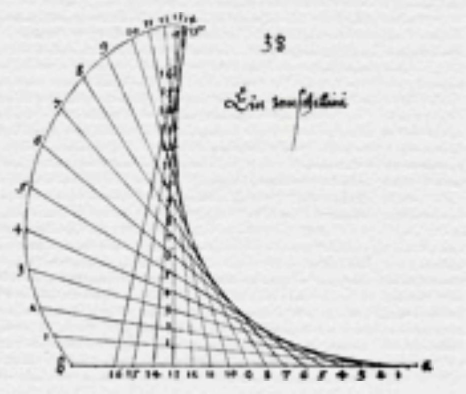
Software-Framework ug 4 - Geometric MultiGrid Scaling

I. Heppner, S. Reiter, **A. Vogel**, G. Wittum

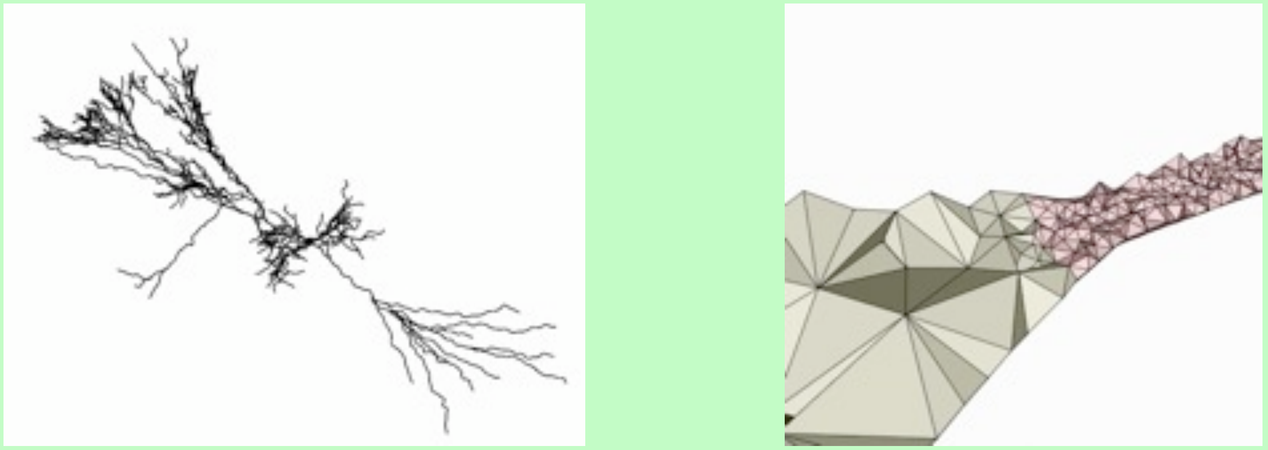
GCSC, Goethe-University of Frankfurt, Germany

NuSim - Meeting, Darmstadt, 16. April 2012

Andreas Vogel
G-CSC
University of Frankfurt

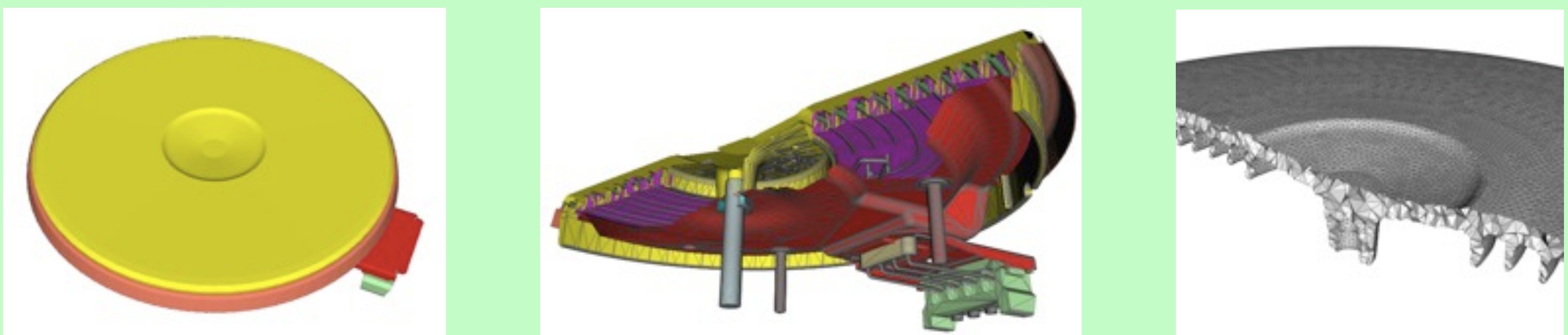


Motivation



Neuron network and volume geometry.

Examples of work of our group.



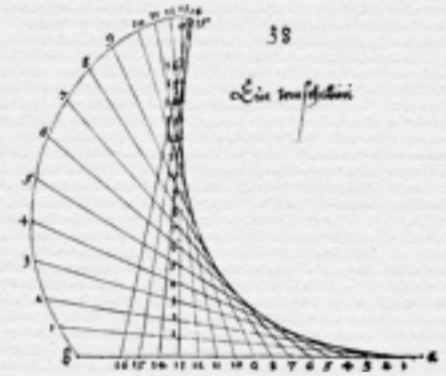
CAD- and volume geometry of heating plate.

Andreas Vogel
G-CSC
University of Frankfurt

Outline

- Programm Framework
  - *Parallel Communication Layer (pcl)*
  - *Distributed Grids*
  - *Distributed Algebra for Multigrid*
- Scaling Study
  - Weak scaling of Laplace Problem in 2d
  - Weak scaling of Laplace Problem in 3d
- Real world Problem: Density Driven Flow

Andreas Vogel
G-CSC
University of Frankfurt



Software Framework

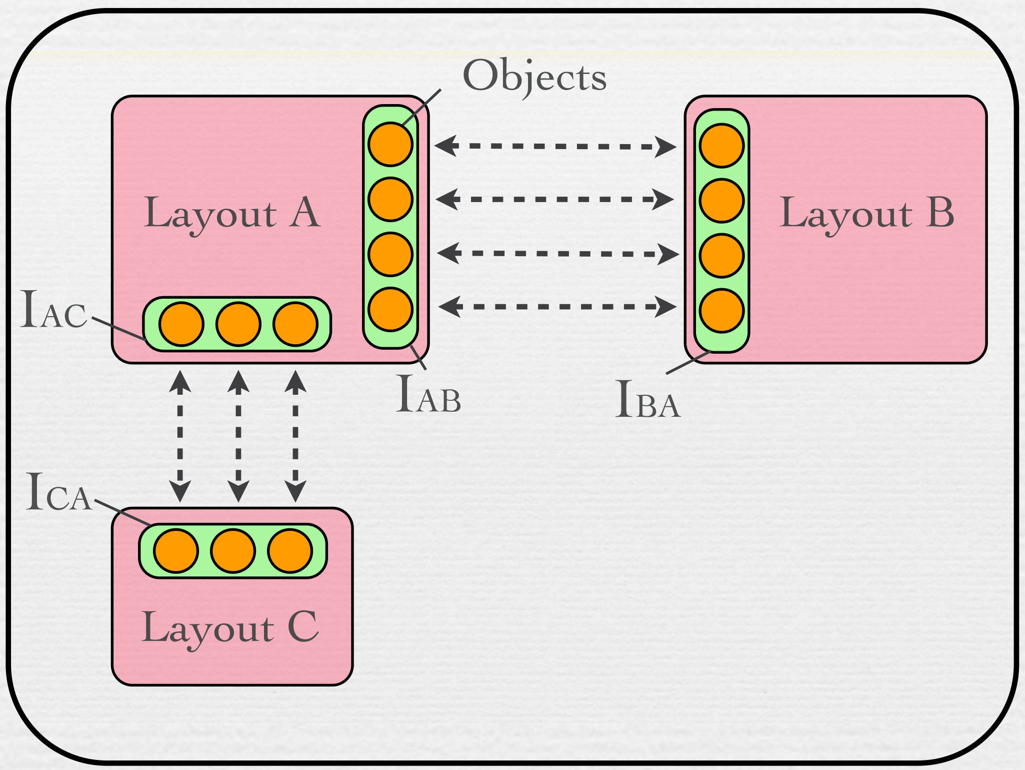
- Software Framework ug („unstructured grids“) for solution of partial differential equations, general purpose library

- Novel implementation ug 4**:**
  - Grids and Algebra - completely independent
  - Algebra structures using cache aware storage (CRS)
  - Parallel Communication Layer (pcl) - based on MPI

Andreas Vogel
G-CSC
University of Frankfurt

Parallel Communication Layer (pcl)

Abstract handling of arbitrary objects (e.g. grid elements, algebra indices) that need connection to copies on other processes



pcl infrastructure

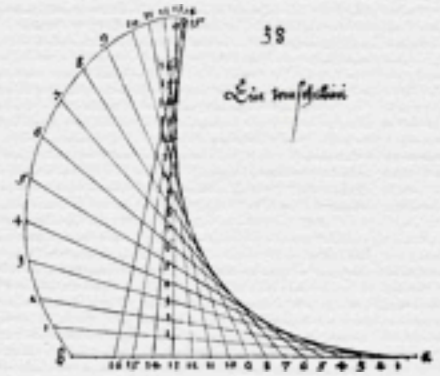
- Objects are grouped in interfaces,
- Interfaces connect groups of elements on different processes,
- Interfaces are grouped in Layouts,
- Communication can be scheduled for interfaces or layouts.

