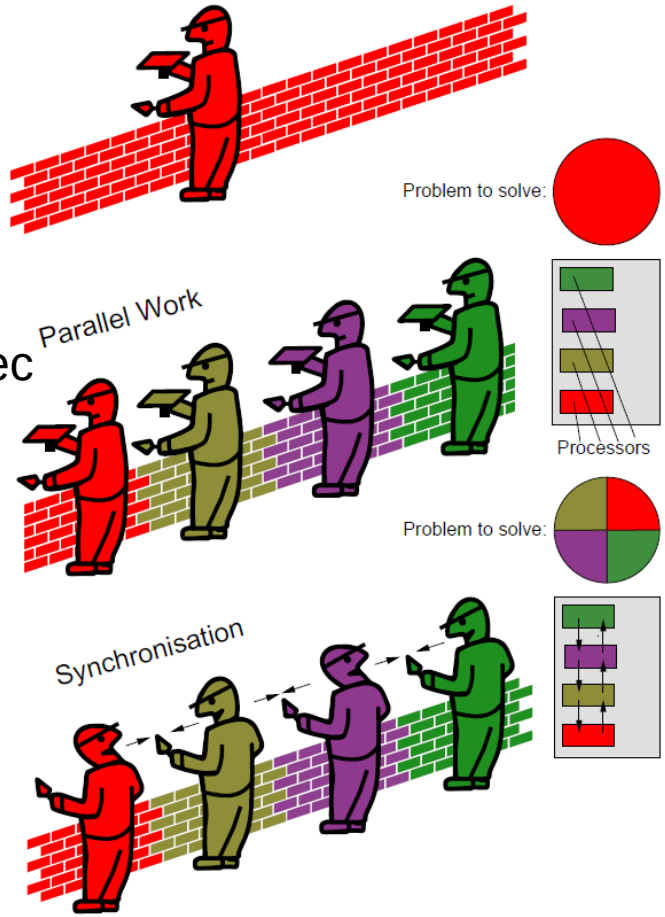
MPI Nachrichtenaustausch: Zu Beachten!!

Als Info für das Verhältnis
Rechnen / Nachrichtenaustausch
soll folgende Abschätzung dienen:

Ein moderner Parallelrechner schafft
ca. 3 Mrd. floating point operationen / Sec

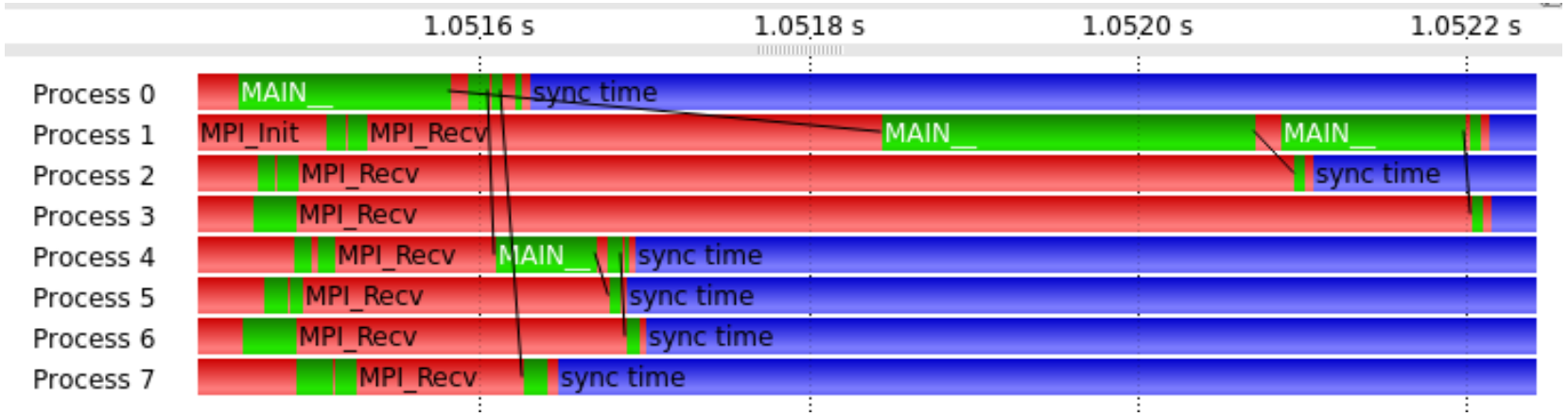
Der Nachrichtenaustausch
aber nur 10 Mio. Wörter / Sec

Faktor 300 !!



[picture by W. Baumann]

Visualisierung des Programmablaufes mit Vampire:



Danke,
gibt es noch Fragen?