# Performance Barriers in highly scaling earth-system models

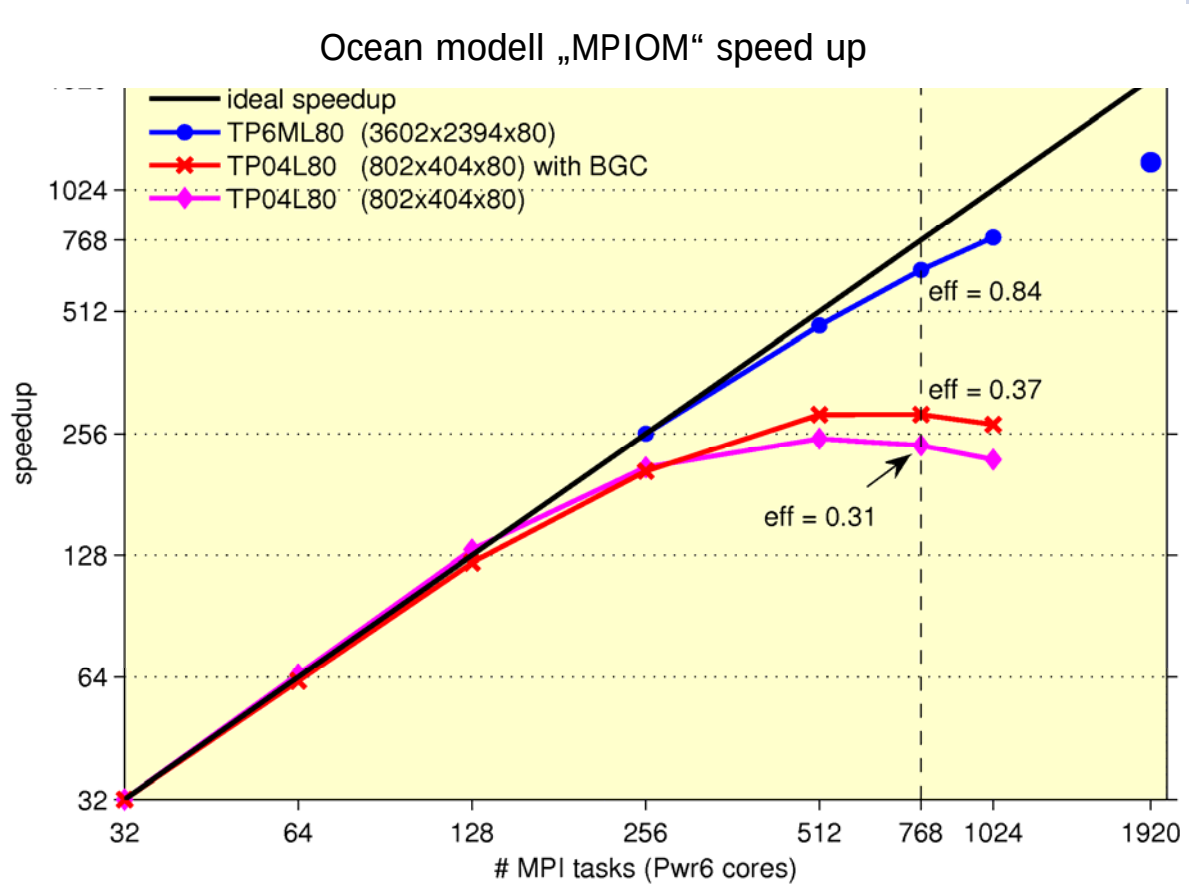
Panagiotis Adamidis
Deutsches Klimarechenzentrum GmbH

Jörg Behrens, Thomas Jahns (DKRZ),
Florian Wilhelm (KIT)

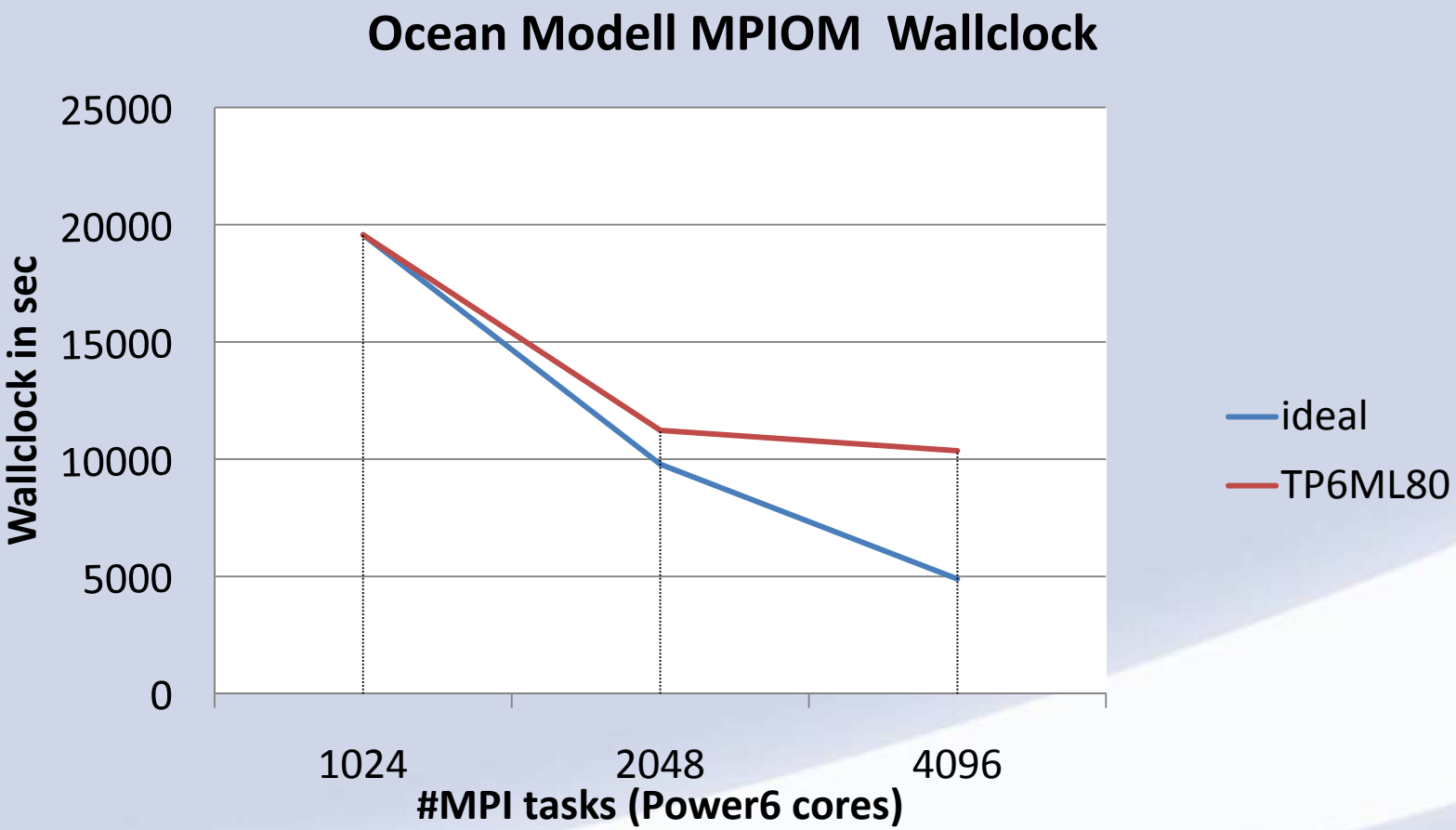
# Bottlenecks

- Bottlenecks of Massively Parallel Computing Systems
  - Communication Network
  - Memory Bandwidth
  - Idle Processors

