

# Parallel Flow

Markus Blatt

HPC-Simulation-Software & Services

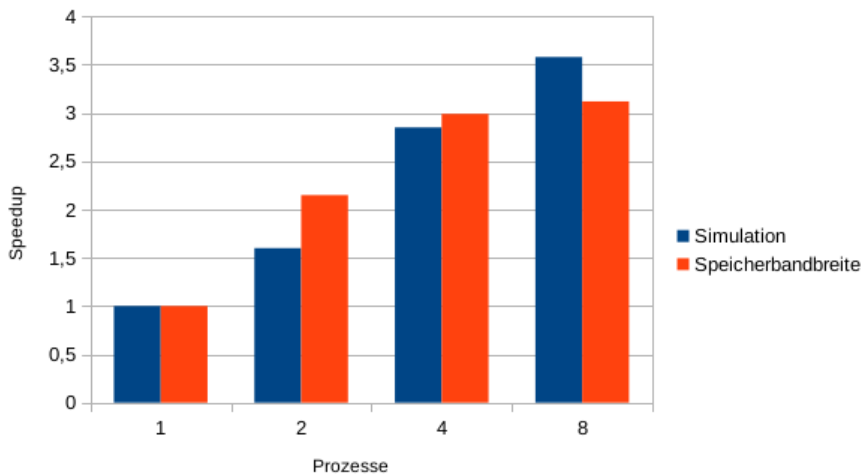
OPM Meeting  
Bergen, NO  
Oct 18-19, 2017

# Scalability Norne before 06/2016

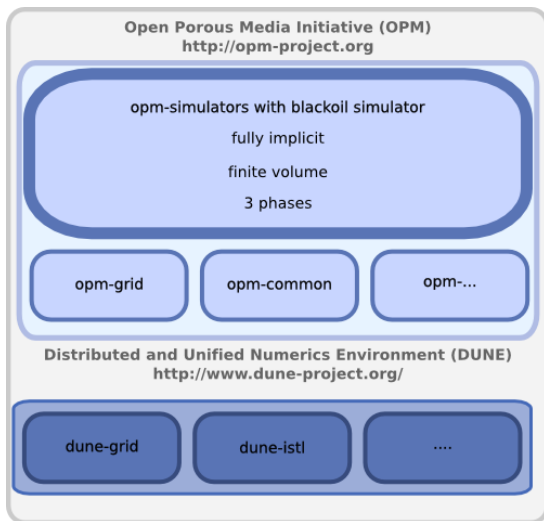
1 Intel(R) Xeon(R) CPU E5-2620 v3 @2.40GHz (6 cores/12 threads)

procs	t_total (s)	n_lin	n_nonlin	speedup
1	2104.59	24572	1626	1.00
2	1516.28	29096	1710	1.39
4	977.21	27208	1673	2.15
8	869.97	32145	1704	2.42

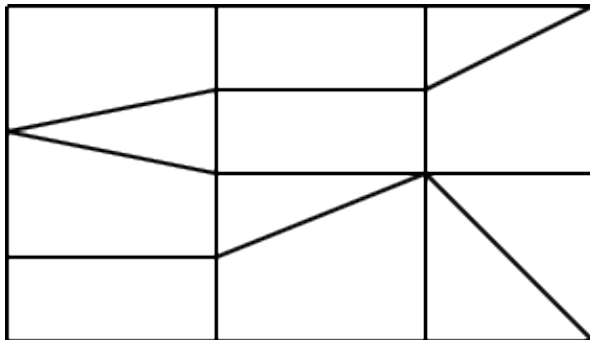
## 1 Intel(R) Xeon(R) CPU E5-2620 v3 @2.40GHz (6 cores/12 threads)