Andreas Vogel
G-CSC
University of Frankfurt

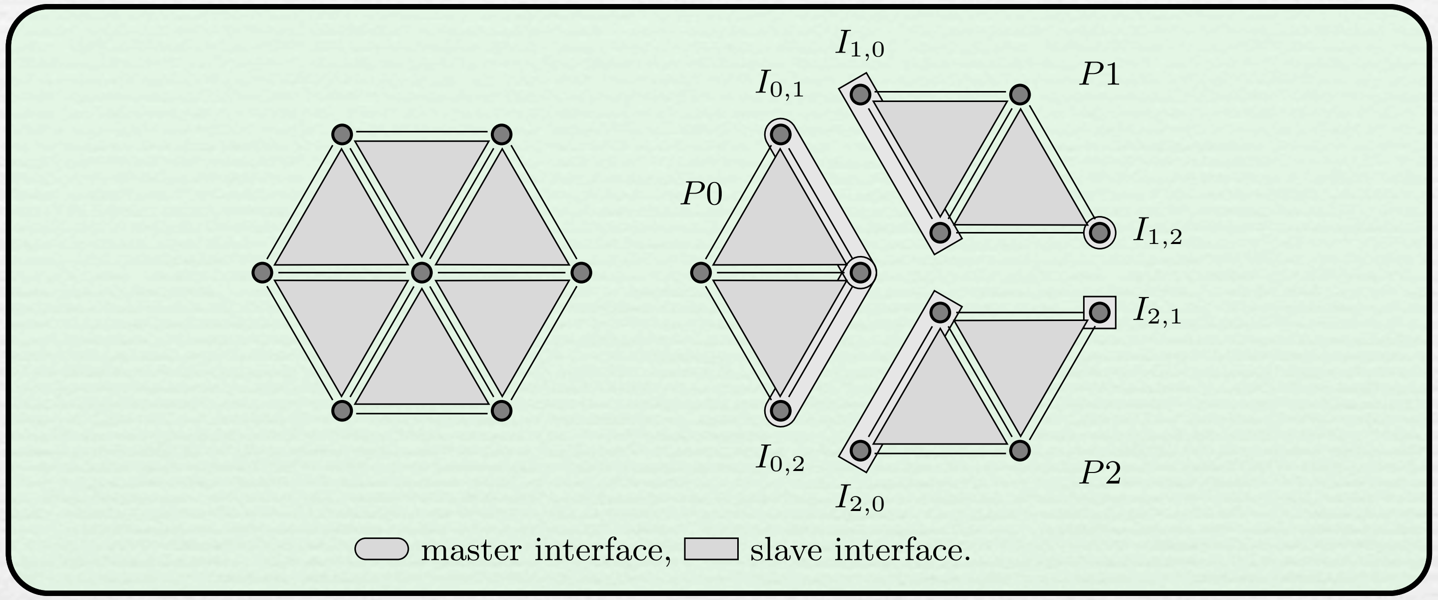
Parallel Communication Layer (PCL)

- Template library for point-to-point communication between abstract object sets.
- Highly adoptable to different graph-structures (e.g. grids or algebra).
- Minimal storage overhead - only references to interfacing objects are stored.
- Identification of objects through local order in process-interfaces. No global IDs required - but can still be generated on request.

Andreas Vogel
G-CSC
University of Frankfurt



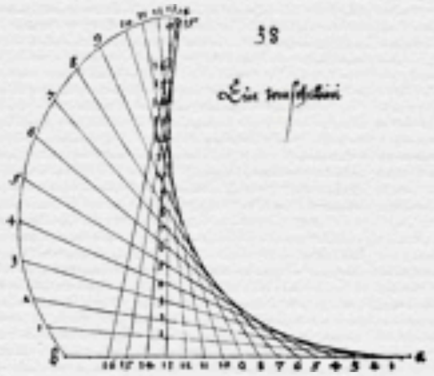
Horizontal Grid Layout



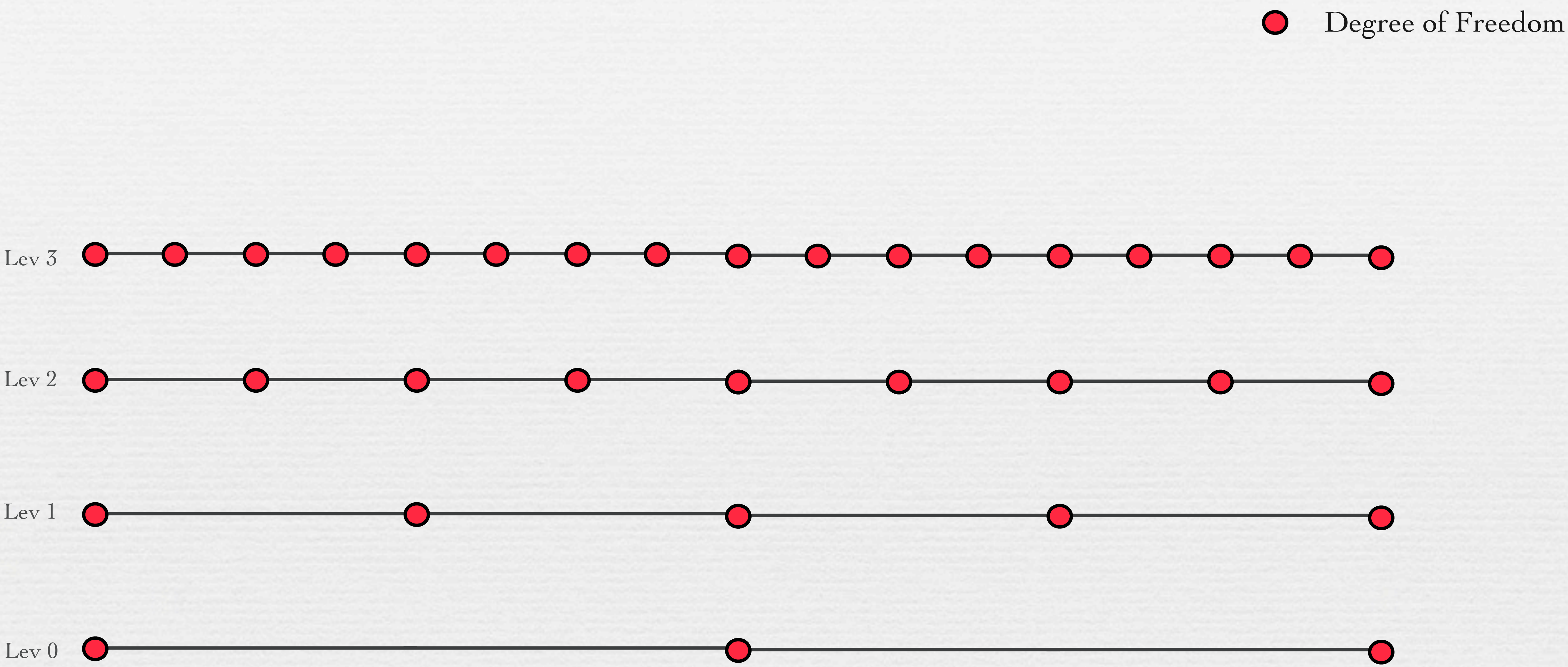
Serial- (left) and distributed-grid (right). Vertex-interfaces are depicted. (S. Reiter)

- Distinction in ***master***- and ***slave***-interfaces.
- Communication *master*→*slave* or *slave*→*master* (not *slave*→*slave*).
- Separate layouts for ***vertices***, ***edges***, ***faces*** and ***volumes*** on each level.
- Allows communication only within one grid level.