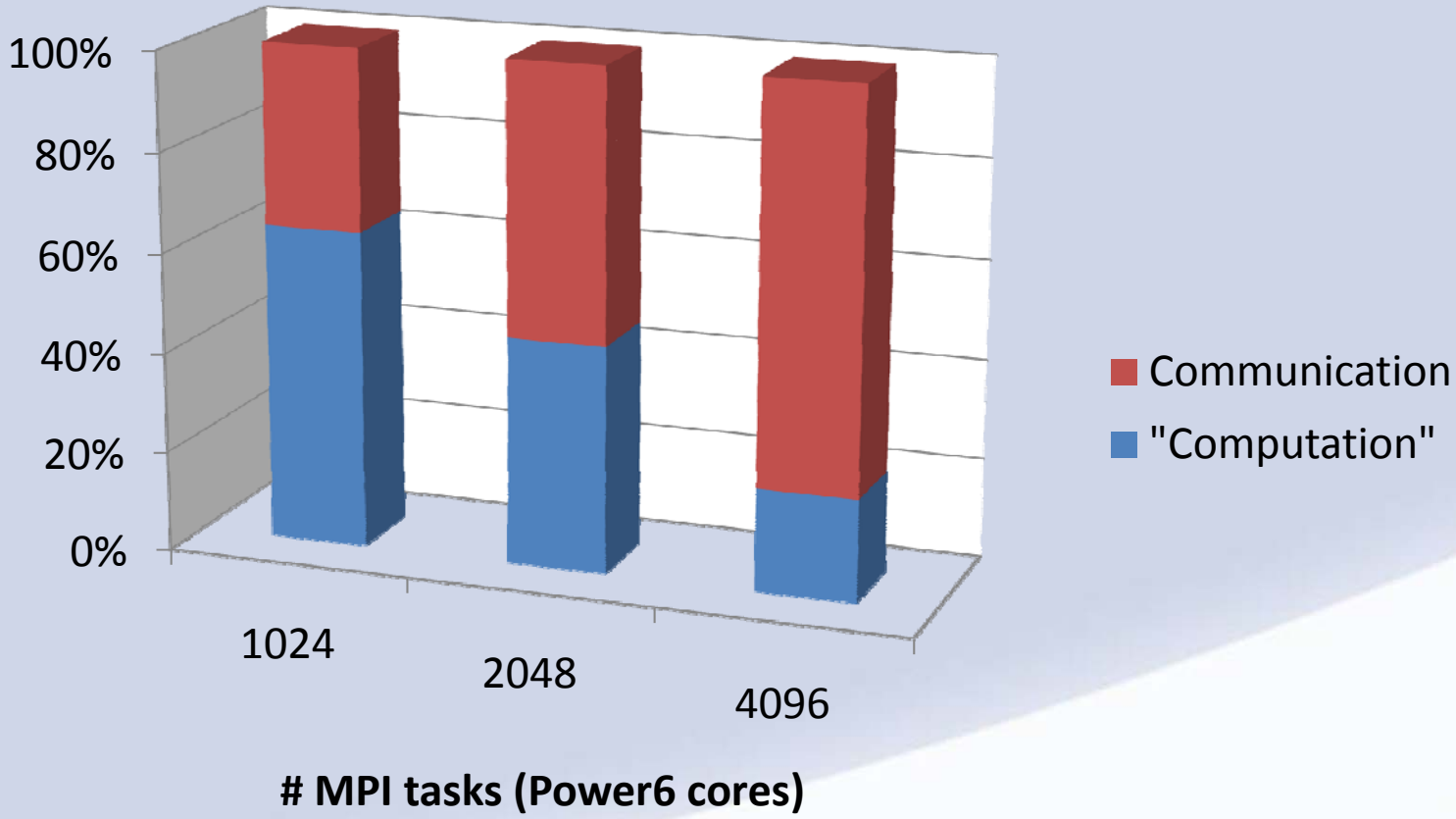
# Scaling today



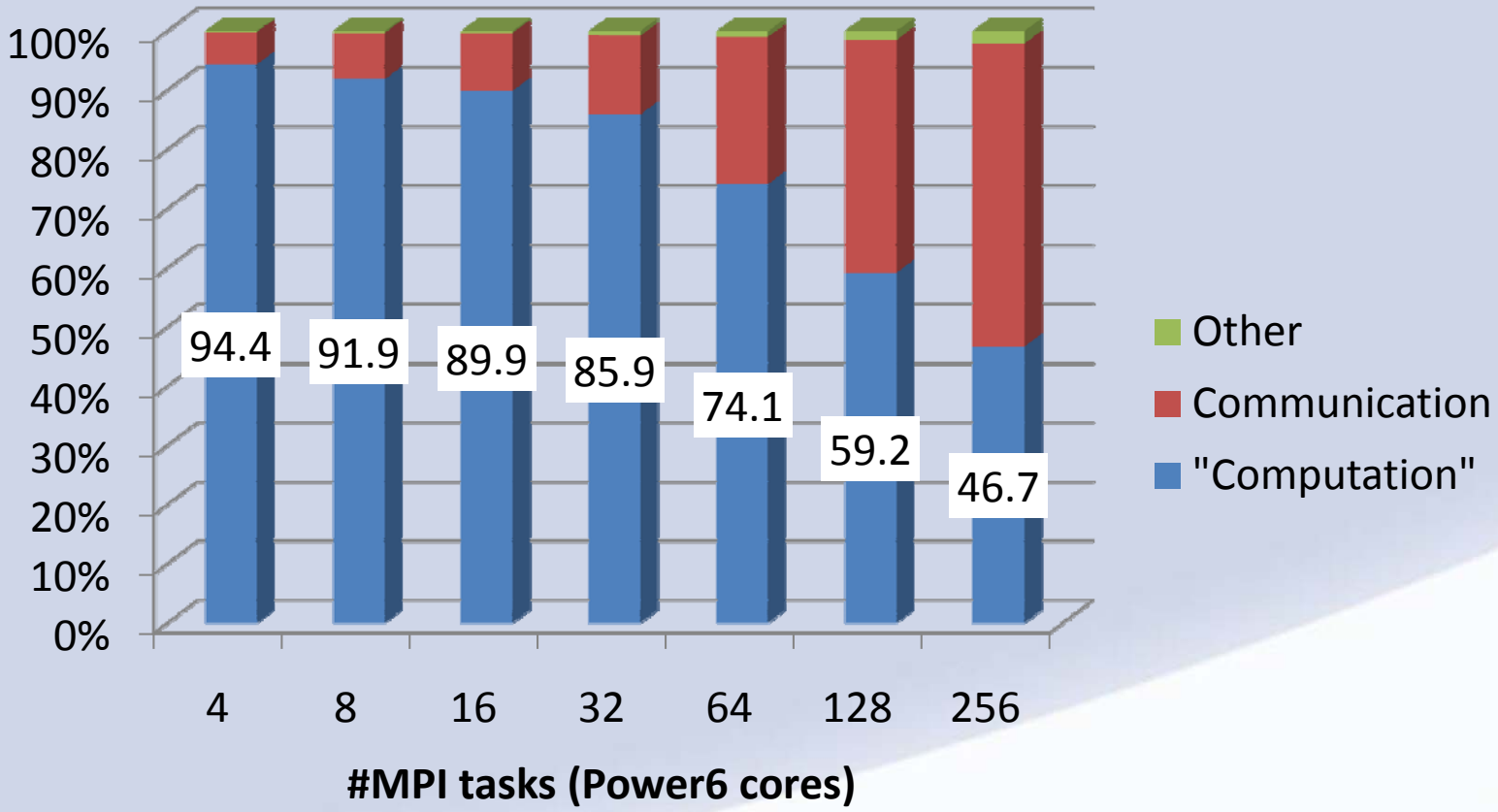
# Simulating 1 month with dt=600



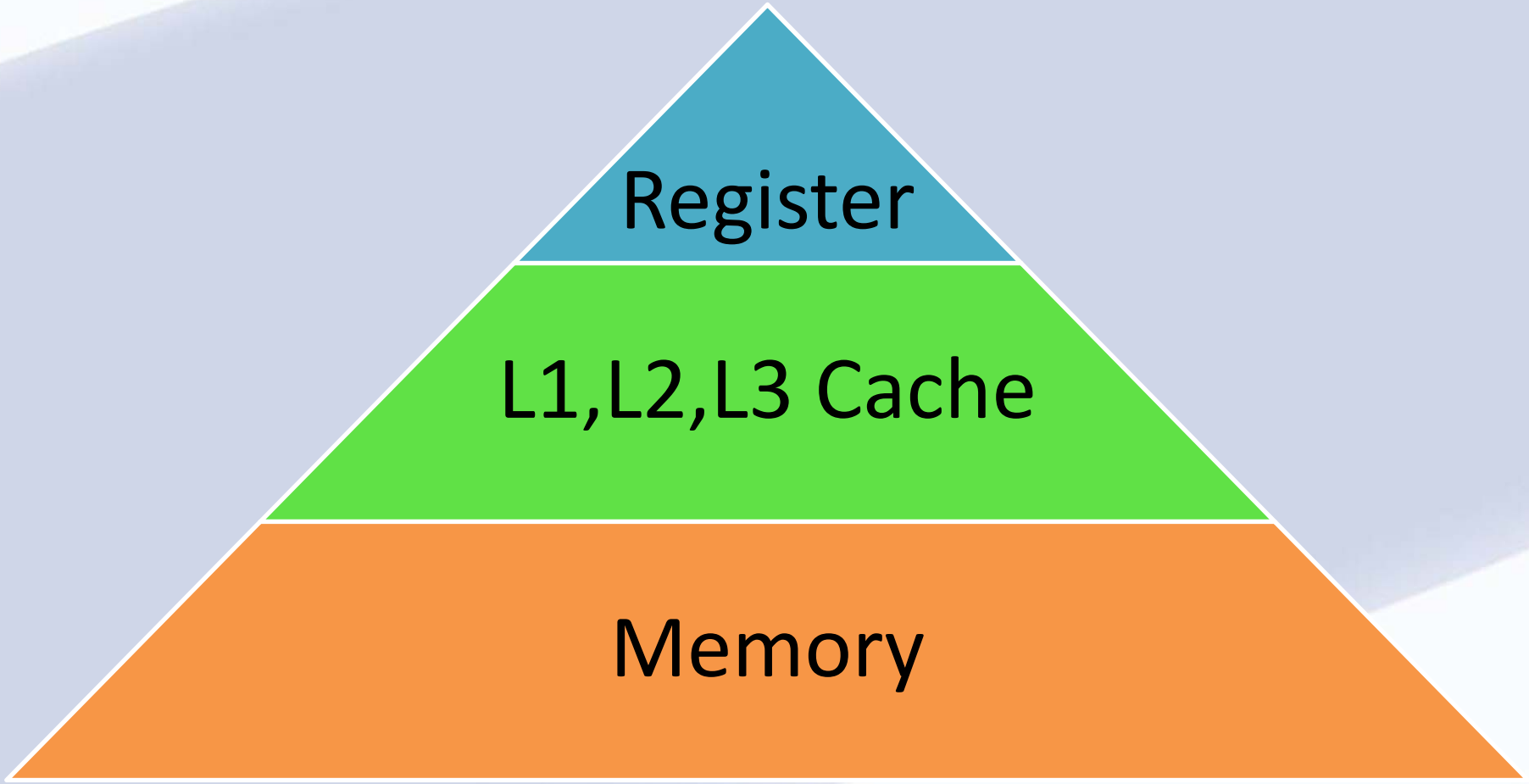
# Computation vs Communication TP6ML80



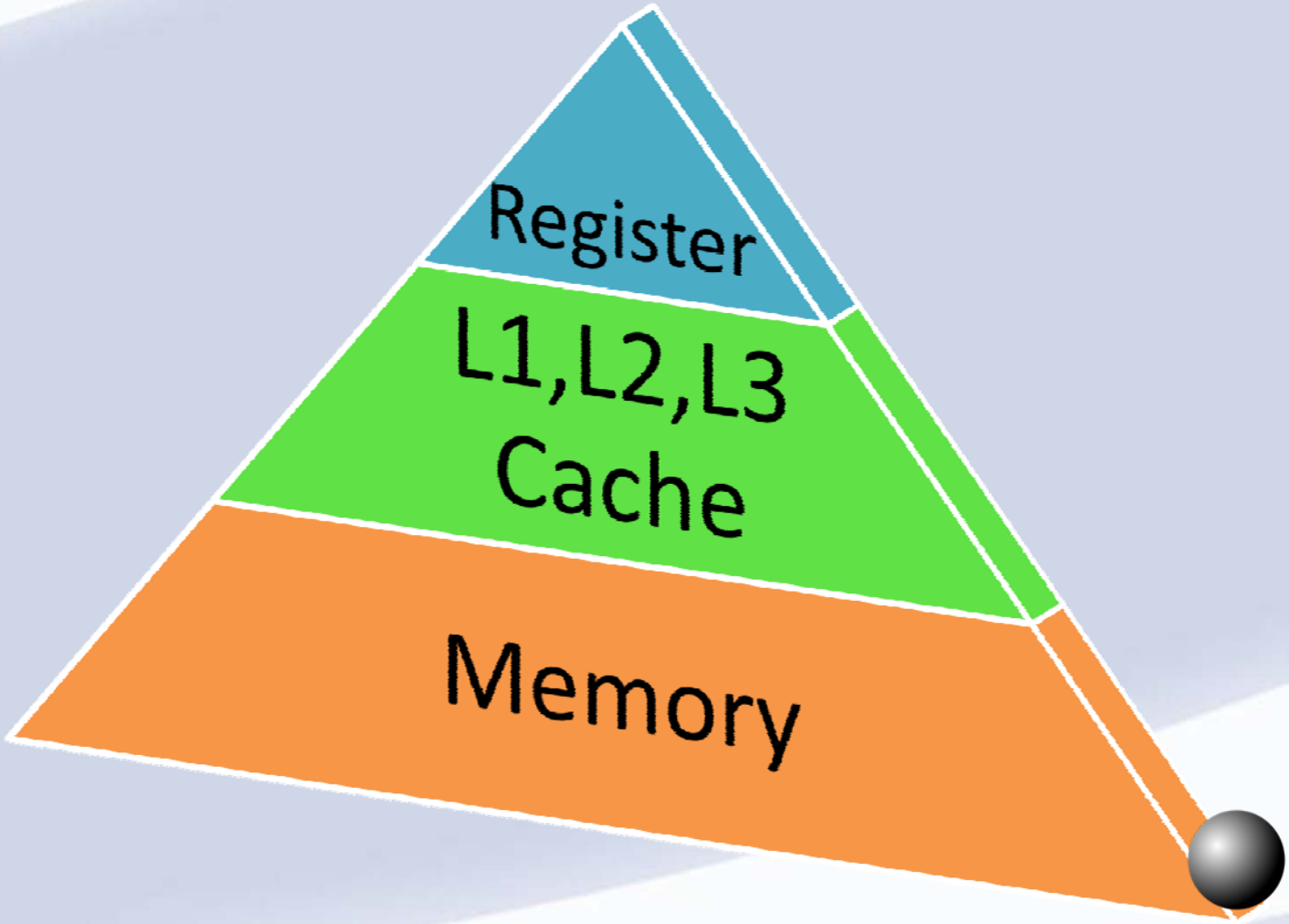
# Computation vs Communication TP04L80



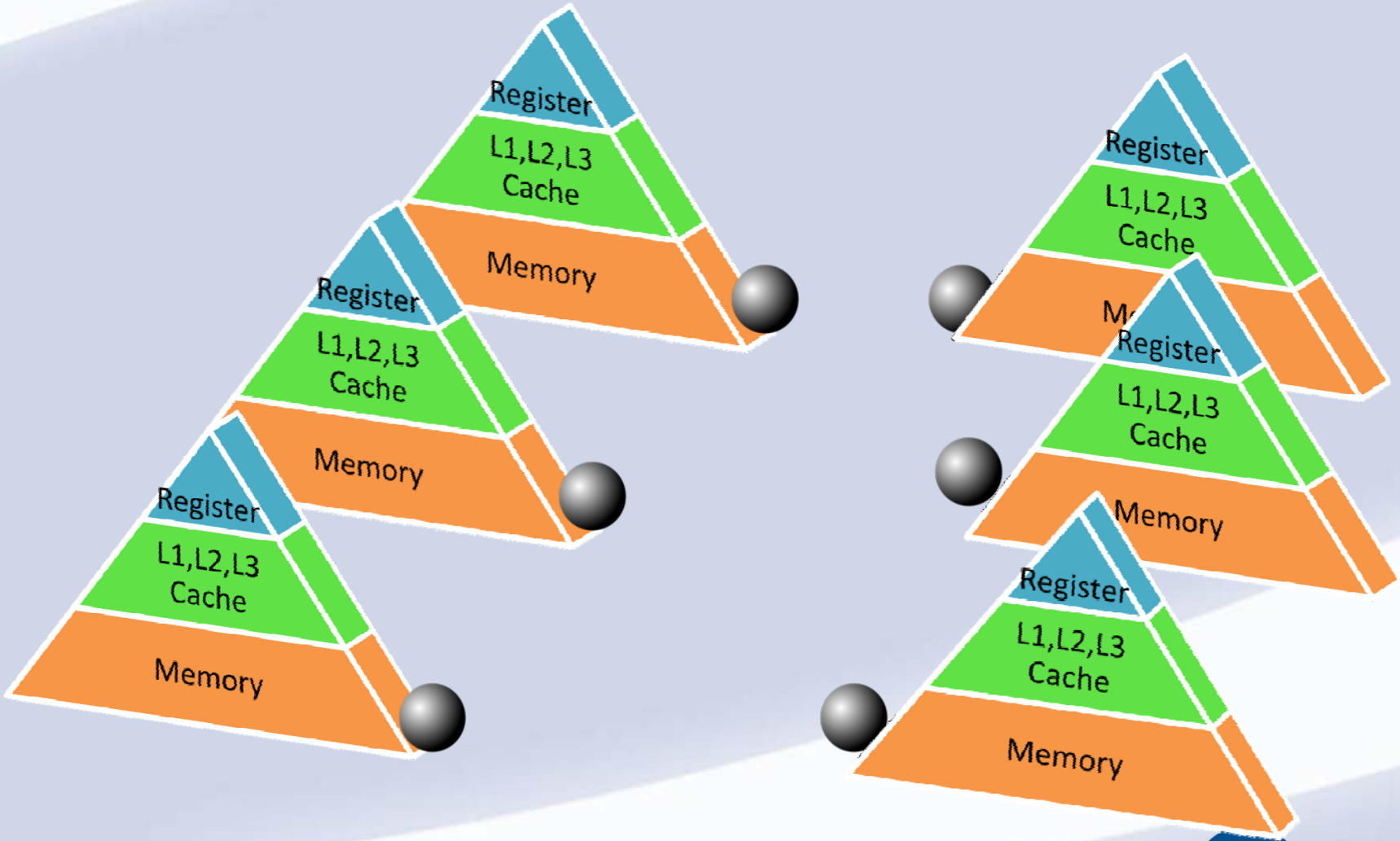
# Memory Hierarchy



# Data Movement



# Data Movement in Parallel Systems



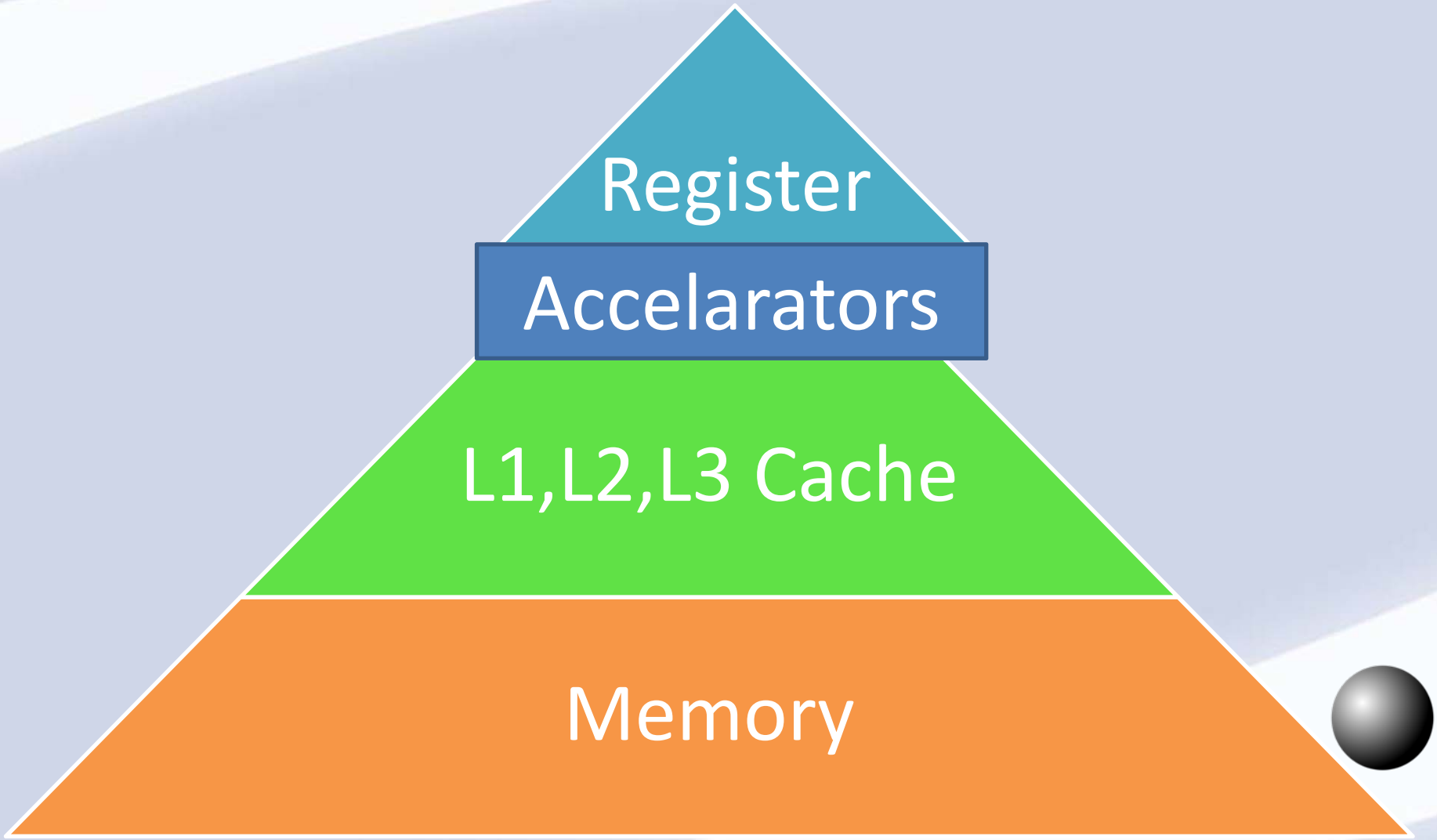