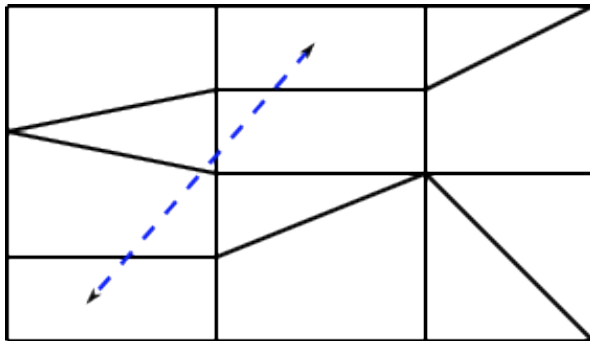
# Flow in OPM



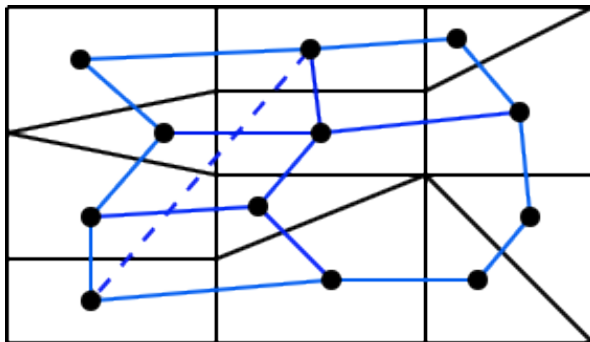
# Finite volumes on cornerpoint grids



# Wells influence distant cells

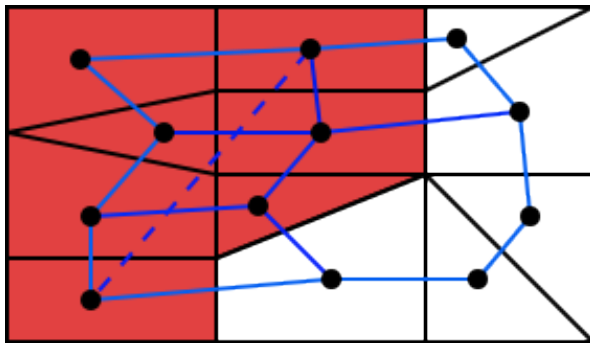


# Graph of direct influences



- Transmissibilities between cells represent weights
- Max transmissibilities along a borehole
- Load balancing with PT-Scotch/Zoltan

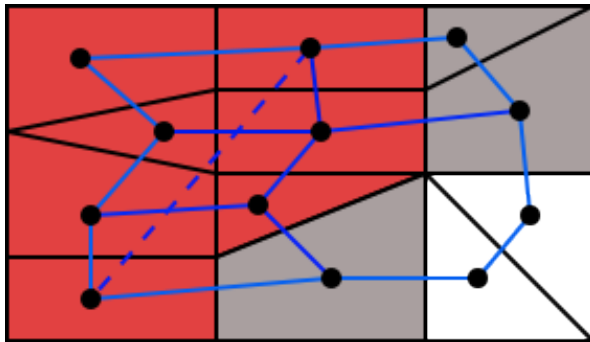
# Partitioning



- Complete well in one partition
- Process computes correct data in red region ...



## Halo

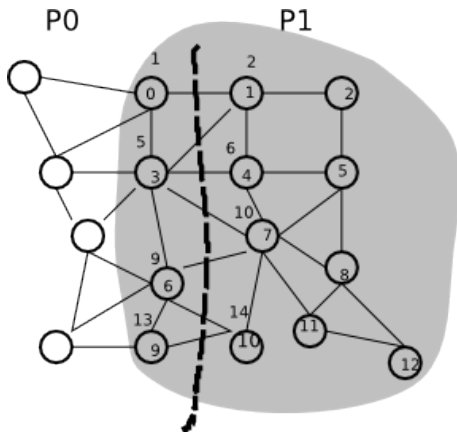


... using a halo region.

# Iterative Solver Template Library

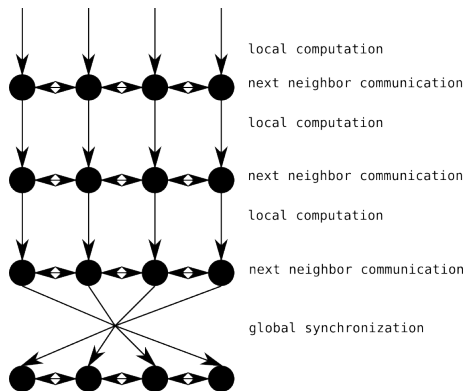
- MPI based parallelization (except for the Exa-Scale version)
- reuse efficient sequential solver components
- abstract simple parallelization approach based on index sets.
- efficient aggregation algebraic multigrid

## Index set with global and local numbering



## Semi-agnostic parallelization . . .

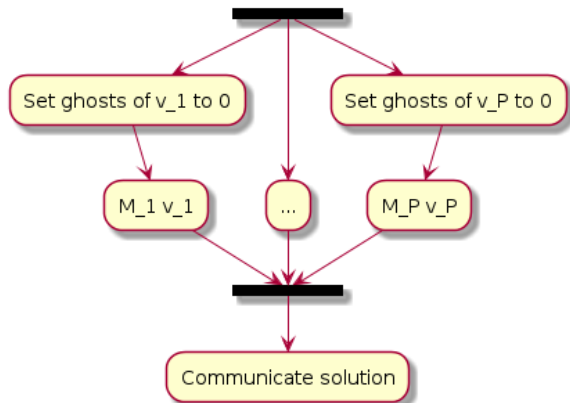
. . . allows for reusing major parts  
of a sequential program/solver