Andreas Vogel
G-CSC
University of Frankfurt



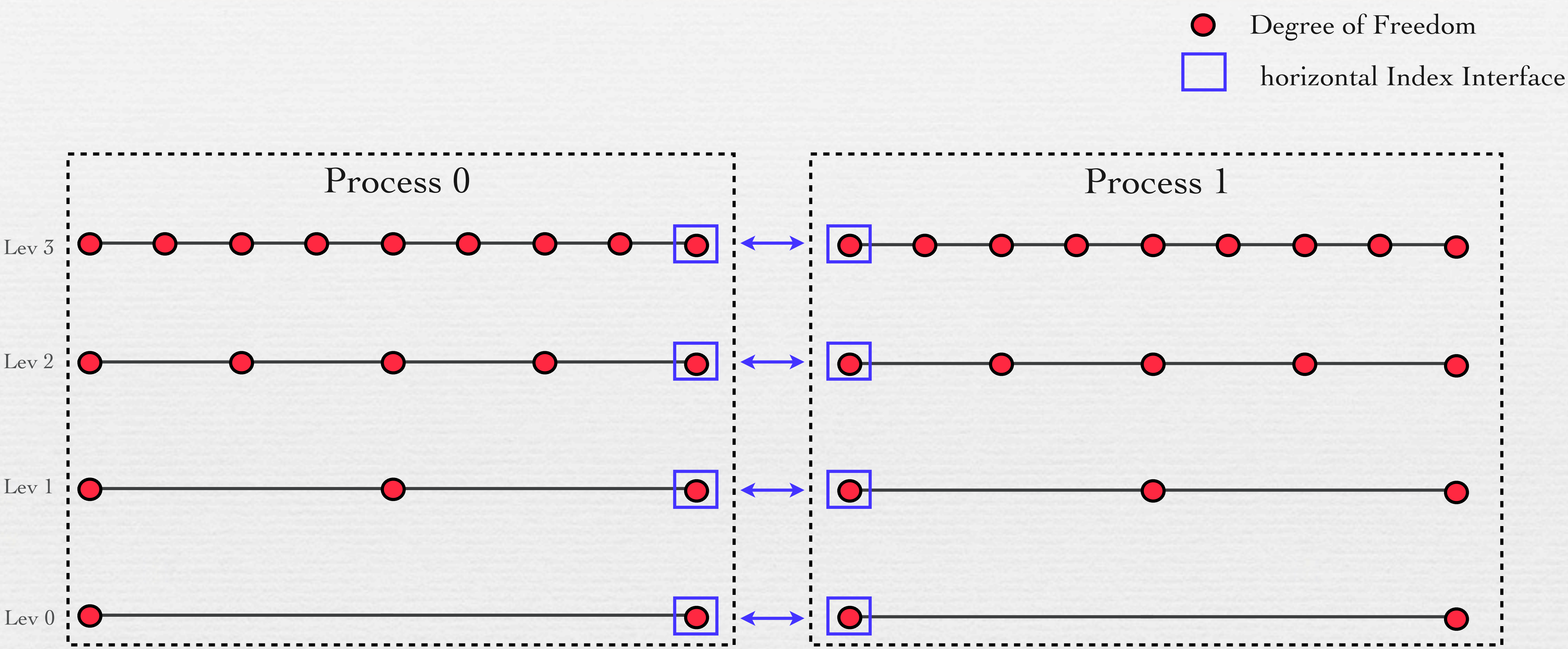
MultiGrid Layout Serial



1D Example: Coarse grid with 2 elements, the grid levels are produced using uniform refinement.

Andreas Vogel
G-CSC
University of Frankfurt

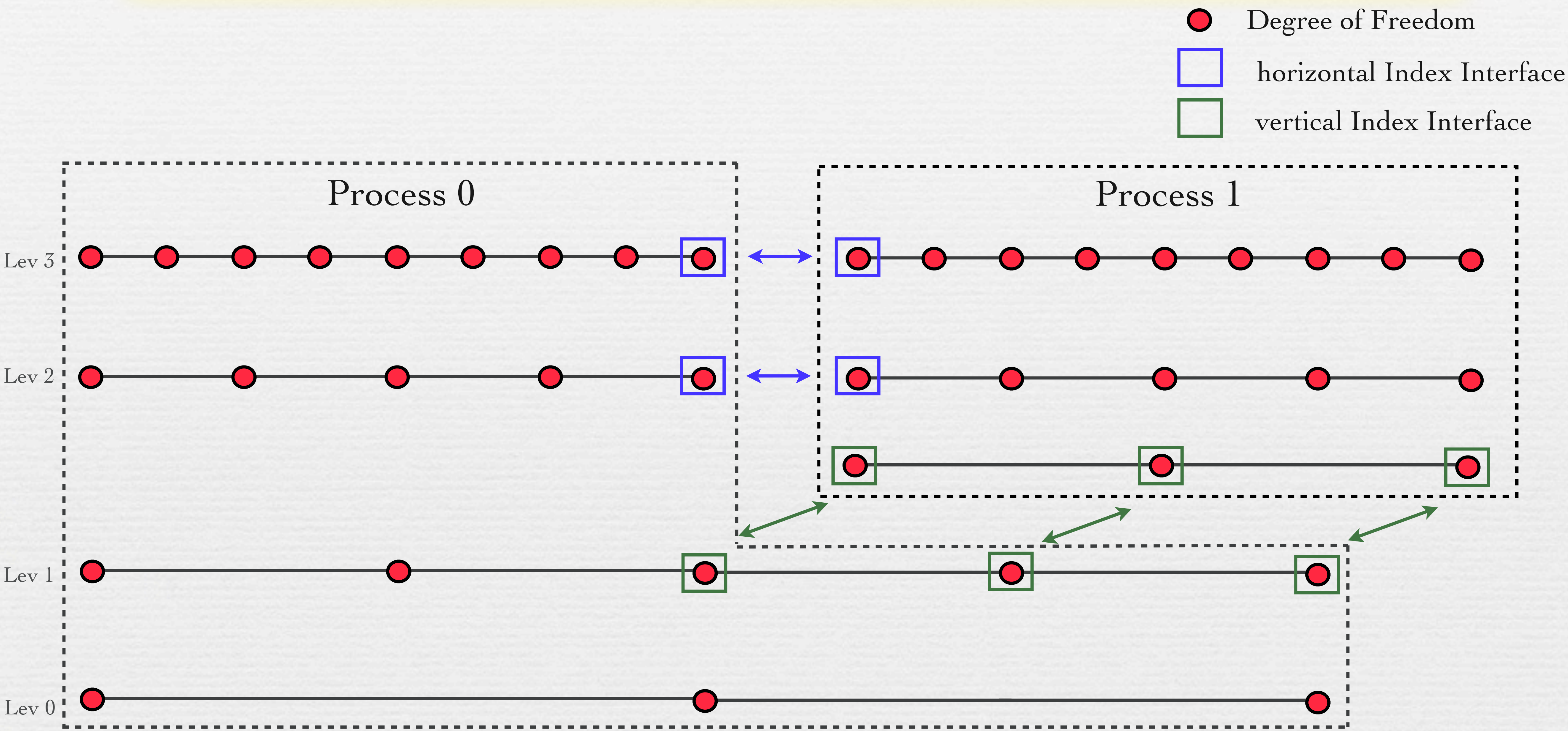
MultiGrid Layout*



1D Example for 2 PE: Starting with one element on each of PE, the grid levels are produced using uniform refinement.