# Sisyphean Challenge

- According to greek mythology, “Sisyphus” was a greek King, famous for his cleverness.
- He played his tricks not only on humans but also on the Gods, thus being the craftiest of men
- He even managed to outwit “Thanatos” (death) himself

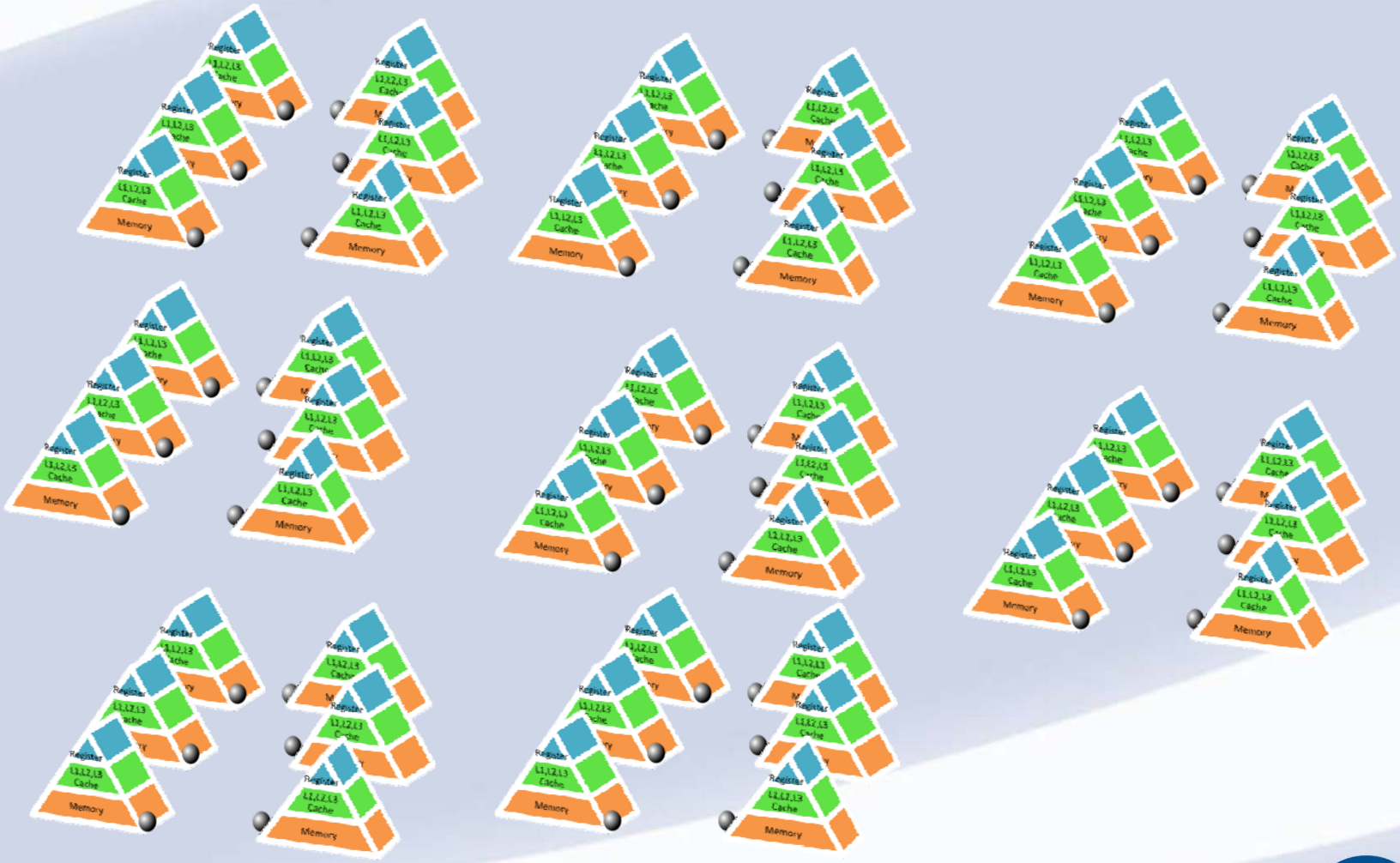
# Sisyphean Challenge

- As a punishment for his trickery, Sisyphus was compelled to roll a huge rock up a steep hill, only to watch it roll back down, and to repeat this throughout eternity