# Go parallel by choosing parallel components

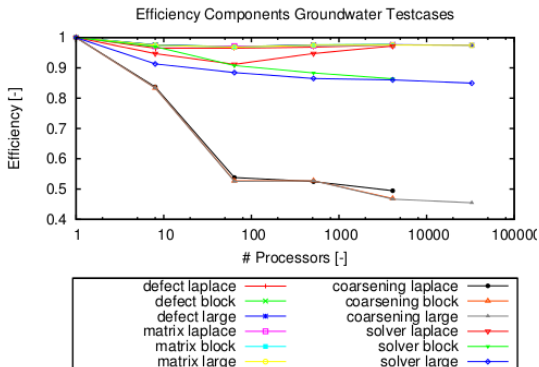
- parallel linear operator
- parallel preconditioner
- parallel scalar product

# Hybrid Smoothers



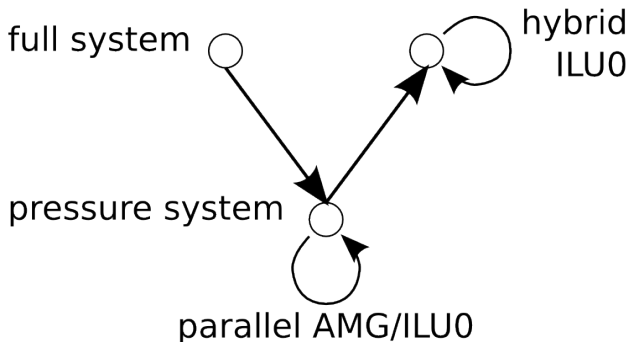
- Common choice for AMG methods

I guess we are set, right?



Timings done by Olaf Ippisch (TU Clausthal) on Jugene.  
Well, at least for simpler model problems. . .

## CPR Preconditioner (flow\_legacy)



- Preconditioner honors well contributions
- Sequential version by Atgeirr Rasmussen (SINTEF)



# Do not neglect the overlap for the smoother (ILU)

- Discretize/decomposition on local domain including halo region.
- forward/backward solve with updated halo regions.
- More communication but better stability

## Sad but true

For our problems of interest (Norne, etc.) CPR with AMG does not beat (CPR)-ILU.

# The new kid on the block (flow\_ebos)

- (Nearly) complete rewrite of discretization routines using Dumux/Ewoms
- Linear systems passed to preconditioner lack well contributions
- Linear operator still incorporates well contributions
- Much faster (Eat that Eclipse!)
- Work performed by Klöfkorn, Sandve (IRIS), and Lauser

# flow-ebos scalability Norne

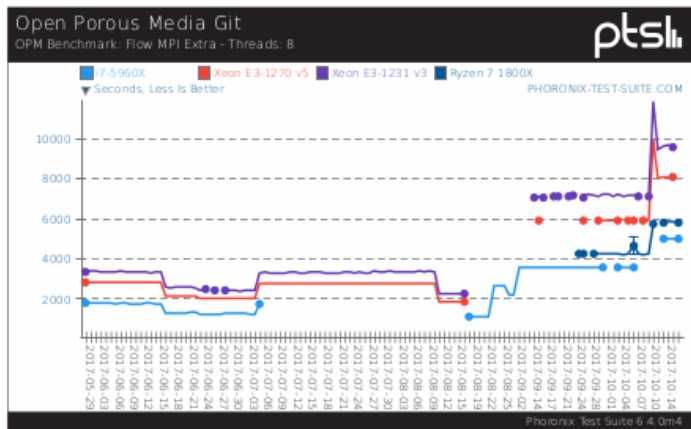
procs	t_total (s)	t_ass (s)	t_lin (s)	n_lin	n_nonlin
1	870.86	408.794	408.79	23988	1574
2	497.52	198.37	257.90	26553	1624
4	455.16	194.61	228.34	31379	2622
8	229.81	74.08	134.64	25272	1552

Speedup 3.8 on 8 ranks

## flow\_ebos scalability Model 2

procs	t_total (s)	t_ass (s)	t_lin (s)	n_lin	n_nonlin
1	3651.33	1434.23	2083.86	47439	1974
2	1746.28	555.691	1058.56	48736	1891
4	1792.04	372.88	1319.89	88569	2095
8	2820.01	593.35	2087.45	120573	5139

# Scalability fluctuations



## Comparing legacy and ebos version

procs	ebos 2017(s)	legacy 2016? (s)
1	870.86	2104.59
2	497.52	1516.28
4	455.16	977.21
8	229.81	869.97

2016 speedup was only 2.4! Now 3.8 !

# Outlook / Future Work

- Bugfix / test CPR with flow\_ebos (ongoing)
- Revisit parallel ILU with scheduling
- Test effect of well connections to preconditioner
- Other load balancing strategies?



# Contact

Markus Blatt

Dr. Blatt - HPC-Simulation-Software & Services

Pedettstraße 38

85072 Eichstätt

markus@dr-blatt.de <http://dr-blatt.de>

Special thanks to my collaborators from IRIS, SINTEF, and Statoil.