Andreas Vogel
G-CSC
University of Frankfurt

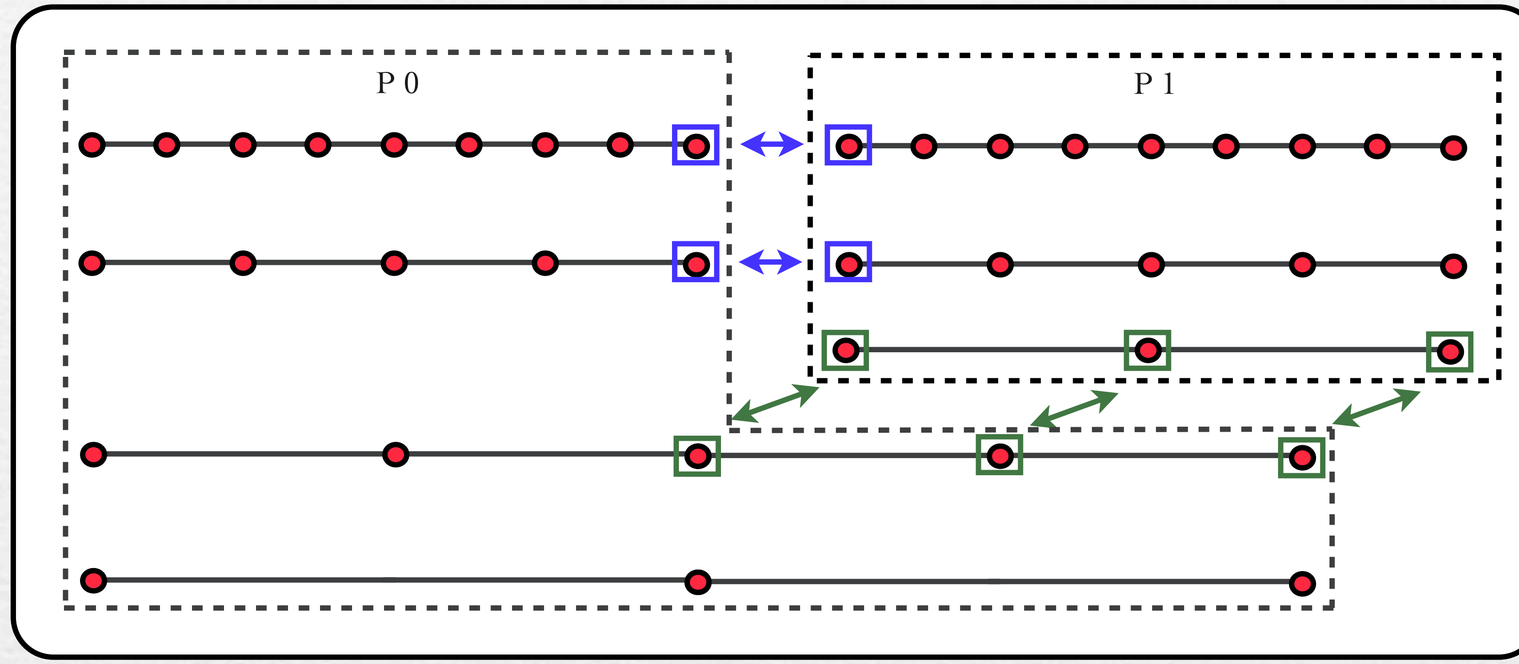
MultiGrid Layout



Andreas Vogel
G-CSC
University of Frankfurt

Multi-Grid Distribution

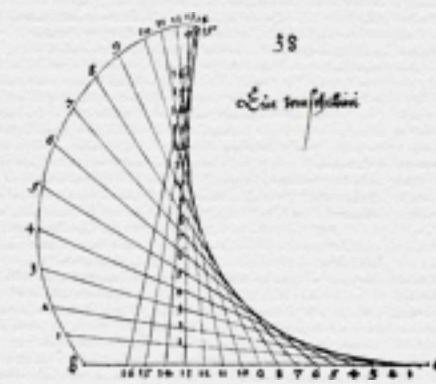
- Idea:
  - Load coarse grid on 1 proc,
  - perform refinement,
  - distribute top level,
  - perform further refinement,
  - solve



Distribution of a multi-grid

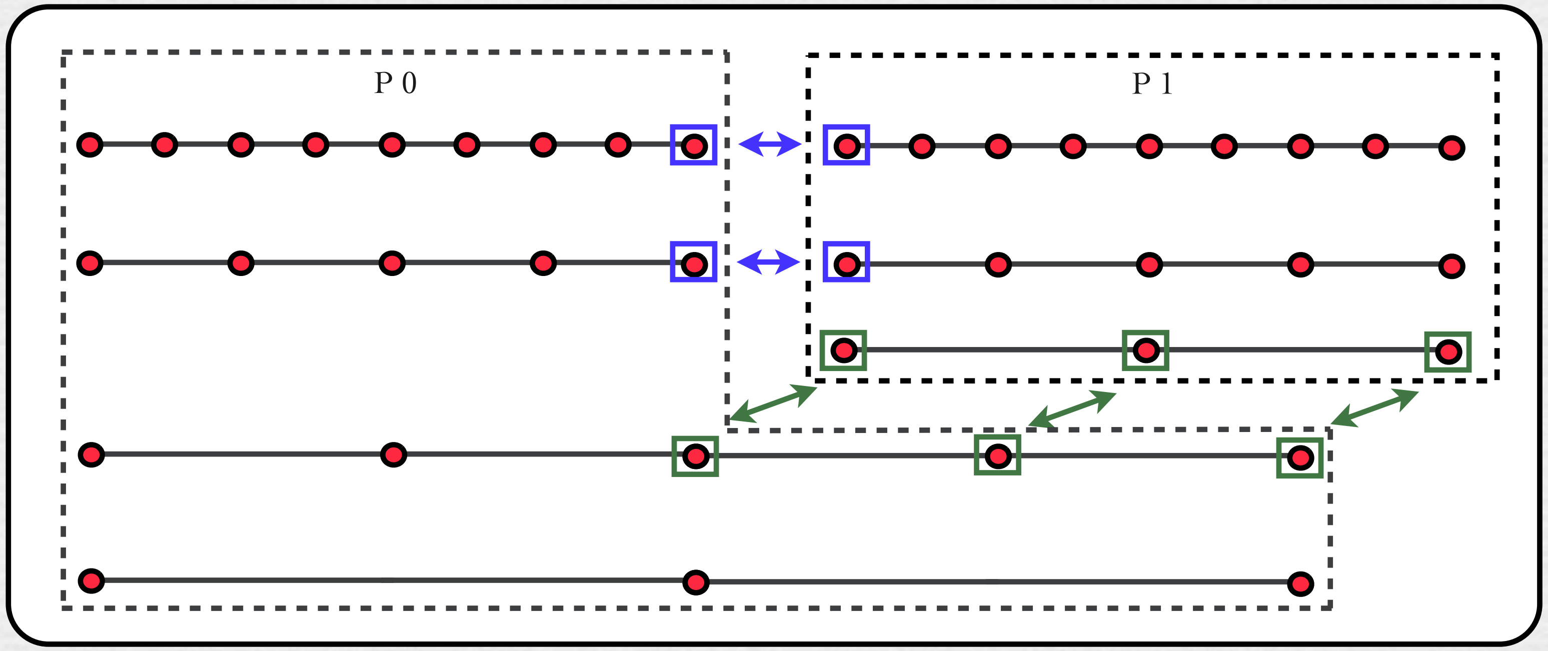
- Observation:
  - Involves a *one-to-all* communication of big data chunks during startup.
  - Restriction / Prolongation involves *all-to-one* and *one-to-all* communication.

Andreas Vogel
G-CSC
University of Frankfurt

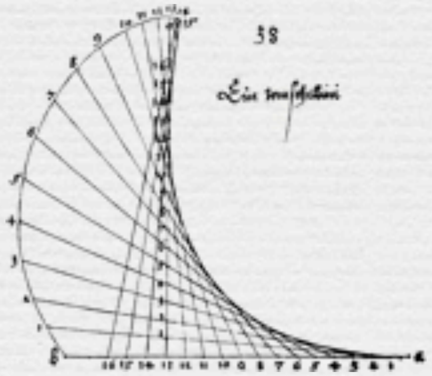


MultiGrid and Interfaces

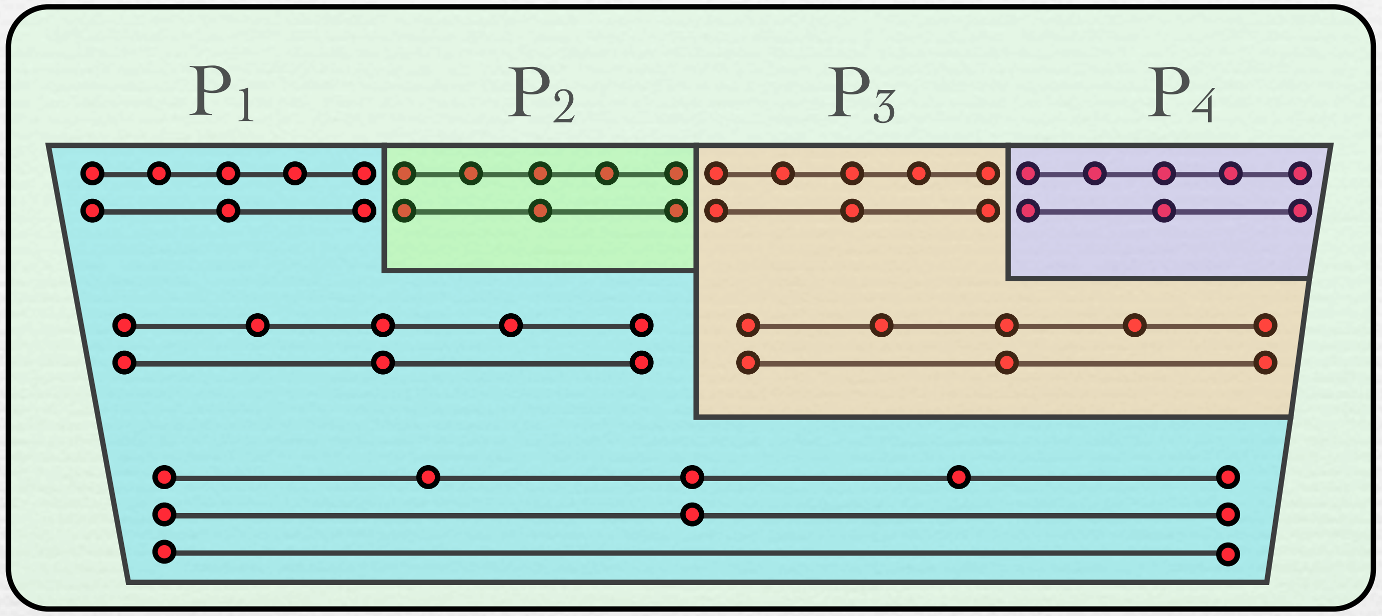
- Vertical interfaces are required to allow distribution of parts of a ***multi-grid hierarchy***.
- ***Smoothing*** using horizontal interfaces.
- ***Prolongation*** / ***Restriction*** using vertical interfaces.



Andreas Vogel
G-CSC
University of Frankfurt

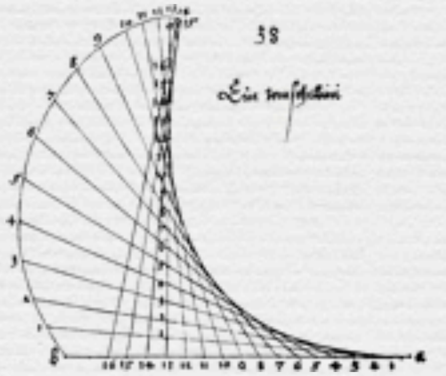


Hierarchical Redistribution

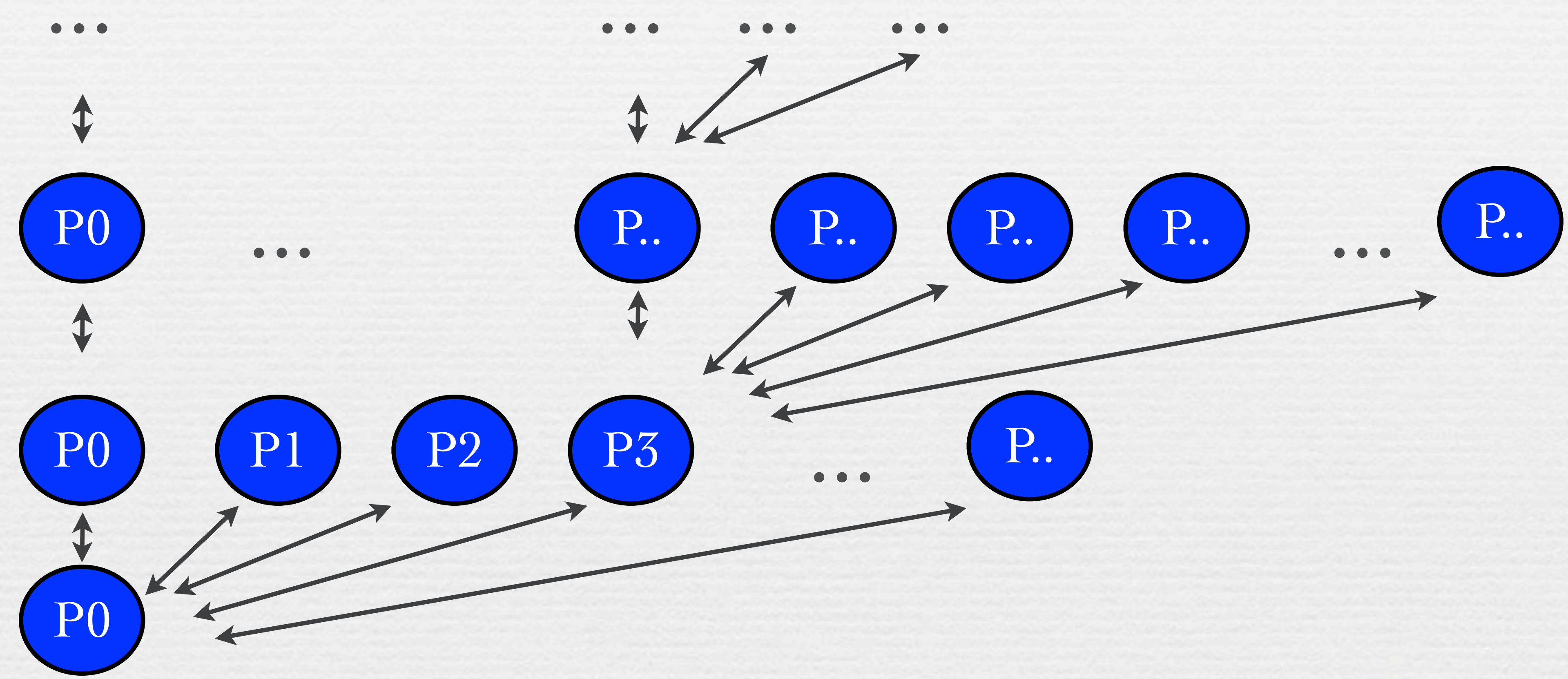


- Idea:
  - load grid on one process,
  - refine grid on all processes, which have a grid,
  - distribute top level to some free processes,
  - iterate...

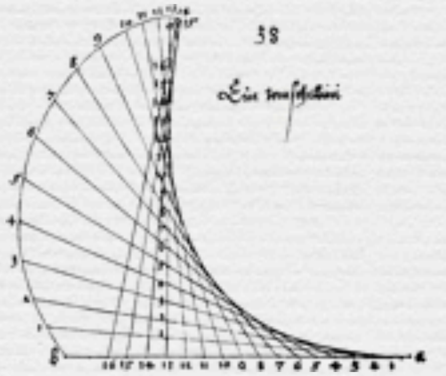
- Infinitely many vertical interfaces



Hierarchical Redistribution



- Tree of agglomarations of processes



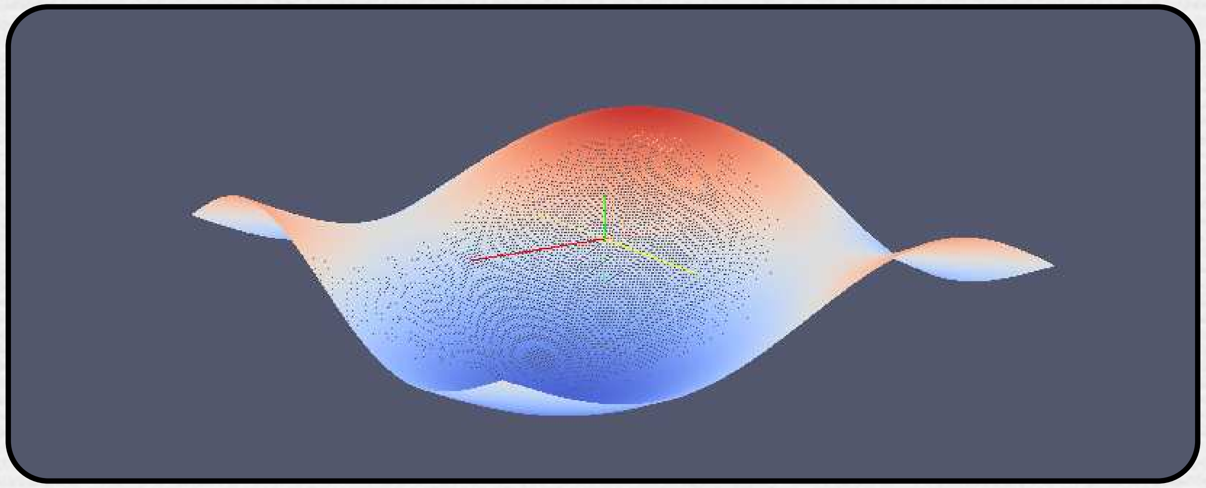
Andreas Vogel
G-CSC
University of Frankfurt

MultiGrid

- **Geometric Multi - Grid Solver:**
  - Coarse Grid Matrices assembled on coarser grids
  - usual Prolongation / Restriction, taking into account the vertical interfaces
  - Smoother: Jacobi, Gauss-Seidel, ILU, ...
  - Coarse Problem Solver: LU - Factorization or iterative linear solvers

Scaling Tests

- Model problem: $-\Delta u = f$ on $\Omega = [0,1] \times [0,1]$,
- with $f(x,y) = (2\pi)^2(\sin(2\pi x) + \sin(2\pi y))$,
- and $u(x,y) = \sin(2\pi x) + \sin(2\pi y)$ on $\partial\Omega$,
- Analogical for 3d,
- Calculations were performed on *JuGene, FZ Jülich, Germany*,
- Discretization: vertex-centered finite volume,
- Solver: *geometric multi-grid method* with *Jacobi* smoother.



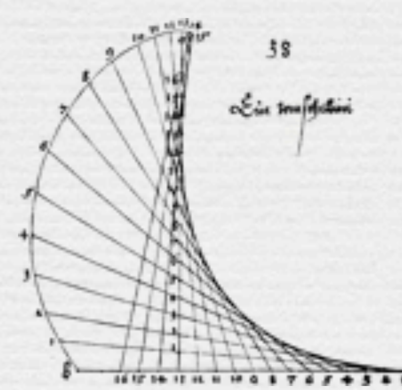
Andreas Vogel
G-CSC
University of Frankfurt

Weak scaling 2d

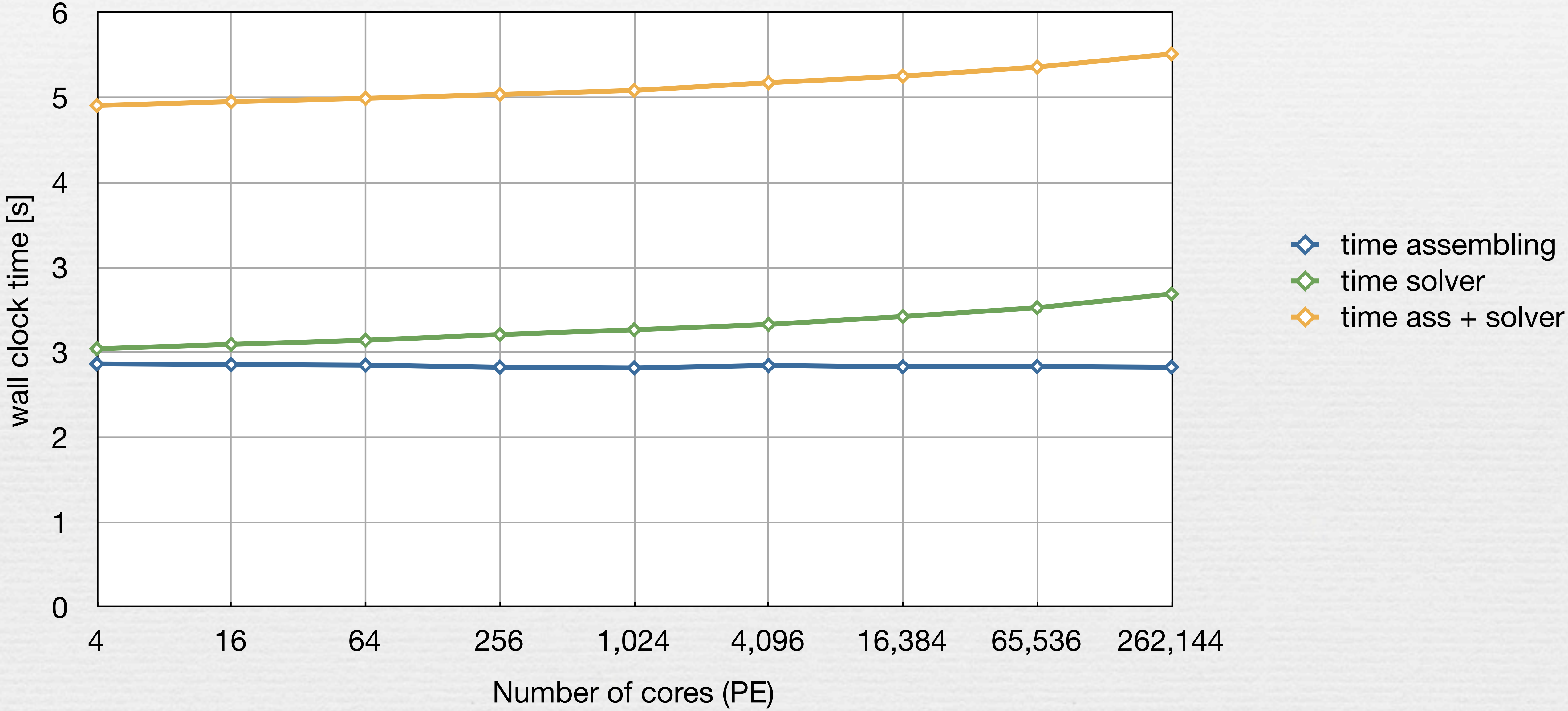
#PE	level	#DoF	T_{total} (s)	$T_{assemble}$ (s)	T_{solver} (s)	$T_{ass+solve}$ (s)	Efficiency (%) (ass+solve)	Speedup (ass+solve)	Speedup ideal	Efficiency (%) (solve)	Speedup (solve)
4	6	263,169	13.242	2.456	2.608	5.064	--	--	--	--	--
16	7	1,050,625	13.305	2.449	2.653	5.102	99.2	4.0	4	98.3	3.9
64	8	4,198,401	13.416	2.443	2.694	5.136	98.6	15.8	16	96.8	15.5
256	9	16,785,409	13.677	2.423	2.752	5.175	97.8	62.6	64	94.8	60.6
1,024	10	67,125,249	16.053	2.416	2.800	5.216	97.1	248.5	256	93.1	238.4
4,096	11	268,468,225	18.724	2.440	2.854	5.294	95.7	979.5	1,024	91.4	935.7
16,384	12	1,073,807,361	20.787	2.427	2.934	5.360	94.5	3,869.6	4,096	88.9	3,641.5
65,536	13	4,295,098,369	23.844	2.430	3.023	5.452	92.9	15,216.6	16,384	86.3	14,136.3
262,144	14	17,180,131,329	61.612	2.423	3.162	5.585	90.7	59,424.0	65,536	82.5	54,051.5

Scaling study for the weak scaling of the laplace problem on a unit square $[0,1]^2$. Initial grid with 8x8 quadrilaterals and uniform refinement for each grid level. Solving of the linear equation system is done using a geometric multigrid solver with Jacobi smoother. All computations need 10 iterations until a defect norm of 10^{-12} is reached. No problem specific optimizations have been made. Load per process ~65,500 DoFs.
(Abbreviations are: PE = Processing entities (cores), DoF = Degrees of Freedom; T_{total} = total run time, $T_{assemble}$ = time for assembling of system matrix and coarse grid matrices, T_{solver} = time for solver; parallel Speedup $S(P_4, P_i) = (T_4 \cdot P_i)/(T_i \cdot P_4)$, Efficiency $E = T_4/T_i$).

Andreas Vogel
G-CSC
University of Frankfurt

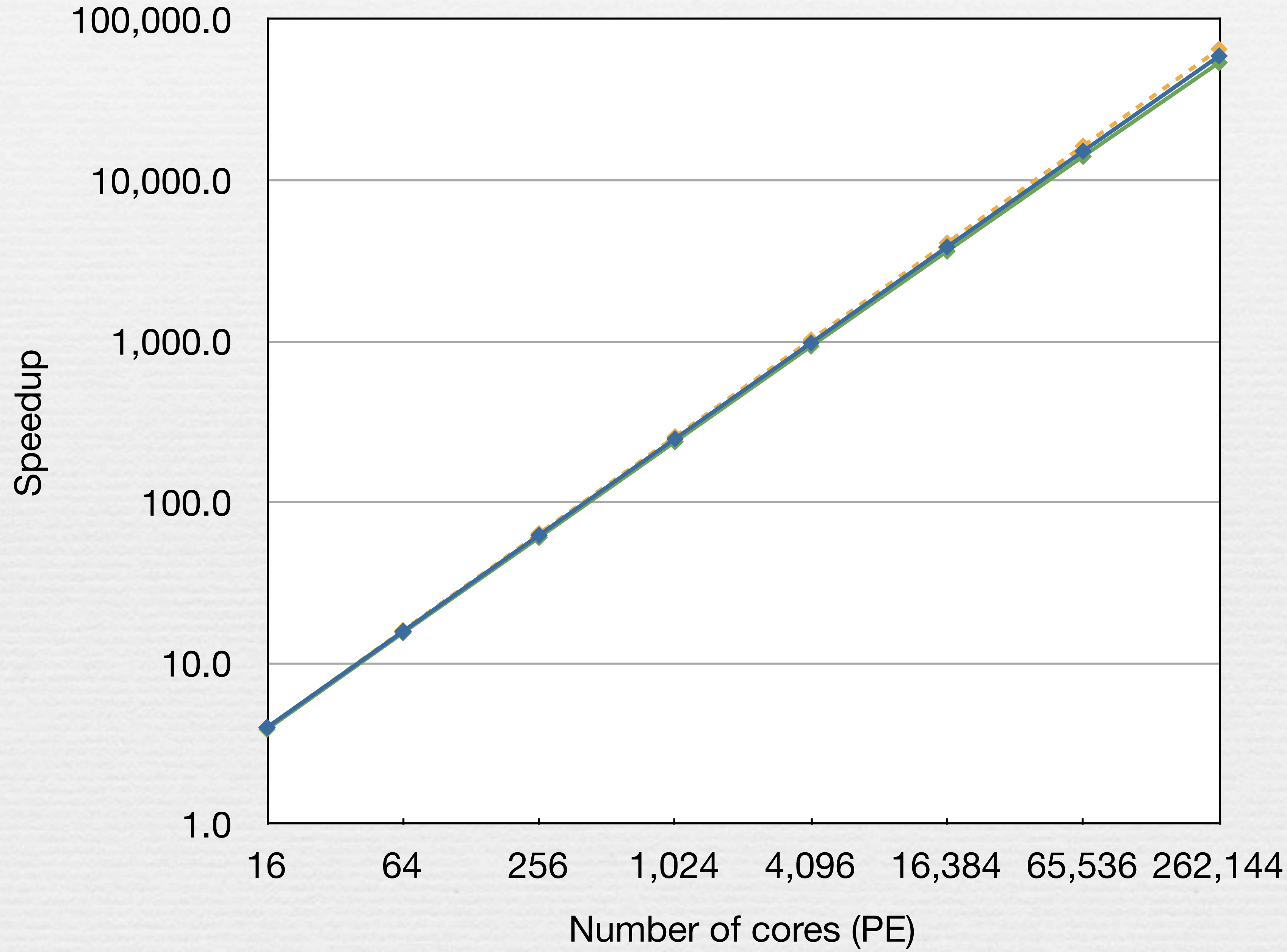


Weak scaling 2d

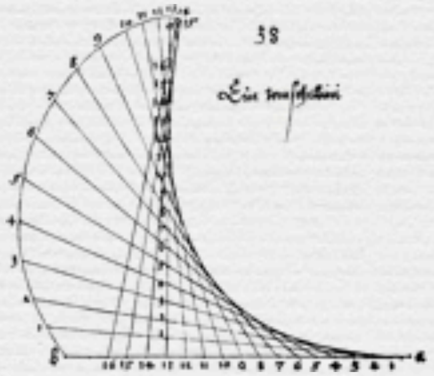


Andreas Vogel
G-CSC
University of Frankfurt

Weak scaling 2d



Andreas Vogel
G-CSC
University of Frankfurt

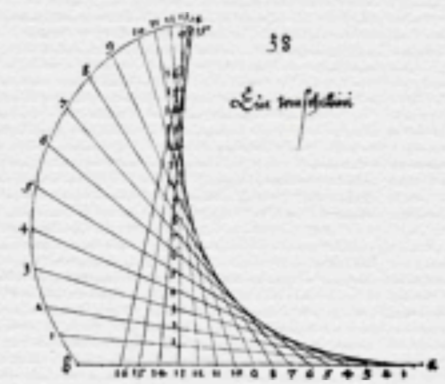


Weak scaling 3d

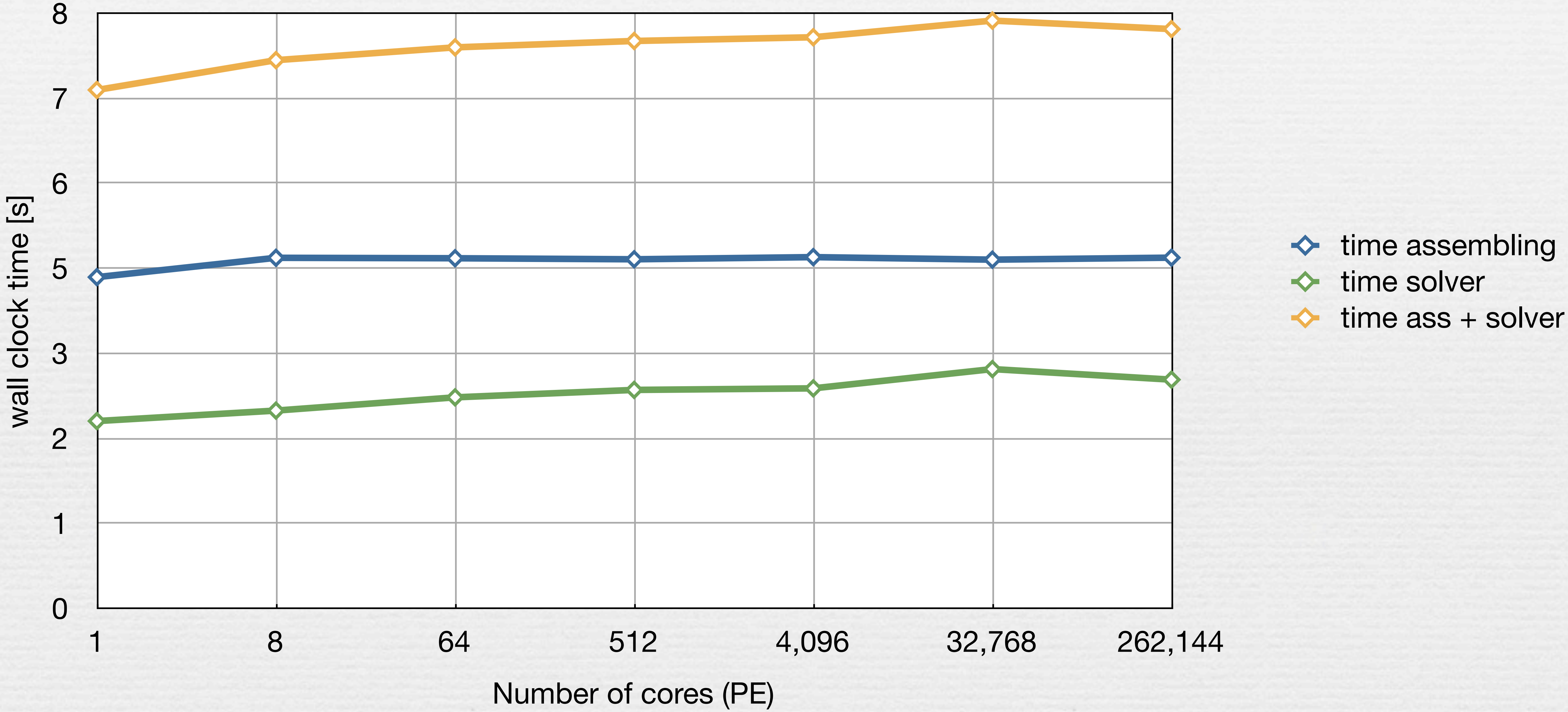
#PE	level	#DoF	T_{total} (s)	$T_{assemble}$ (s)	T_{solver} (s)	$T_{ass+solve}$ (s)	Efficiency (%) (ass+solve)	Speedup (ass+solve)	Speedup ideal	Efficiency (%) (solve)	Speedup (solve)
1	4	35,937	14.568	4.451	2.516	6.967	--	--	--	--	--
8	5	274,625	16.311	4.711	2.657	7.368	94.6	7.6	8	94.7	7.6
64	6	2,146,689	19.584	4.705	2.835	7.540	92.4	59.1	64	88.8	56.8
512	7	16,974,593	18.810	4.691	2.935	7.626	91.4	467.7	512	85.7	438.8
4,096	8	135,005,697	23.129	4.720	2.956	7.676	90.8	3,717.6	4,096	85.1	3,485.9
32,768	9	1,076,890,625	24.980	4.686	3.215	7.900	88.2	28,897.9	32,768	78.3	25,647.7
262,144	10	8,602,523,649	52.422	4.713	3.073	7.786	89.5	234,575.1	262,144	81.9	214,661.2

Scaling study for the weak scaling of the laplace problem on a unit cube $[0,1]^3$. Initial grid with 2x2x2 hexahedrons and uniform refinement for each grid level. Solving of the linear equation system is done using a geometric multigrid solver with Jacobi smoother until a defect norm of 10^{-12} is reached. No problem specific optimizations have been made. (Abbreviations are: PE = Processing entities (cores), DoF = Degrees of Freedom; T_{total} = total run time, $T_{assemble}$ = time for assembling of system matrix and coarse grid matrices, T_{solver} = time for solver).

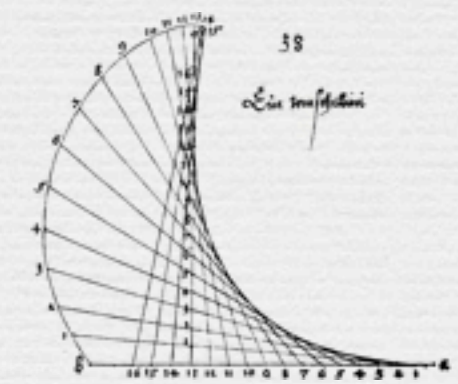
Andreas Vogel
G-CSC
University of Frankfurt



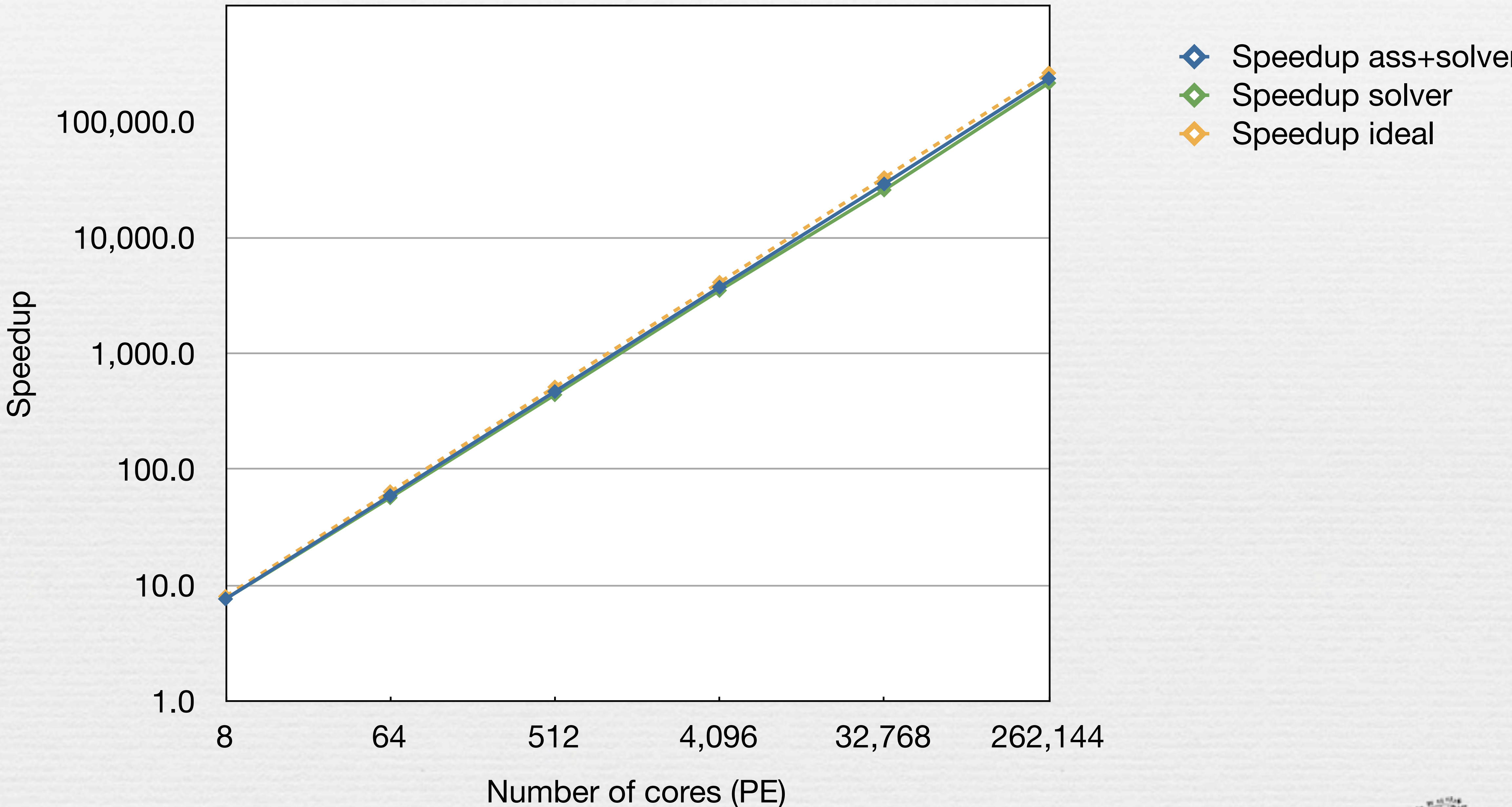
Weak scaling 3d

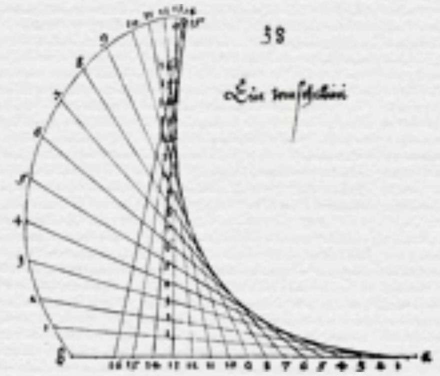


Andreas Vogel
G-CSC
University of Frankfurt



Weak scaling 3d





Andreas Vogel
G-CSC
University of Frankfurt

Observations

- Number of target processes during vertical distribution is crucial
- At most 512 target processes allows to hide the „one to many“ communication during prolongation / restriction
- In run for 256Ki cores 3 levels of hierarchical distribution necessary

- Usage of shared libraries are possible on Jugene (Jülich, Germany). But for very large number of processes the start up time increases dramatically.
- Using statically build binaries is recommendable

Andreas Vogel
G-CSC
University of Frankfurt

Density Driven Flow

Transport of saltwater in porous media

[cf. Bear `91, Leijnse `92, Holzbecher `98, ...]

<u>Governing equations</u>: Two nonlinear, coupled PDE

$$\partial_t(\phi\rho) + \nabla \cdot (\rho\mathbf{q}) = 0$$

$$\partial_t(\phi\rho\omega) + \nabla \cdot (\rho\omega\mathbf{q} - \rho D(\mathbf{q}) \cdot \nabla\omega) = 0$$

$$\mathbf{q} = -\frac{\mathbf{K}}{\mu} \cdot (\nabla p - \rho\mathbf{g})$$

<u>Unknowns</u>:

ω = Mass fraction of brine

p = Pressure

<u>Material laws</u>:

$\rho(\omega)$ = Density

$\mu(\omega)$ = Viscosity

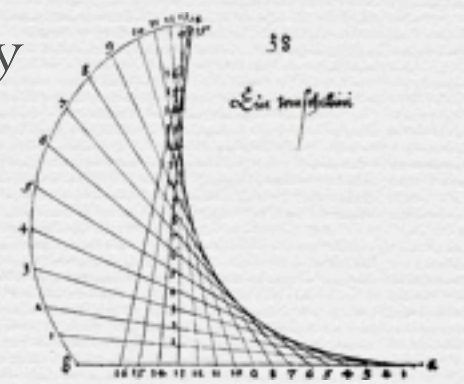
<u>Parameters</u>:

ϕ = Porosity

$\mathbf{K}$ = Permeability

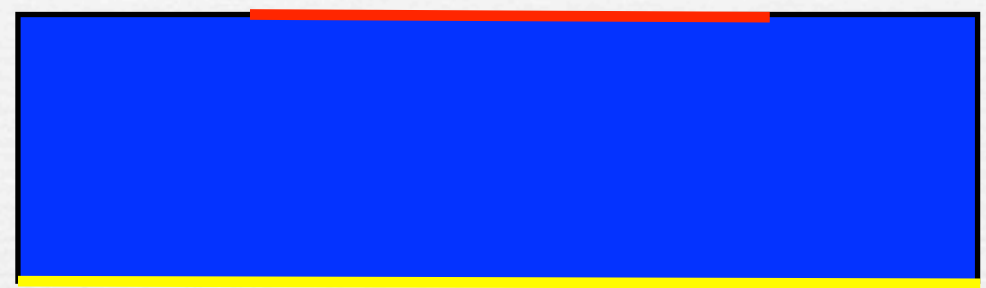
$\mathbf{g}$ = Gravity

Andreas Vogel
G-CSC
University of Frankfurt



Elder Problem

- Boundary Conditions:



- Model: Boussinesq - Approximation
- Discretization: vertex-centered finite volume
- Upwinding: exponential
- Time-stepping: fully implicit

- Newton-Method with assembled Jacobian
- Solver: BiCGStab with GMG Preconditioner with ILU Smoother

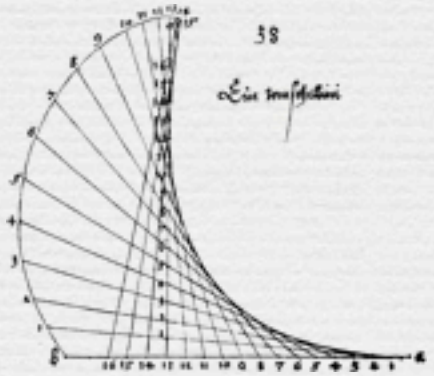
Andreas Vogel
G-CSC
University of Frankfurt

Density Driven Flow

#PE	Level	#TimeSteps	# DoF	$T_{assemble}$ [s]	$T_{linSolver}$ (Total) [s]	Avg Lin Iters	T_{gmg} (each) [s]
64	9	20	8398850	497.28	571.81	7.24	3.95
256	10	20	33574914	476.00	593.36	7.90	3.76
1024	11	20	134258690	474.44	991.08	13.40	3.70
4096	12	20	536952834	475.97	2787.01	37.25	3.74
16384	13	20	2147647490	476.85	1033.36	13.68	3.78

Scaling study for the weak scaling of the elder problem in 2d. Initial grid with 8x2 quadrilaterals and uniform refinement for each grid level. Solution of the non-linear problem using Newton-iteration. Linearized problems are solved using a BiCGStab solver with geometric multigrid preconditioner and ilu smoother.
(Abbreviations are: PE = Processing entities (cores), DoF = Degrees of Freedom; $T_{assemble}$ = time for assembling of system matrix and coarse grid matrices, $T_{linSolver}$ = time for linear solver within newton iteration)

Andreas Vogel
G-CSC
University of Frankfurt



Summary

- GMG with hierarchical splitting of processors and gathering during coarsening is suited for large scale computations.
- Tested for 262144 processes.
- Nice scaling behavior if *one-to-many* communication for more than 10^3 target processes is avoided.

Thank you for your attention.

Andreas Vogel
G-CSC
University of Frankfurt