# Future Processors ?



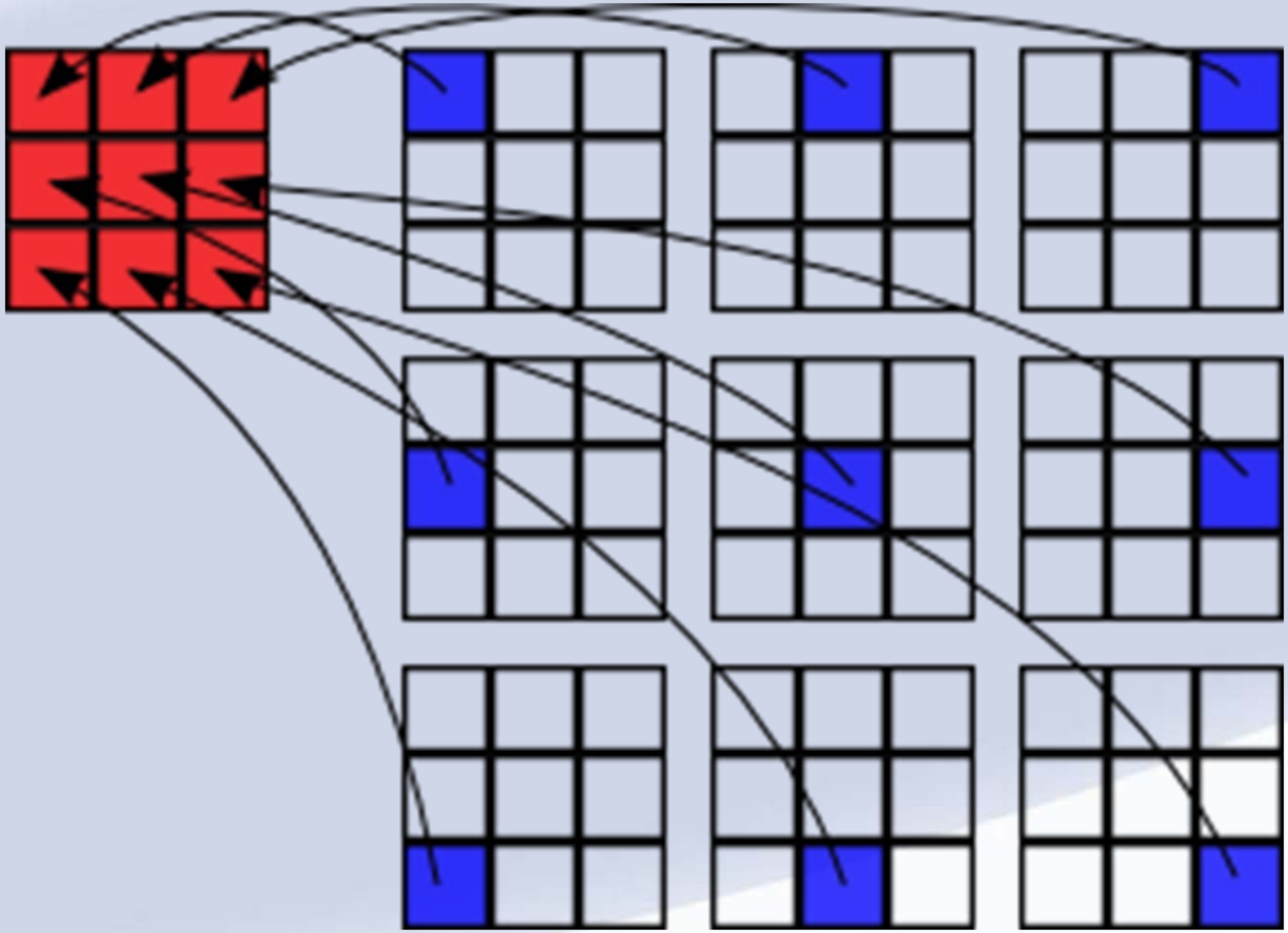
# Future Parallel Systems

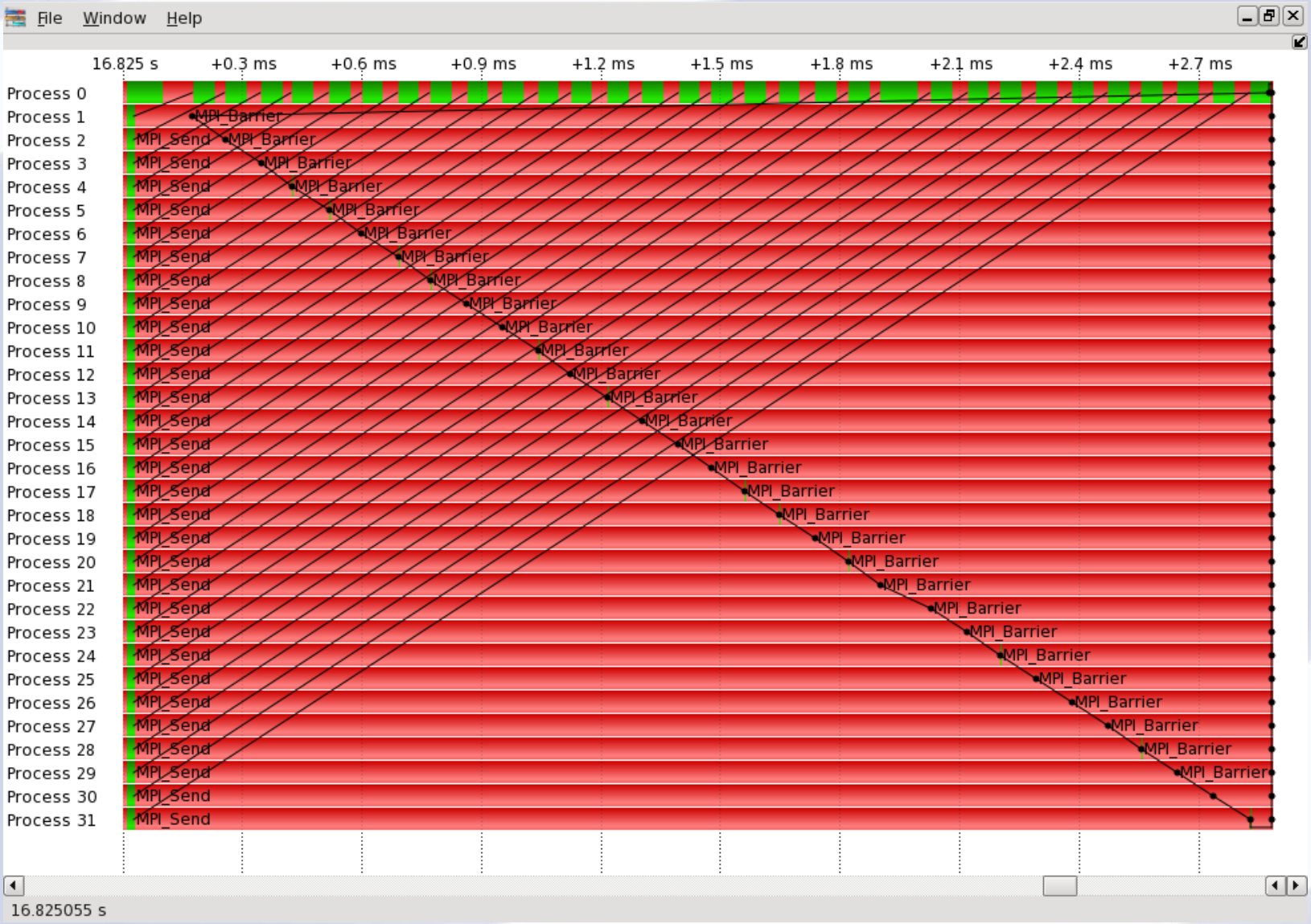


# Approach

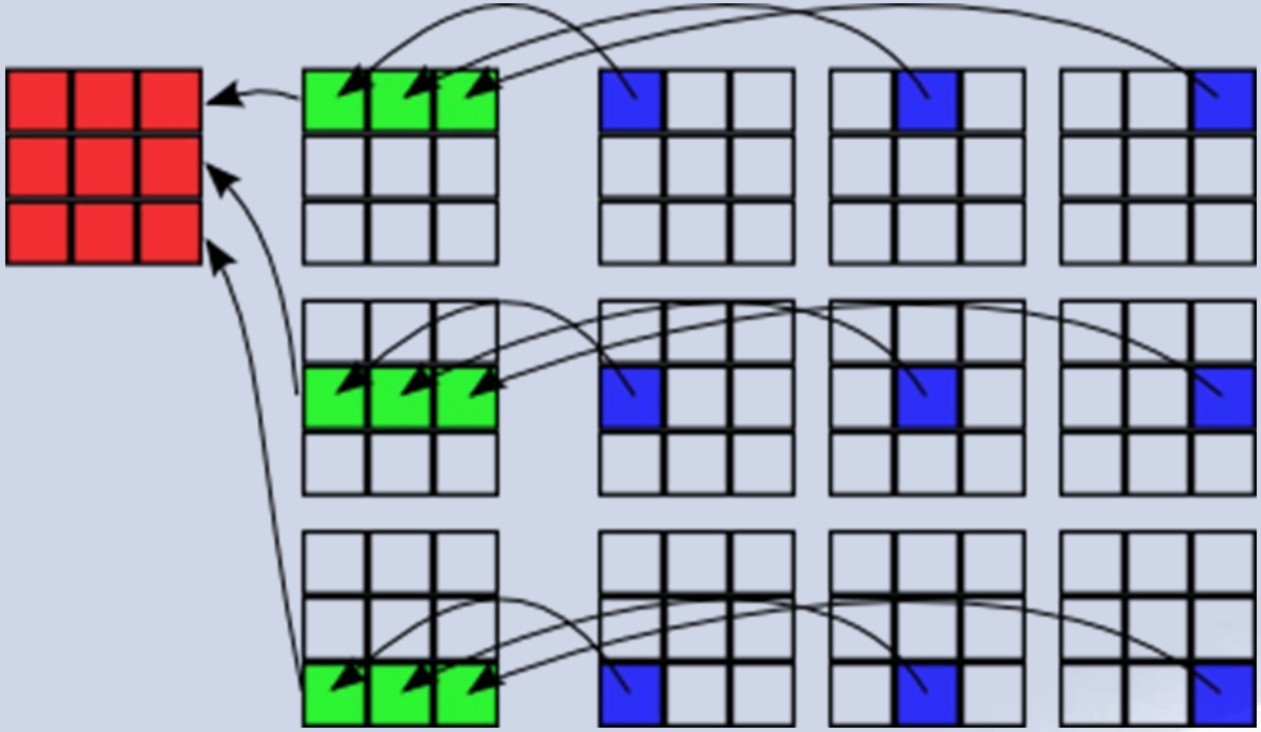
- Solutions
  - Optimizing Communication
  - Load Balancing
  - Algorithms which are better suited for the specific architecture

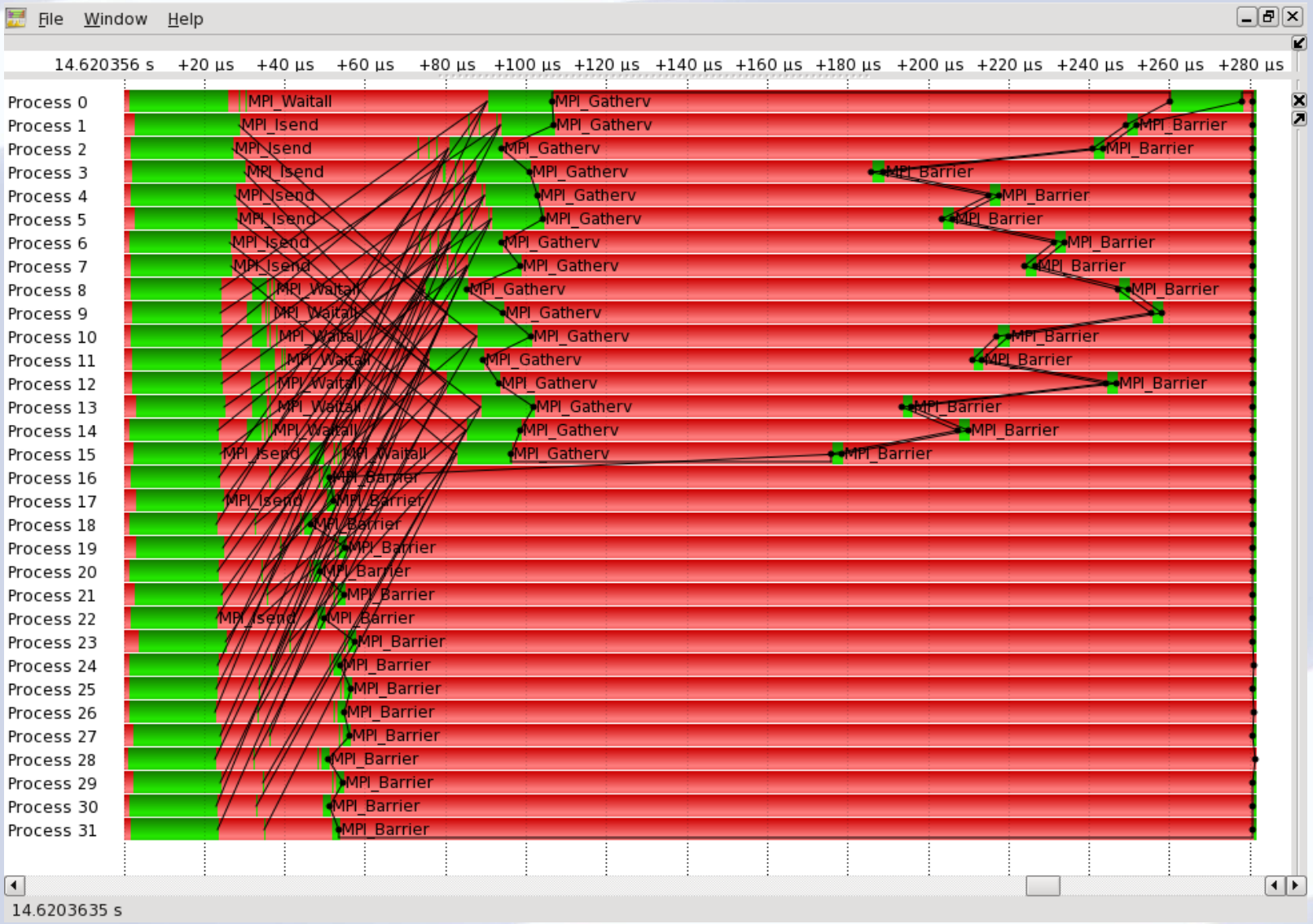
# Optimizing Communication

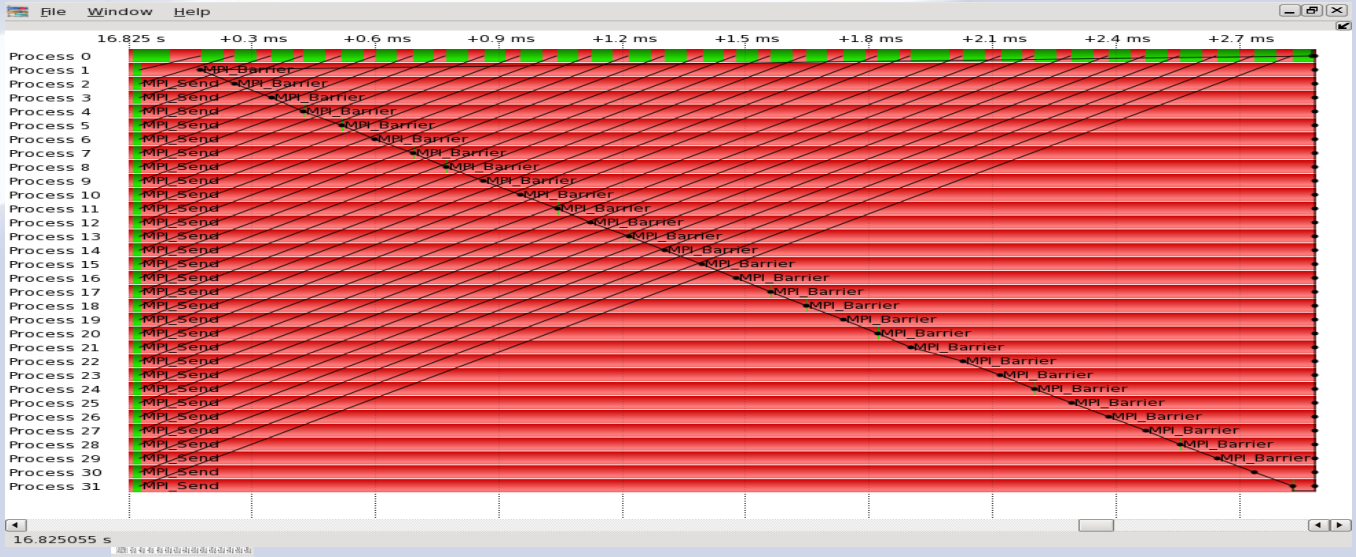



File Window Help
16.825 s
+0.3 ms
+0.6 ms
+0.9 ms
+1.2 ms
+1.5 ms
+1.8 ms
+2.1 ms
+2.4 ms
+2.7 ms
Process 0
Process 1
Process 2
Process 3
Process 4
Process 5
Process 6
Process 7
Process 8
Process 9
Process 10
Process 11
Process 12
Process 13
Process 14
Process 15
Process 16
Process 17
Process 18
Process 19
Process 20
Process 21
Process 22
Process 23
Process 24
Process 25
Process 26
Process 27
Process 28
Process 29
Process 30
Process 31
MPI_Send
MPI_Barrier
16.825055 s

# Optimizing Communication

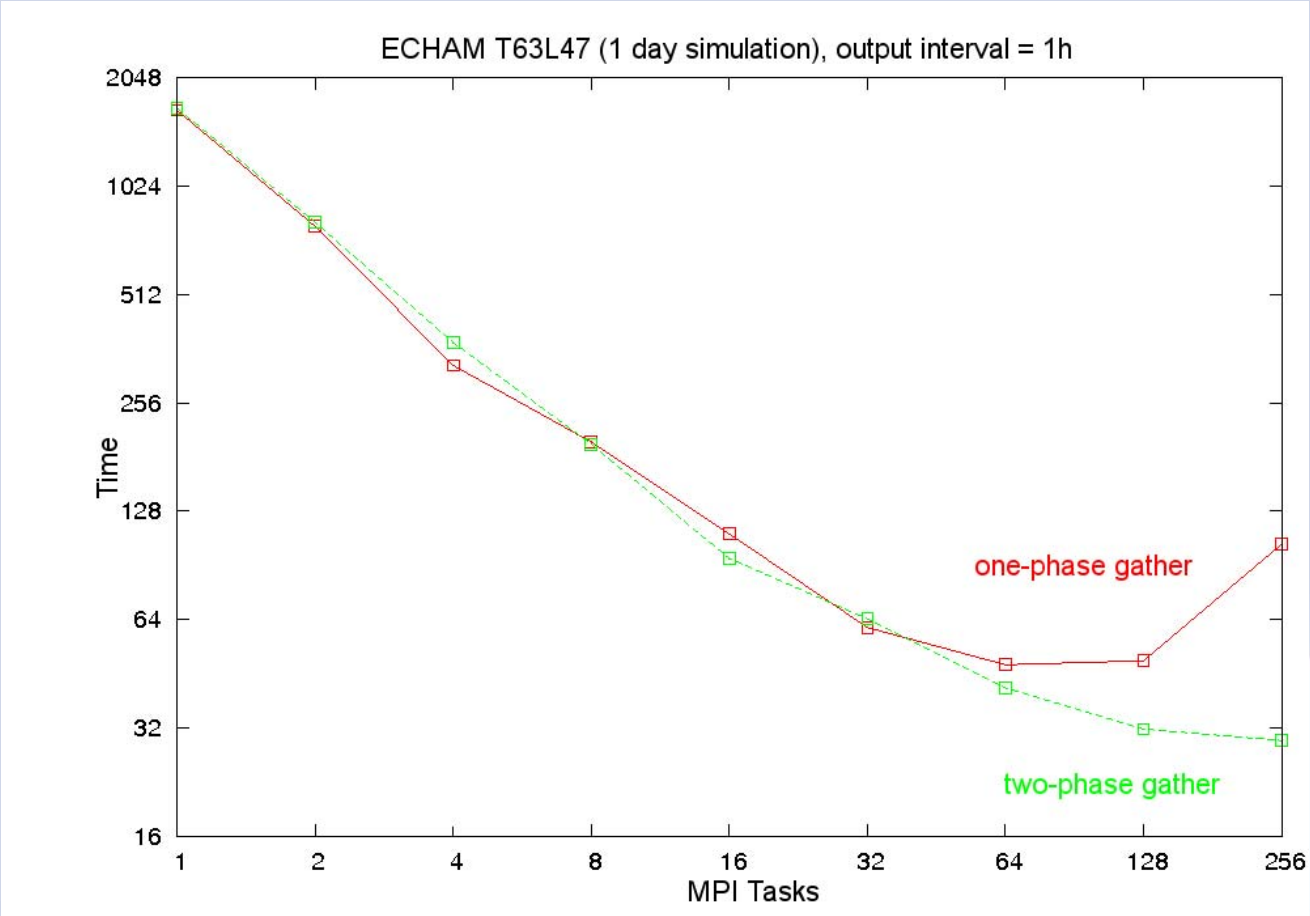



File Window Help
14.620356 s +20 µs +40 µs +60 µs +80 µs +100 µs +120 µs +140 µs +160 µs +180 µs +200 µs +220 µs +240 µs +260 µs +280 µs
Process 0 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 1 MPI_Isend MPI_Gatherv MPI_Barrier
Process 2 MPI_Isend MPI_Gatherv MPI_Barrier
Process 3 MPI_Isend MPI_Gatherv MPI_Barrier
Process 4 MPI_Isend MPI_Gatherv MPI_Barrier
Process 5 MPI_Isend MPI_Gatherv MPI_Barrier
Process 6 MPI_Isend MPI_Gatherv MPI_Barrier
Process 7 MPI_Isend MPI_Gatherv MPI_Barrier
Process 8 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 9 MPI_Waitall MPI_Gatherv
Process 10 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 11 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 12 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 13 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 14 MPI_Waitall MPI_Gatherv MPI_Barrier
Process 15 MPI_Isend MPI_Waitall MPI_Gatherv MPI_Barrier
Process 16 MPI_Barrier
Process 17 MPI_Isend MPI_Barrier
Process 18 MPI_Barrier
Process 19 MPI_Barrier
Process 20 MPI_Barrier
Process 21 MPI_Barrier
Process 22 MPI_Isend MPI_Barrier
Process 23 MPI_Barrier
Process 24 MPI_Barrier
Process 25 MPI_Barrier
Process 26 MPI_Barrier
Process 27 MPI_Barrier
Process 28 MPI_Barrier
Process 29 MPI_Barrier
Process 30 MPI_Barrier
Process 31 MPI_Barrier
14.6203635 s

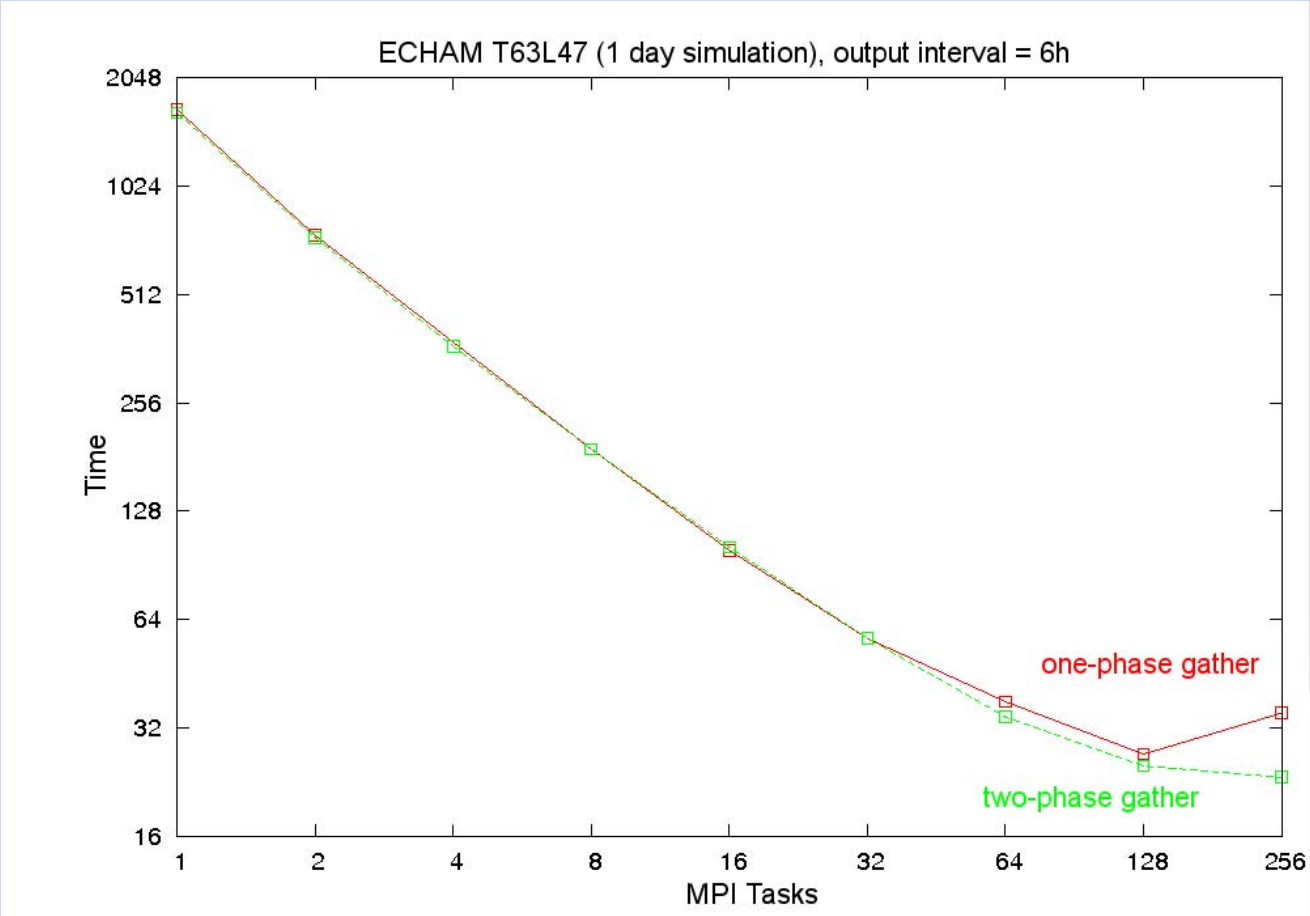

File Window Help
16.825 s
+0.3 ms
+0.6 ms
+0.9 ms
+1.2 ms
+1.5 ms
+1.8 ms
+2.1 ms
+2.4 ms
+2.7 ms
Process 0
Process 1
Process 2
Process 3
Process 4
Process 5
Process 6
Process 7
Process 8
Process 9
Process 10
Process 11
Process 12
Process 13
Process 14
Process 15
Process 16
Process 17
Process 18
Process 19
Process 20
Process 21
Process 22
Process 23
Process 24
Process 25
Process 26
Process 27
Process 28
Process 29
Process 30
Process 31
MPI_Send
MPI_Barrier
16.825055 s


Time
0.28msec
2.8msec

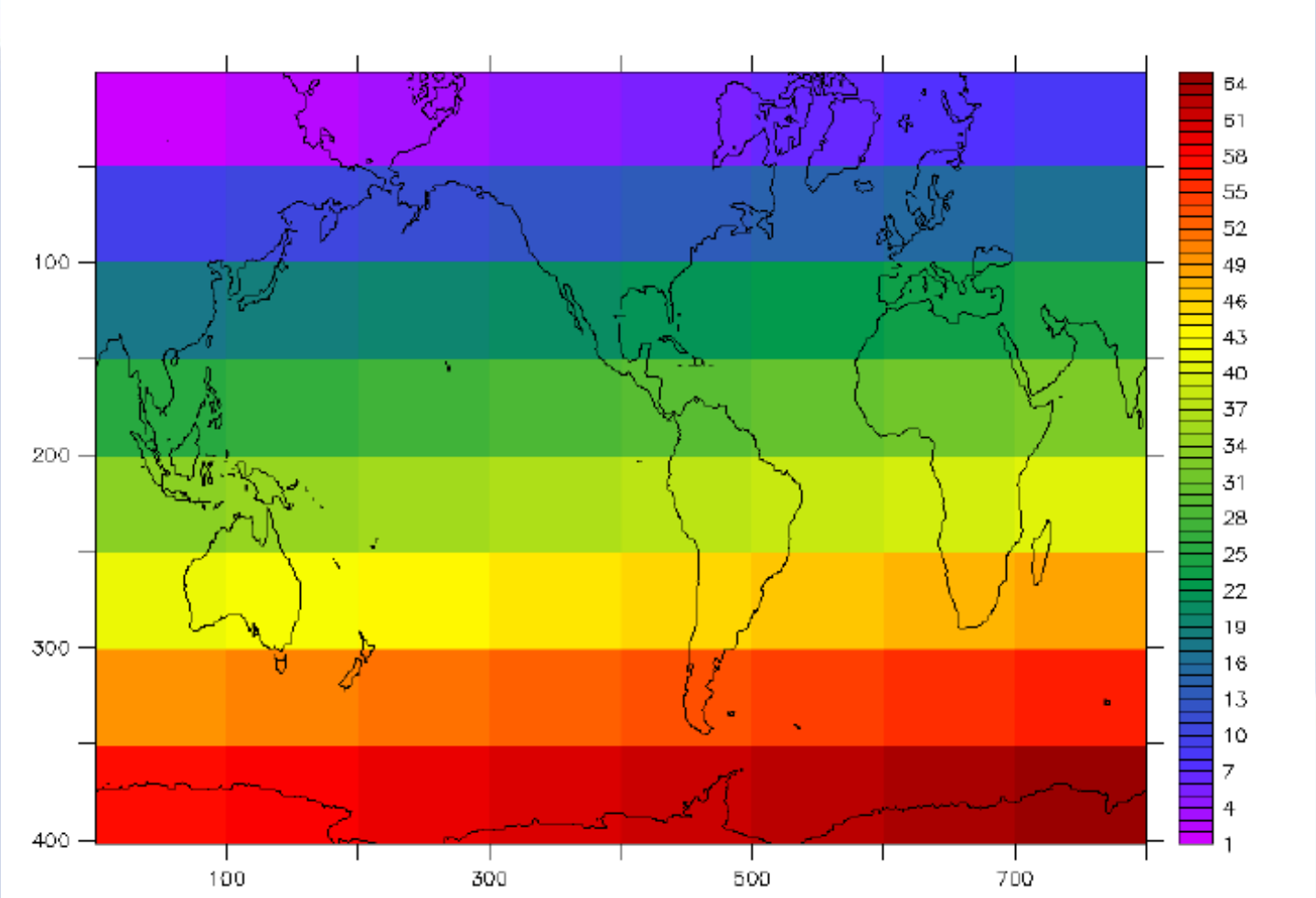
# Optimizing Communication



# Optimizing Communication

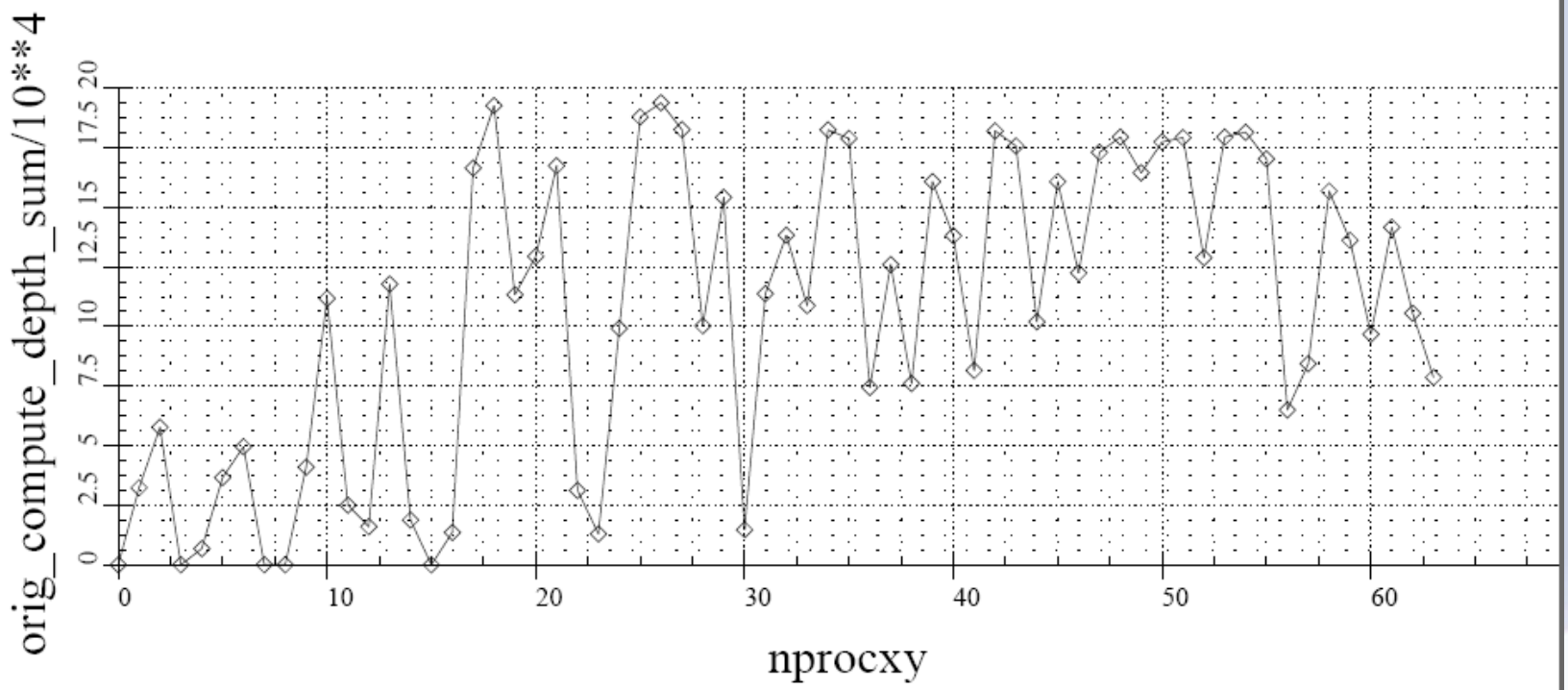


# Load Balancing




ECMWF, 2010

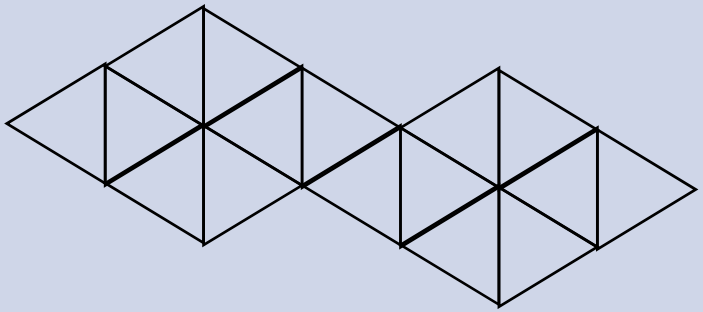
# Load Imbalance



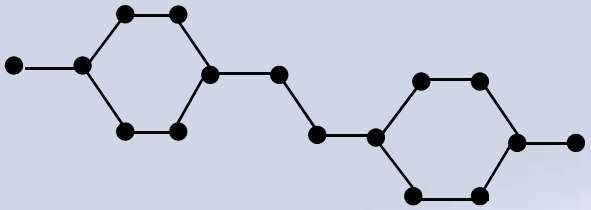

orig_compute_depth_sum

# Mapping Mesh ⇨ Graph

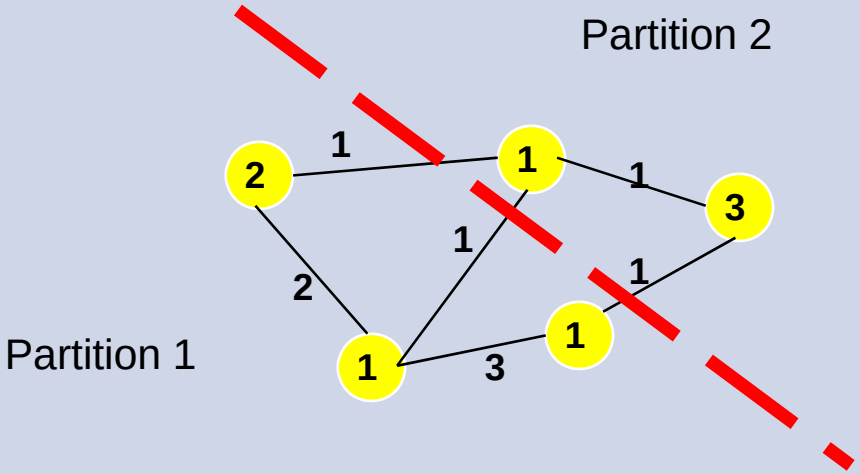
Mesh



Graph

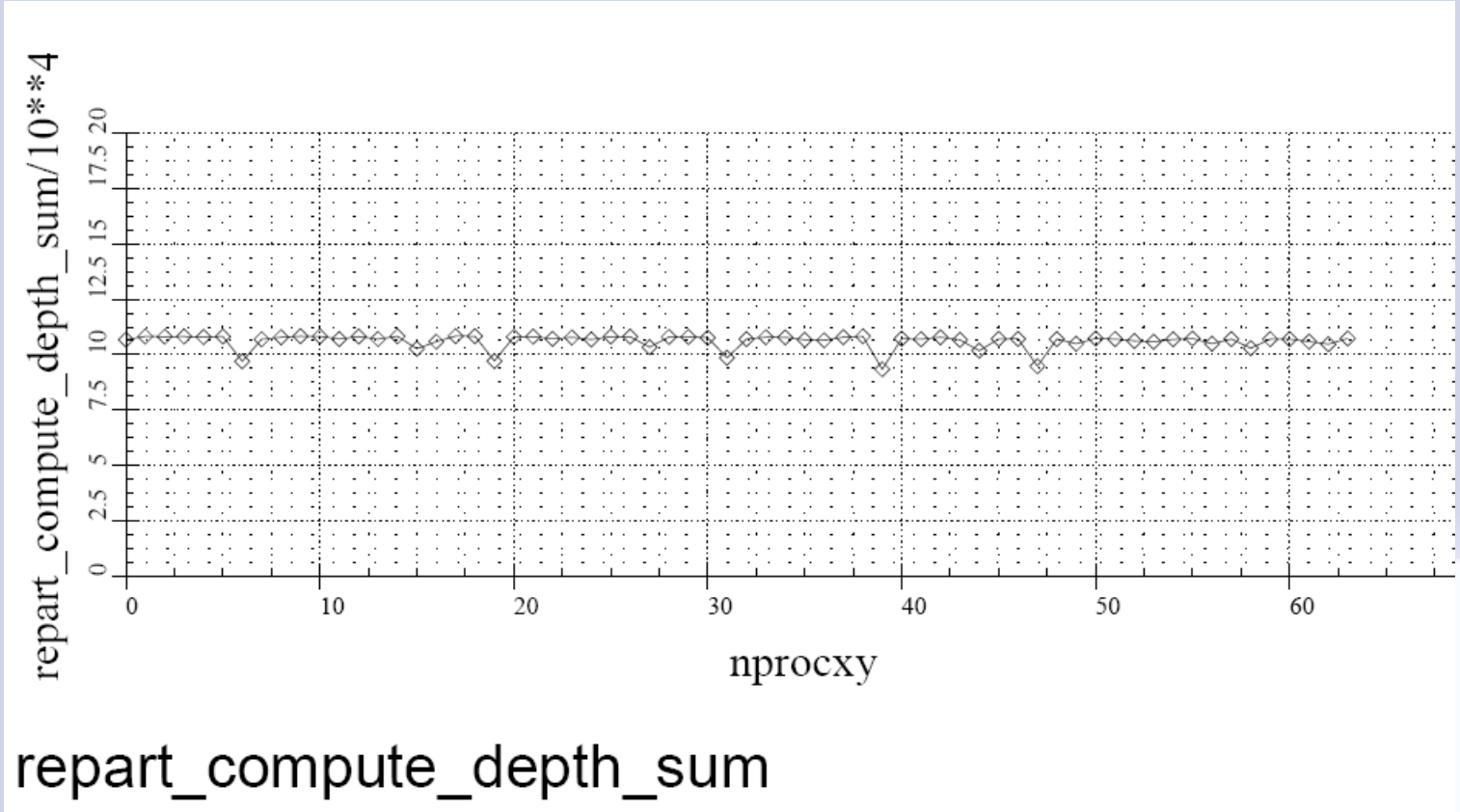


# Graph Partitioning



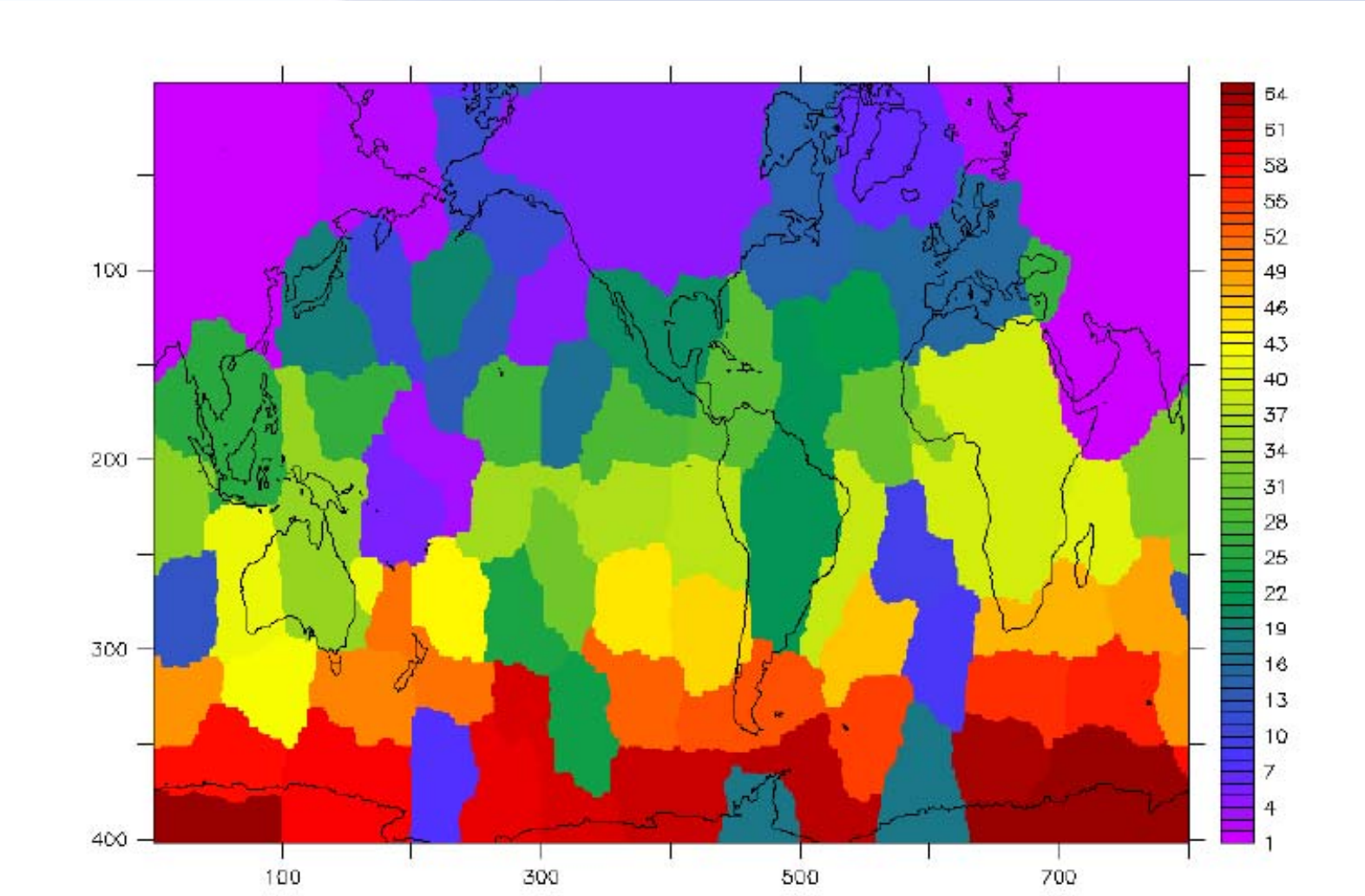

load balance: 100 %
weight of cut edges: 3

# Static Loadbalacing with METIS



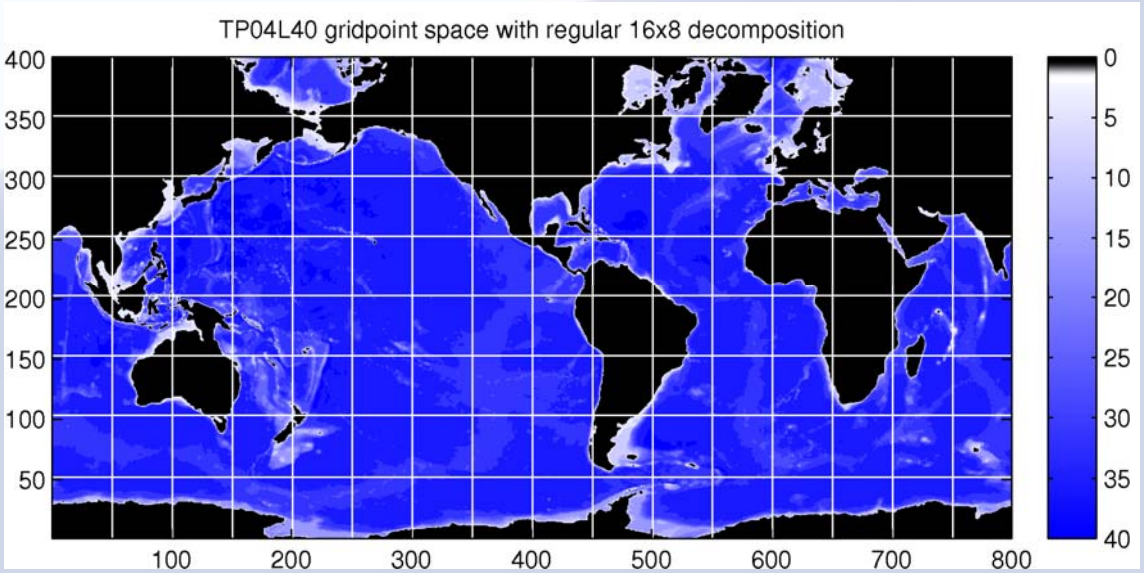

repart_compute_depth_sum

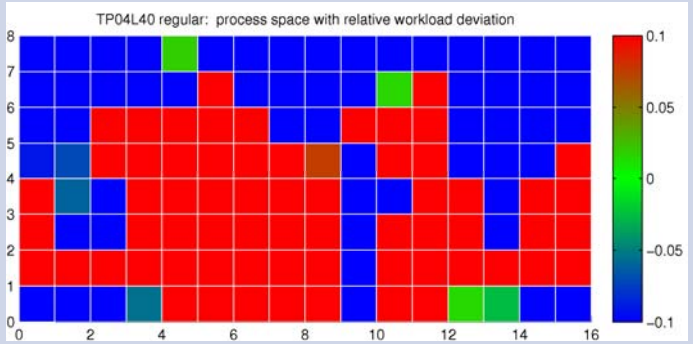
# Problem of Graph Partitioning

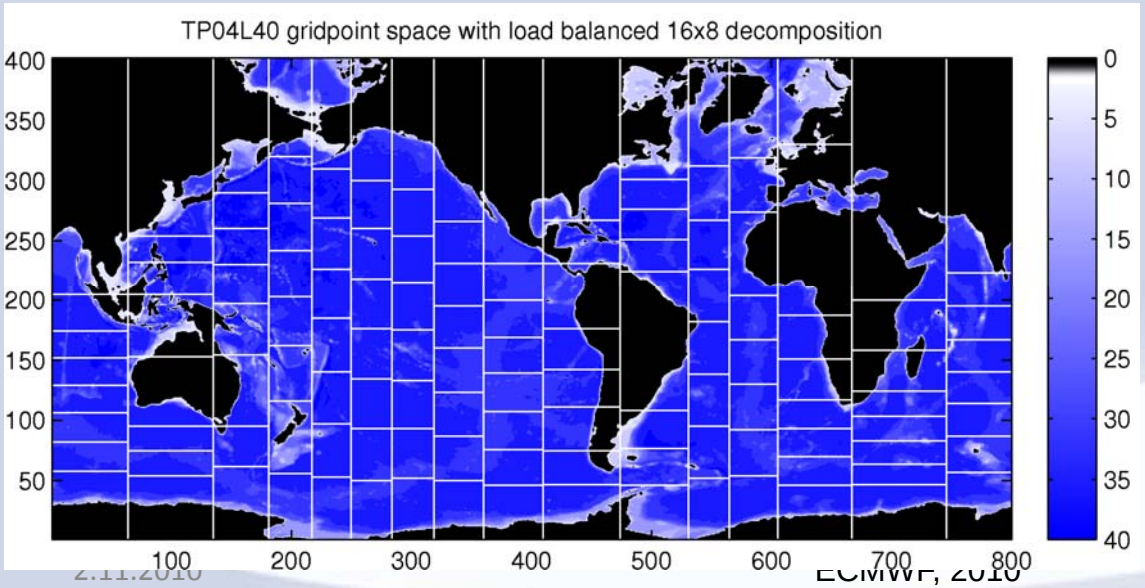


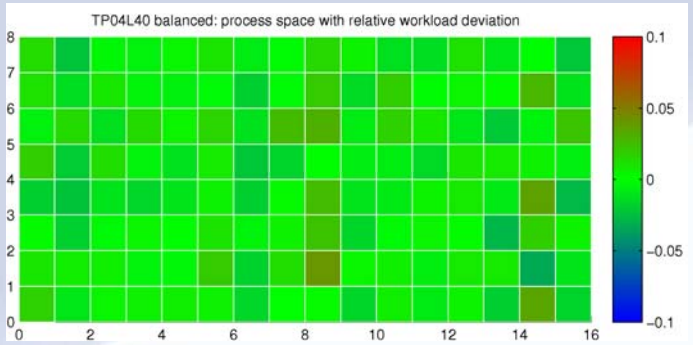

ECMWF, 2010

# Loadbalancing



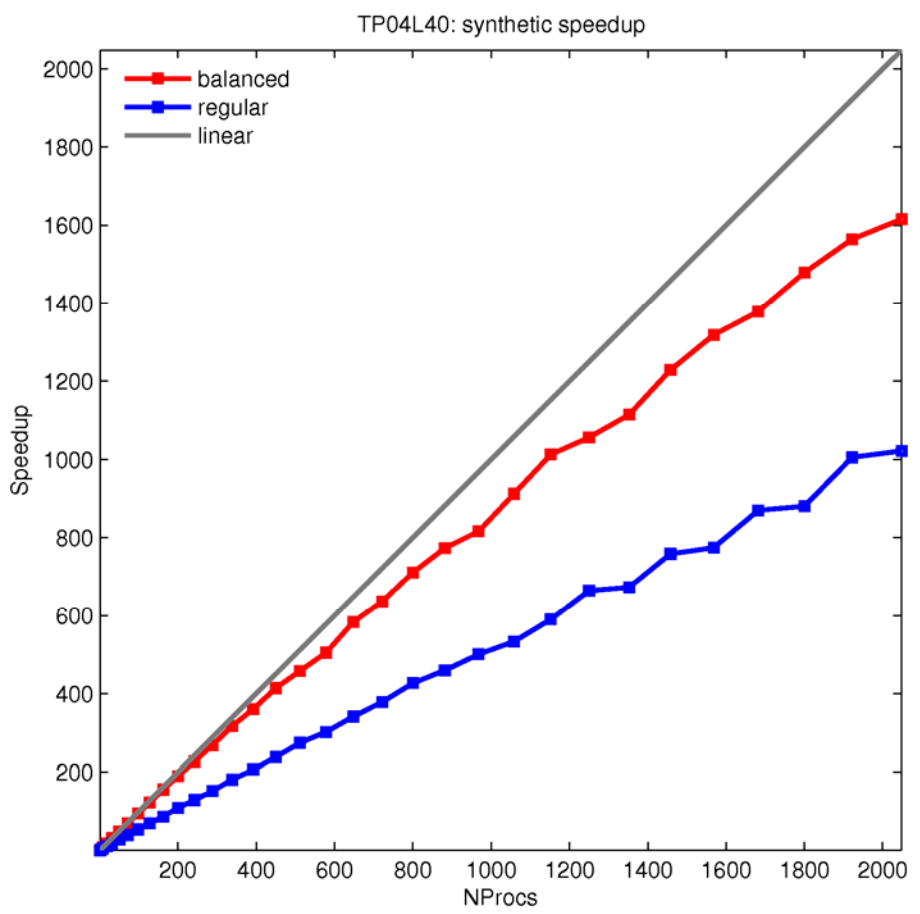



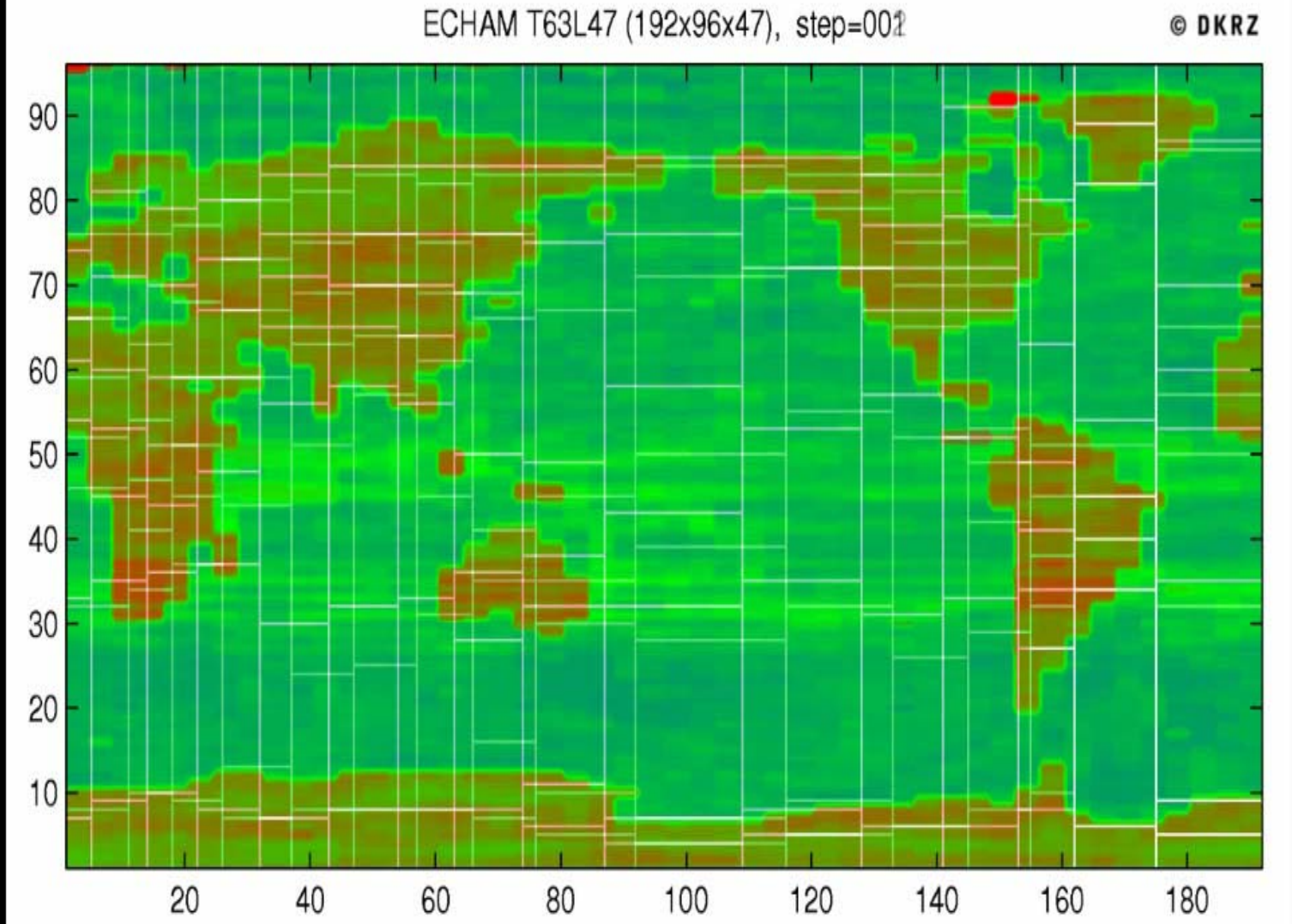
ScalES

GEFÖRDERT VOM
Bundesministerium
für Bildung
und Forschung

# Loadbalancing



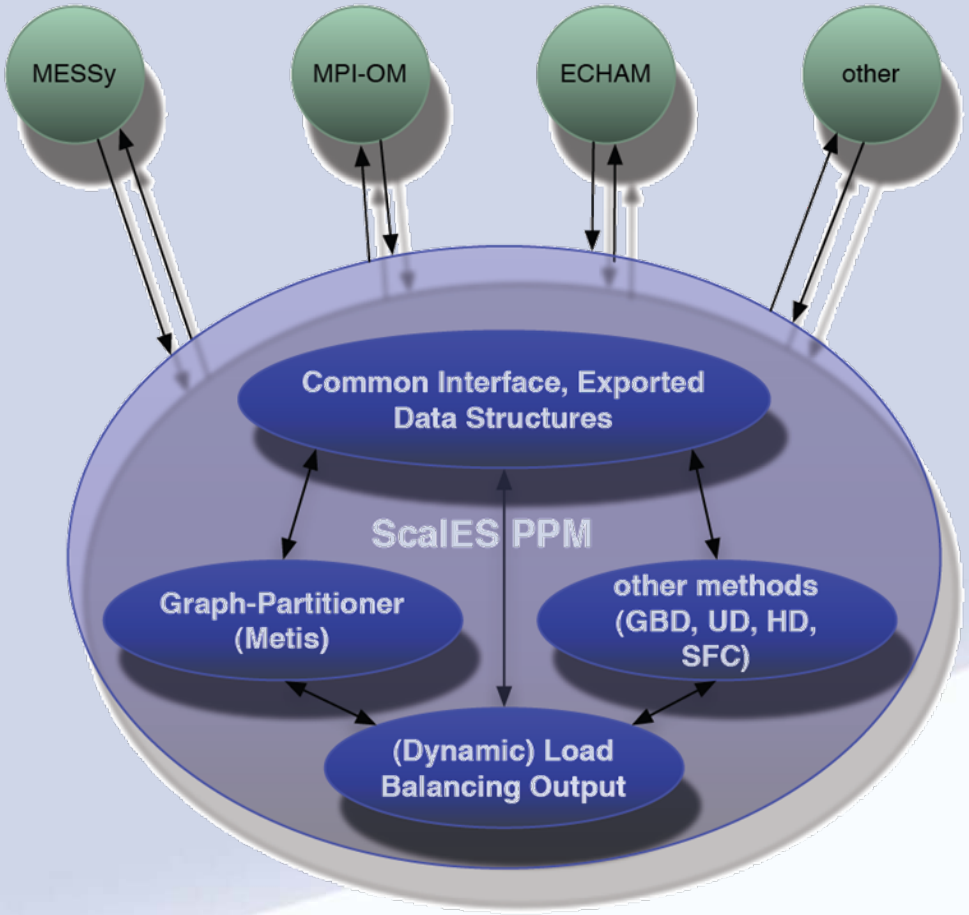
ScalES


ECHAM T63L47 (192x96x47), step=001
© DKRZ
90
80
70
60
50
40
30
20
10
20
40
60
80
100
120
140
160
180

# ScalES PPM (Parallel Partioning Modul)

**Features:**

- Describe partitioning in model-independent data structures.
- Provide convenient API to multiple partitioning algorithms.
- Support data relocation for repartitioning
- Provide solid foundation for other partitioning dependent functionality.




**ScalES PPM schematic library design**

# Optimization

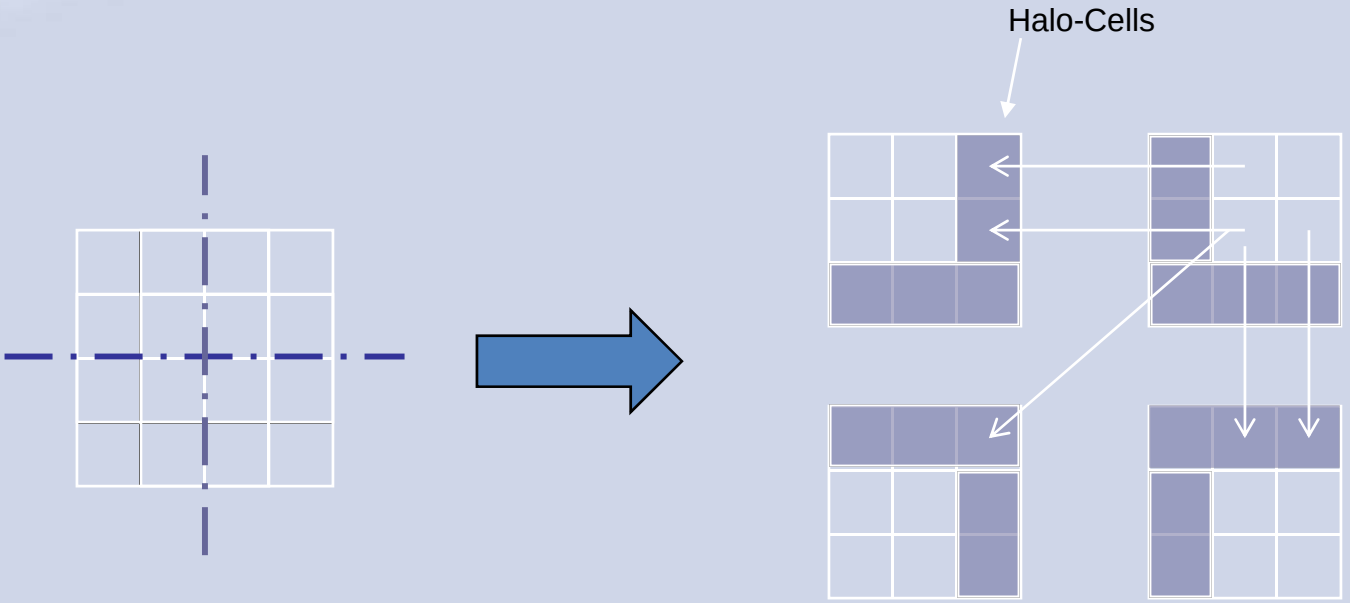
Computation vs Communication

- Expanding Halos
  - More Local Operations
  - Less Communication

CG Method with appropriate Preconditioner

- Faster Convergence =>
  - Less Iterations
- Less Communication

# Structure of a partitioned mesh

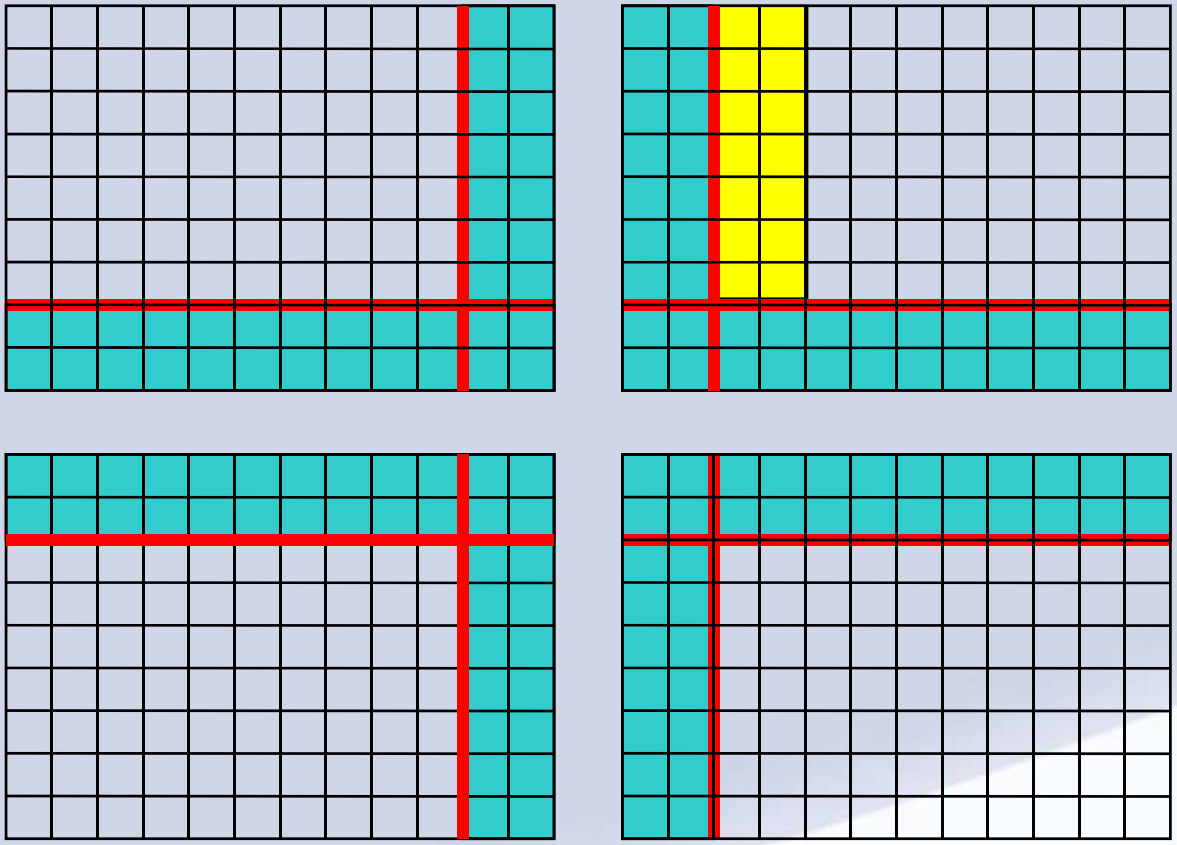



Global mesh

Partitioned mesh

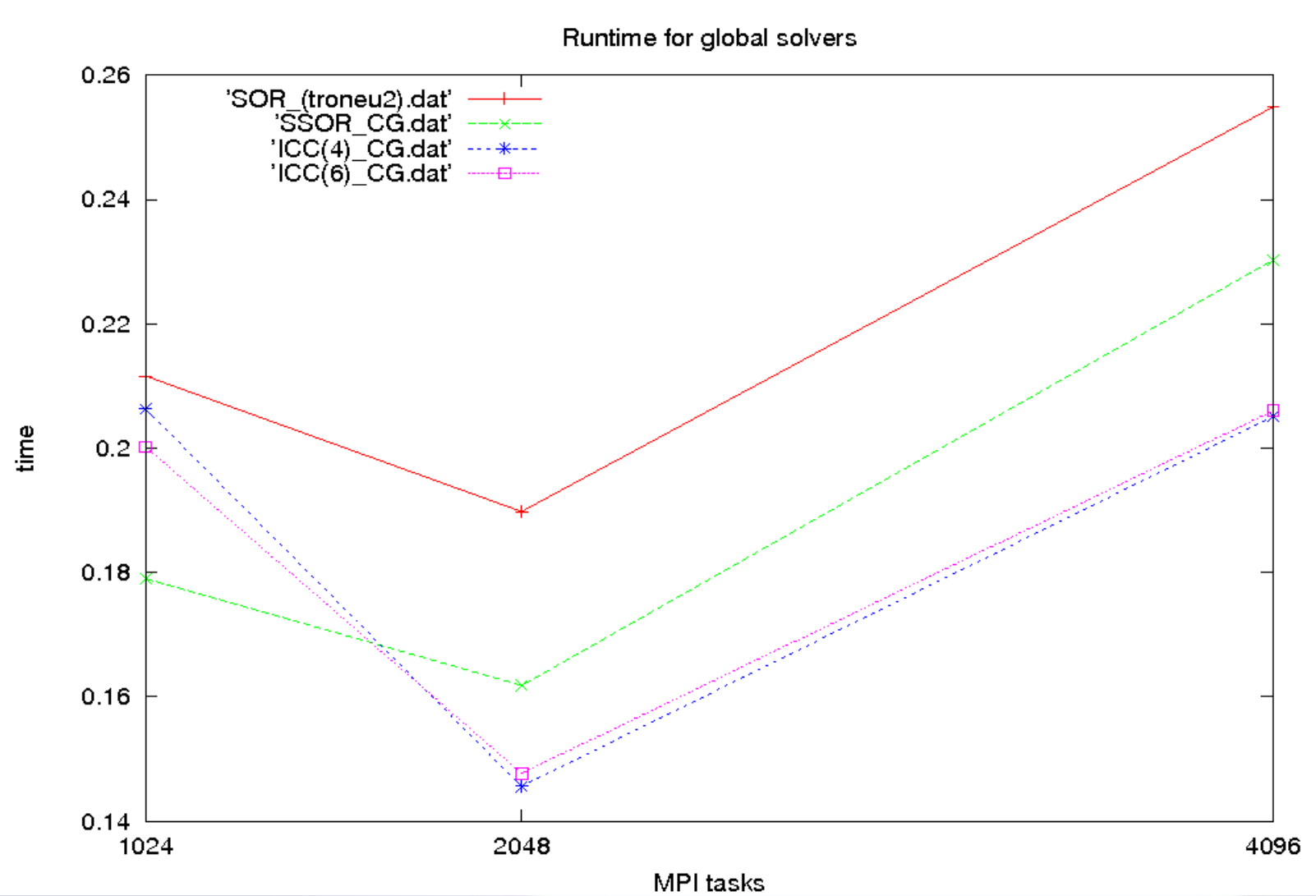
Exchange of halos at each iteration

# Expanding Halos



Exchange of expanded halos every 2nd iteration

# Library of Solvers (KIT)



# Optimization

- Compilers cannot optimize automatically everything
- Optimization is not just finding the right compiler flag
- Major algorithmic changes are necessary
- Solutions must be generic otherwise the whole process is a “Sisyphean Task”

ScalES

GEFÖRDERT VOM

Bundesministerium für Bildung und Forschung

# Scalable Earth-System Models

- **Project:**
  - Facilitate High Productivity Climate Simulations
  - Three year program
  - Funded by Federal Ministry for Science and Research (01IH08004E)
  - Started in January 2009

- **Partners:**
  - DKRZ (German Climate Computing Centre)
  - MPIM (Max-Planck-Institute for Meteorology)
  - MPIC (Max-Planck-Institute for Chemisty)
  - AWI (Alfred-Wegner-Institute for Polar Research)
  - KIT (Karlsruhe Institute for Technology)
  - IBM (International Business Machines)

# ScalES:

## Use case: COSMOS

- ECHAM5 (global atmosphere)
- MESSY/MECCA (atmospheric chemistry)
- MPI-OM (global ocean)
- OASIS4 (coupler)

## Work Packages

- Parallel I/O
- Load Balancing
- Architectural issues
- Coupling (e.g. of atmosphere and ocean models)

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung