Engineering Algorithms for Large Data Sets

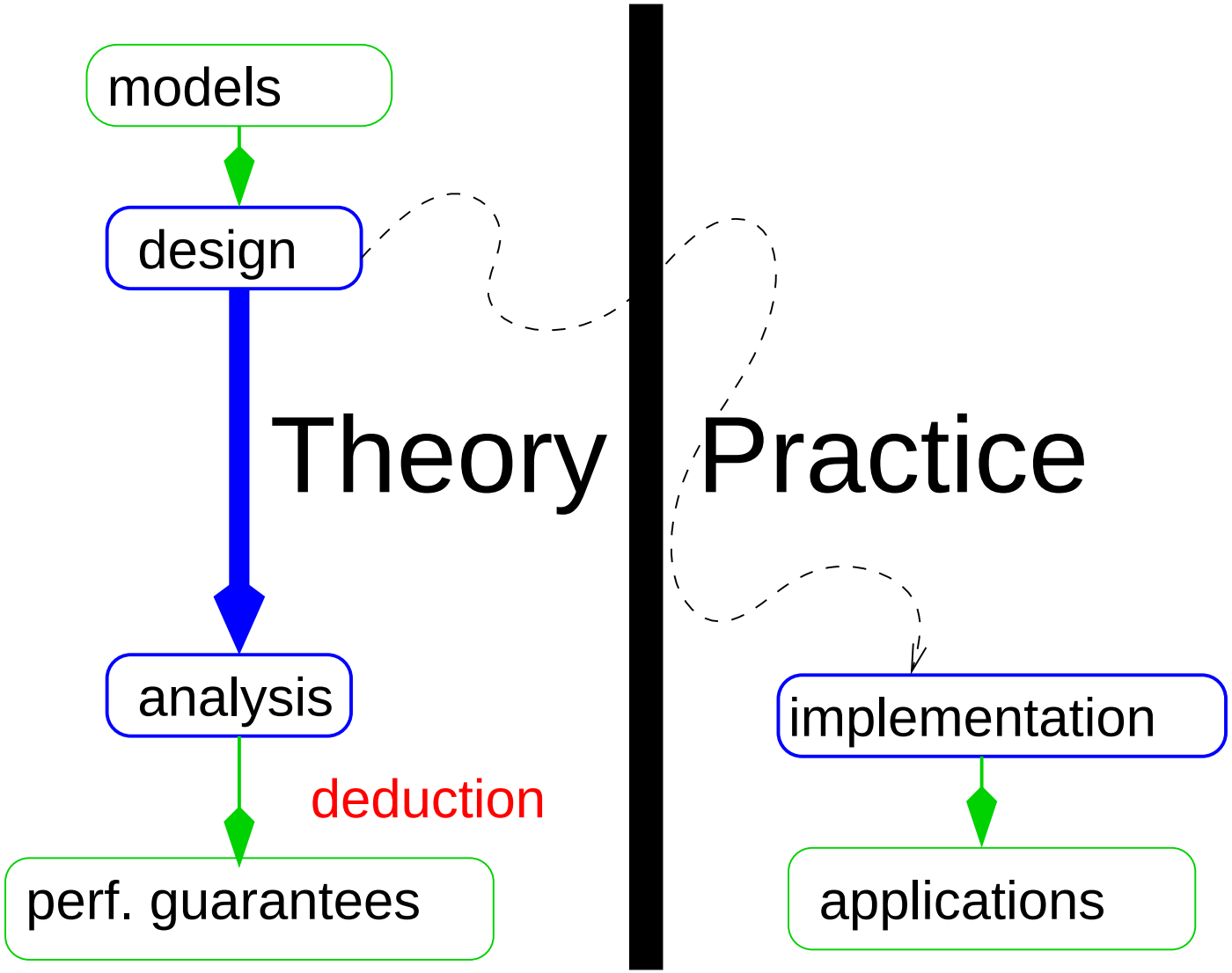
Peter Sanders

Karlsruhe Institute of Technology

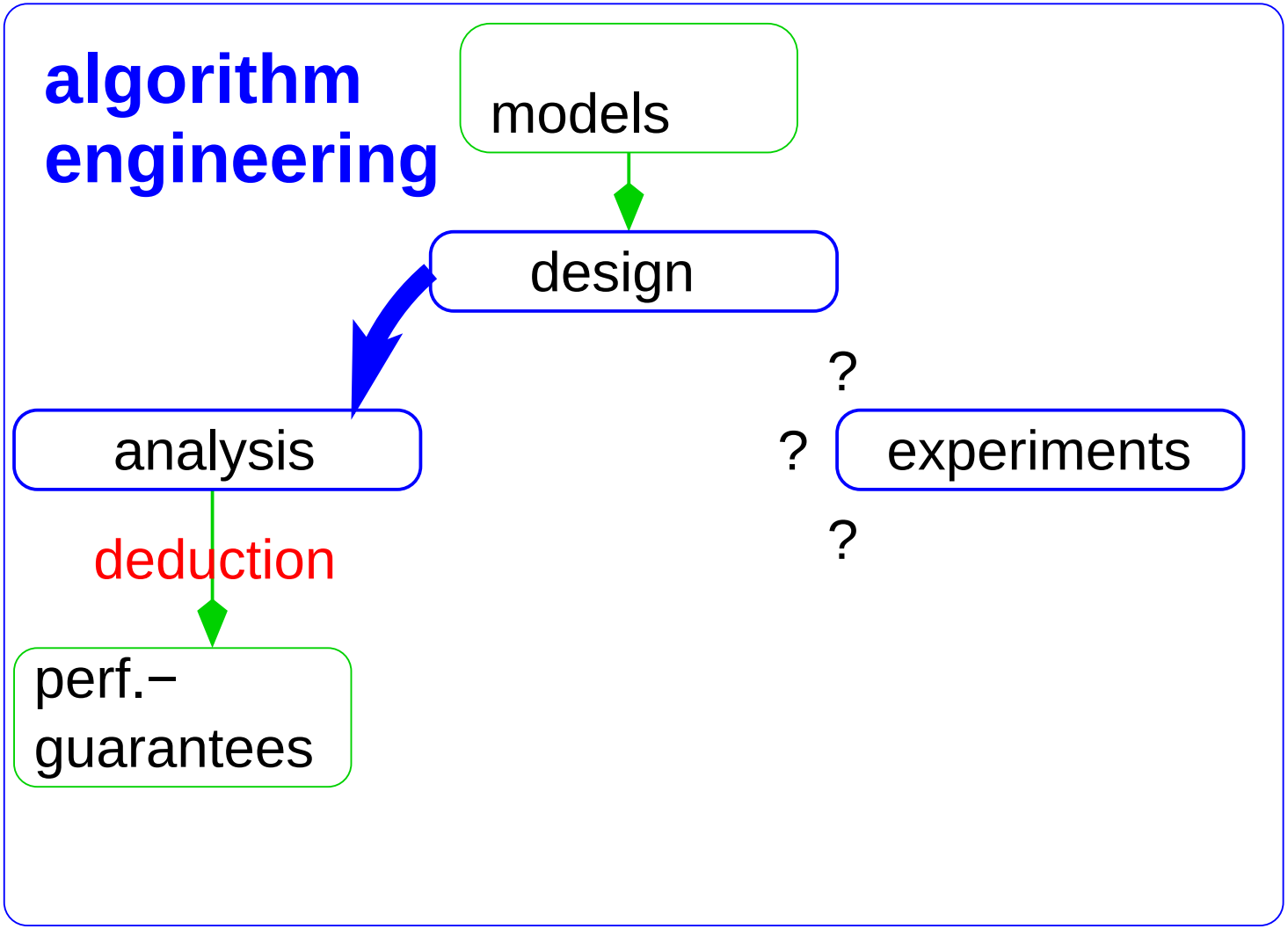
Overview

- Algorithm Engineering
- Algorithmic challenges from large data
- Examples from my work

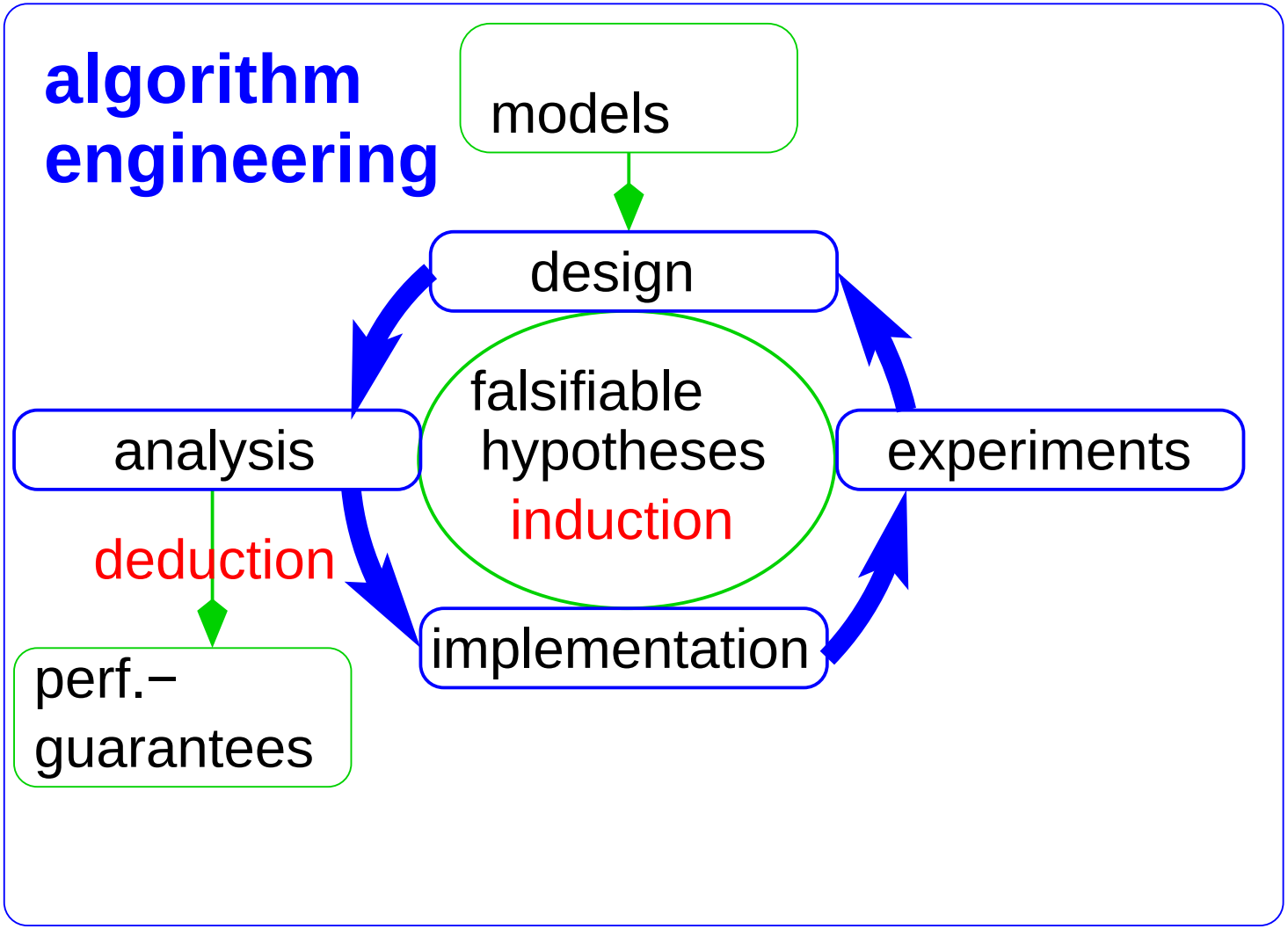
(Caricatured) Traditional View: Algorithm Theory



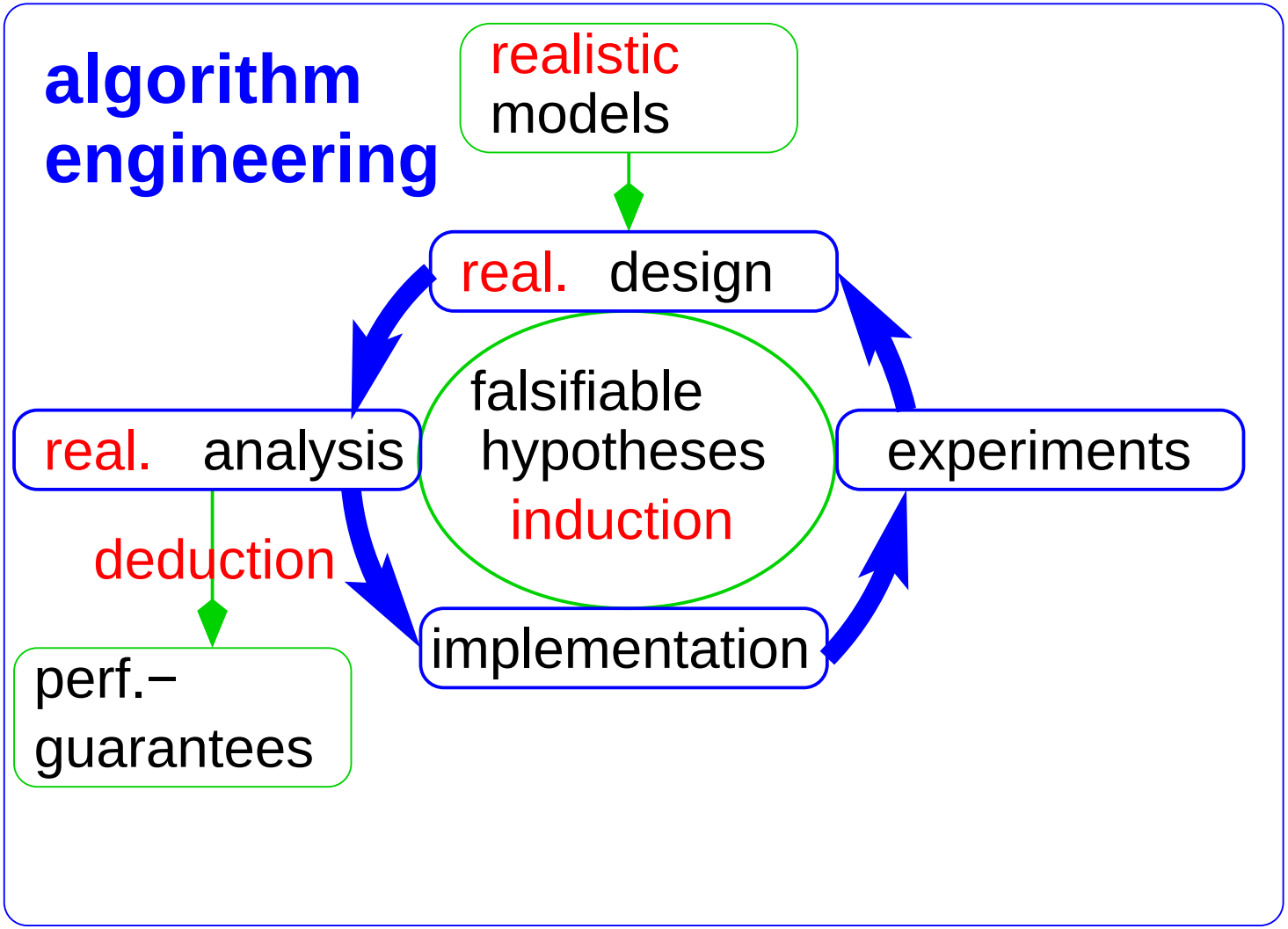
Algorithmics as Algorithm Engineering



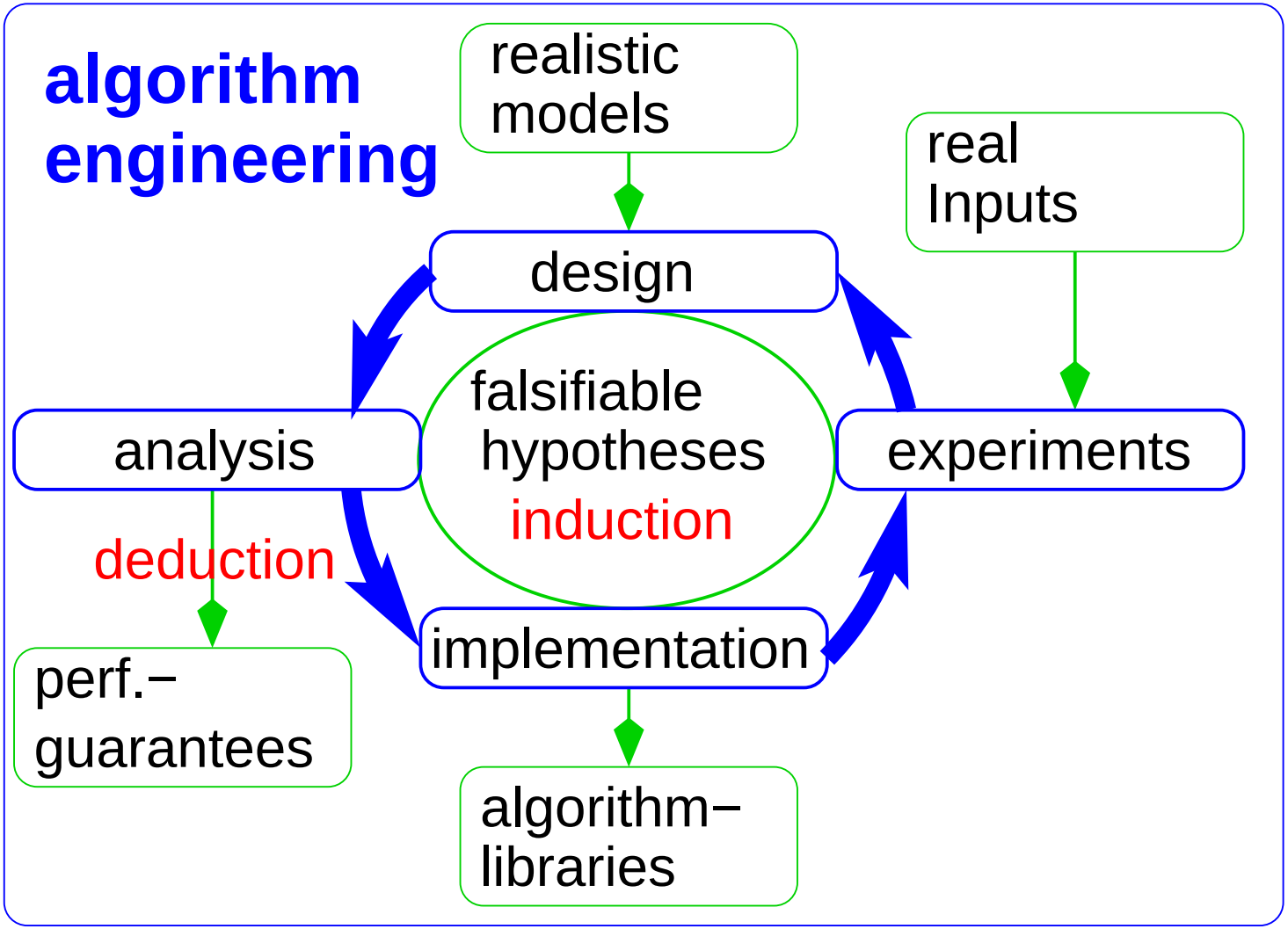
Algorithmics as Algorithm Engineering



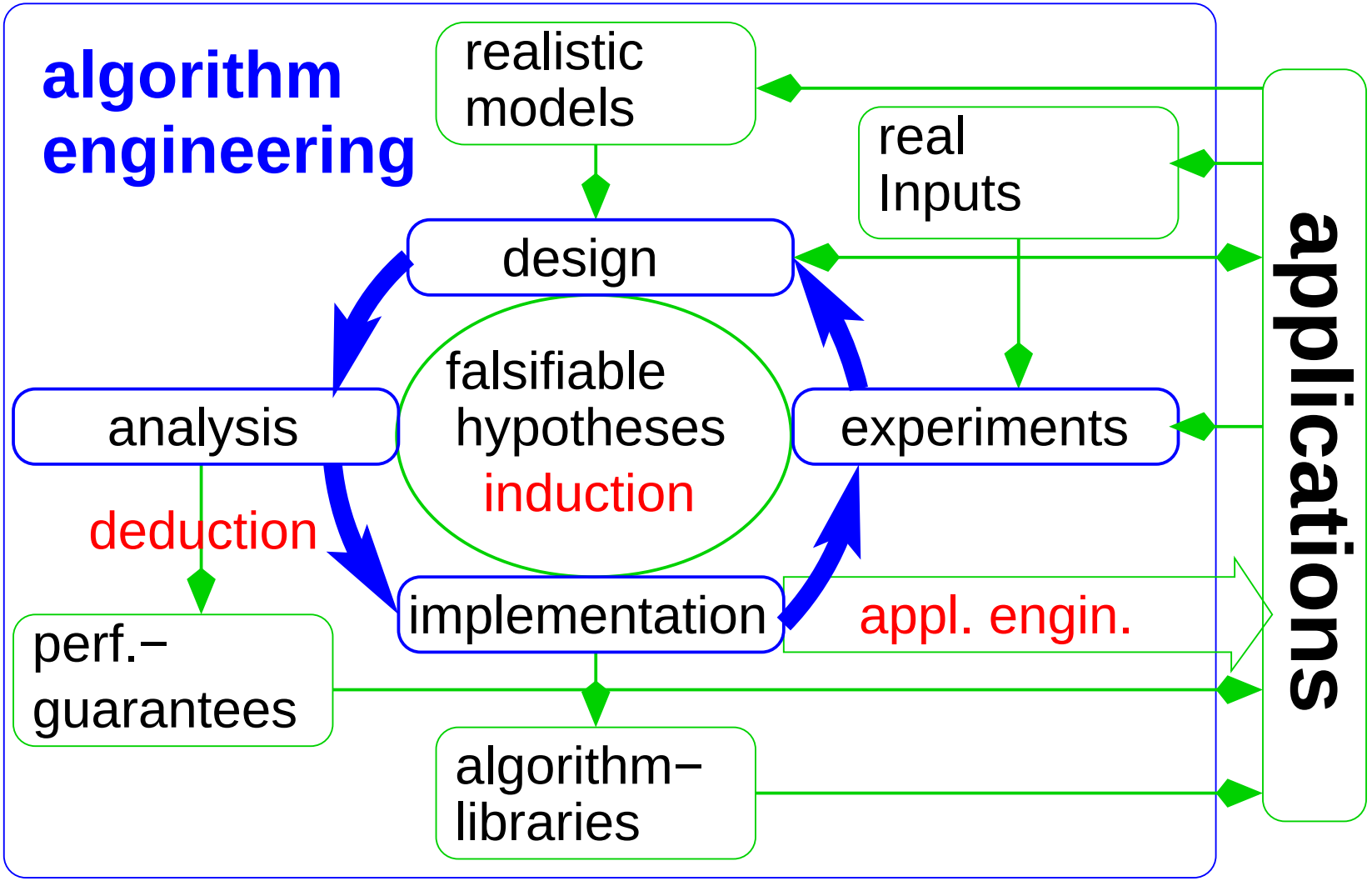
Algorithmics as Algorithm Engineering



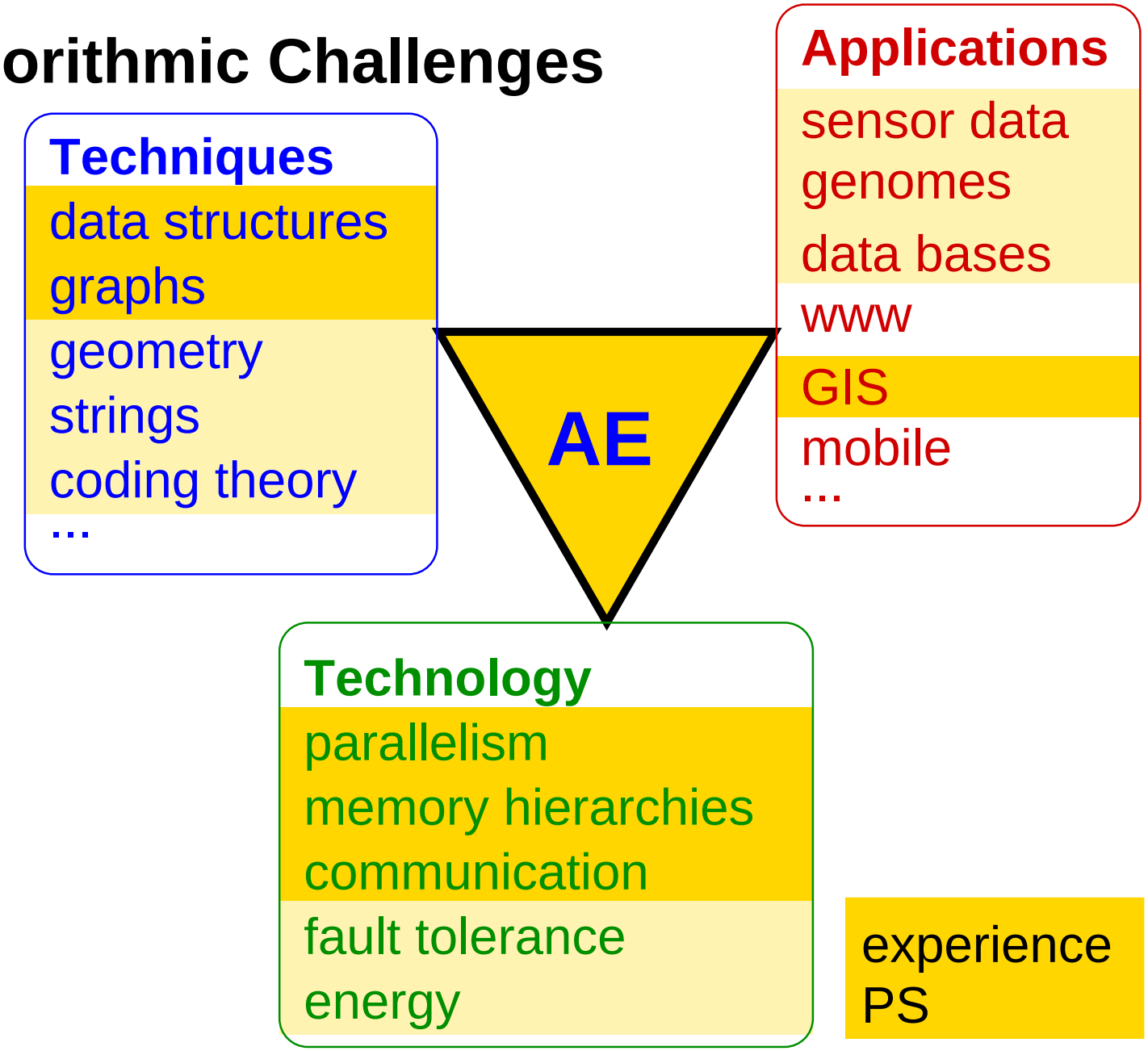
Algorithmics as Algorithm Engineering



Algorithmics as Algorithm Engineering



Algorithmic Challenges



Examples from my Group

- ≈General purpose tools
  - sorting
  - full text indices
  - data bases
  - storage servers
  - graph partitioning and clustering
- Applications
  - CERN-LHC track reconstruction for CMS
  - route planning
  - genome sequencing
  - image processing

GraySort Data Base Benchmark

[RahnSSinger ICDE 2010]

sort 100 TB of 100 byte records

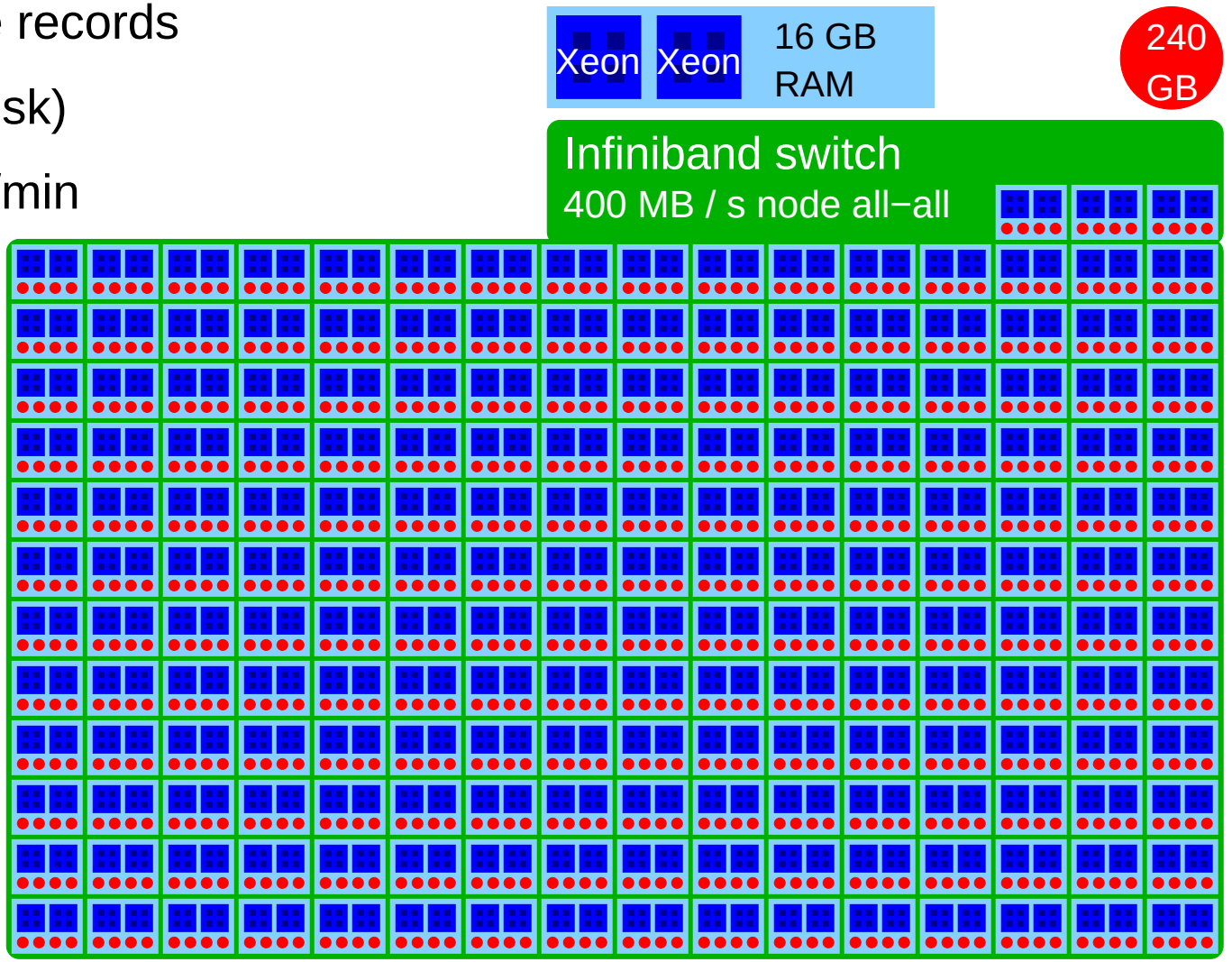
with 10 byte keys (on disk)

2009 winner: 0.564 TB/min

Techniques

- randomization
- load balancing
- disk scheduling
- collective comm.
- multiway merging

IC1 Cluster



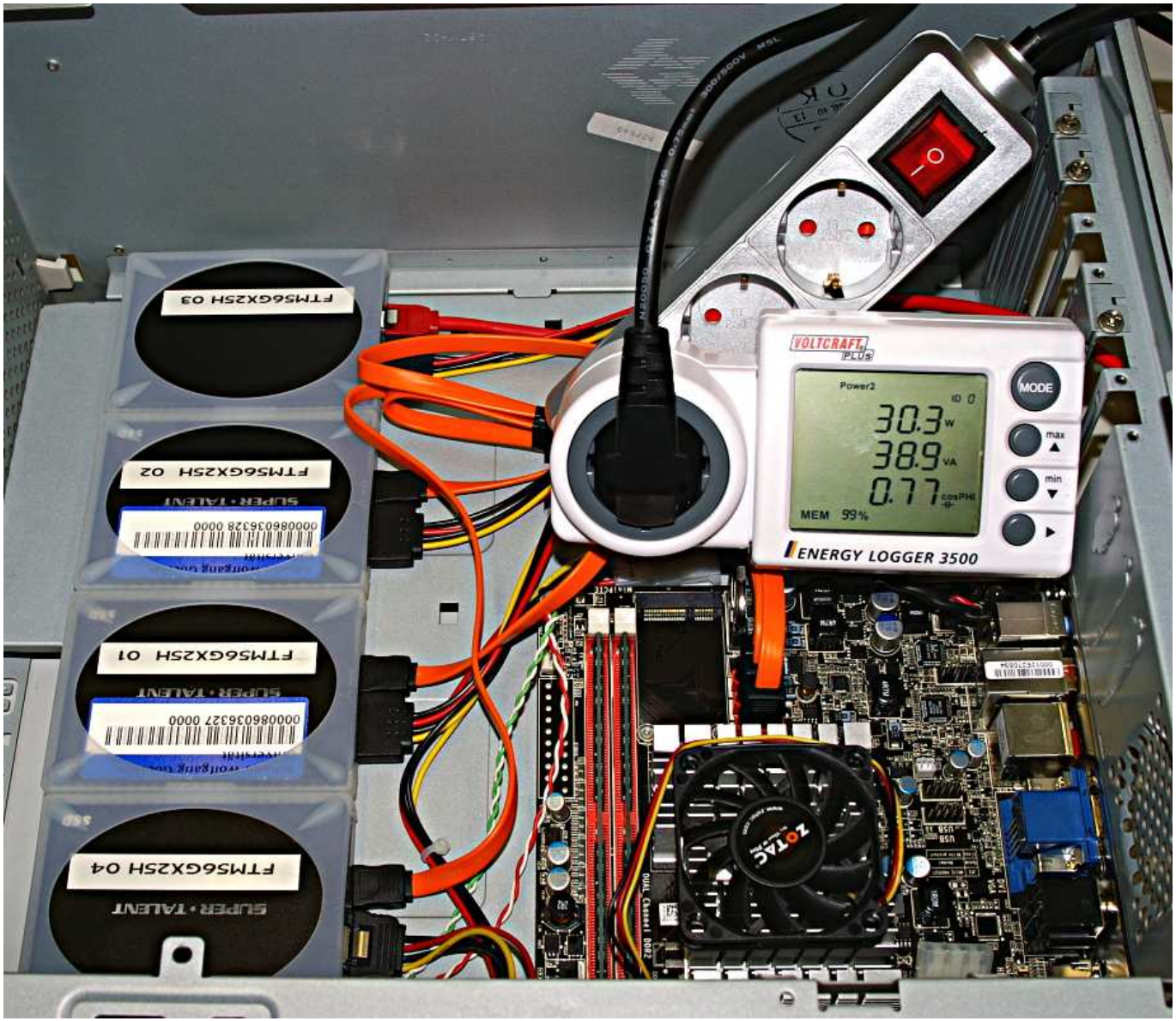
JouleSort 2010

[BeckmannMeyerSSingler 2010]

- Intel Atom N330
- 4 GB RAM
- 4×256 GB SSD (SuperTalent)

Algorithm similar to GraySort

Deutschland Land der Ideen



Call for cooperation: who has a larger cluster with local disks?

KIT IC2: 400×2 disks

Larger Sorting Problems

- millions of processors

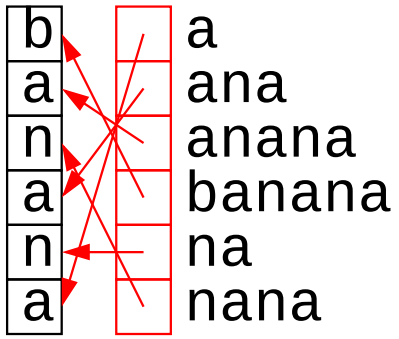
  $\rightsquigarrow$ multipass algorithms

- fault tolerance

- still energy $\sim$ time?

Higly related to MapReduce, index construction,...

Suffix Sorting



sort suffixes $s_i \cdots s_n$ of string

$S = s_1 \cdots s_n, s_i \in \{1..n\}$.

Applications: full text search,

Burrows-Wheeler text compression, bioinformatics,...

E.g. phrase search in time logarithmic

or even independent of input size.

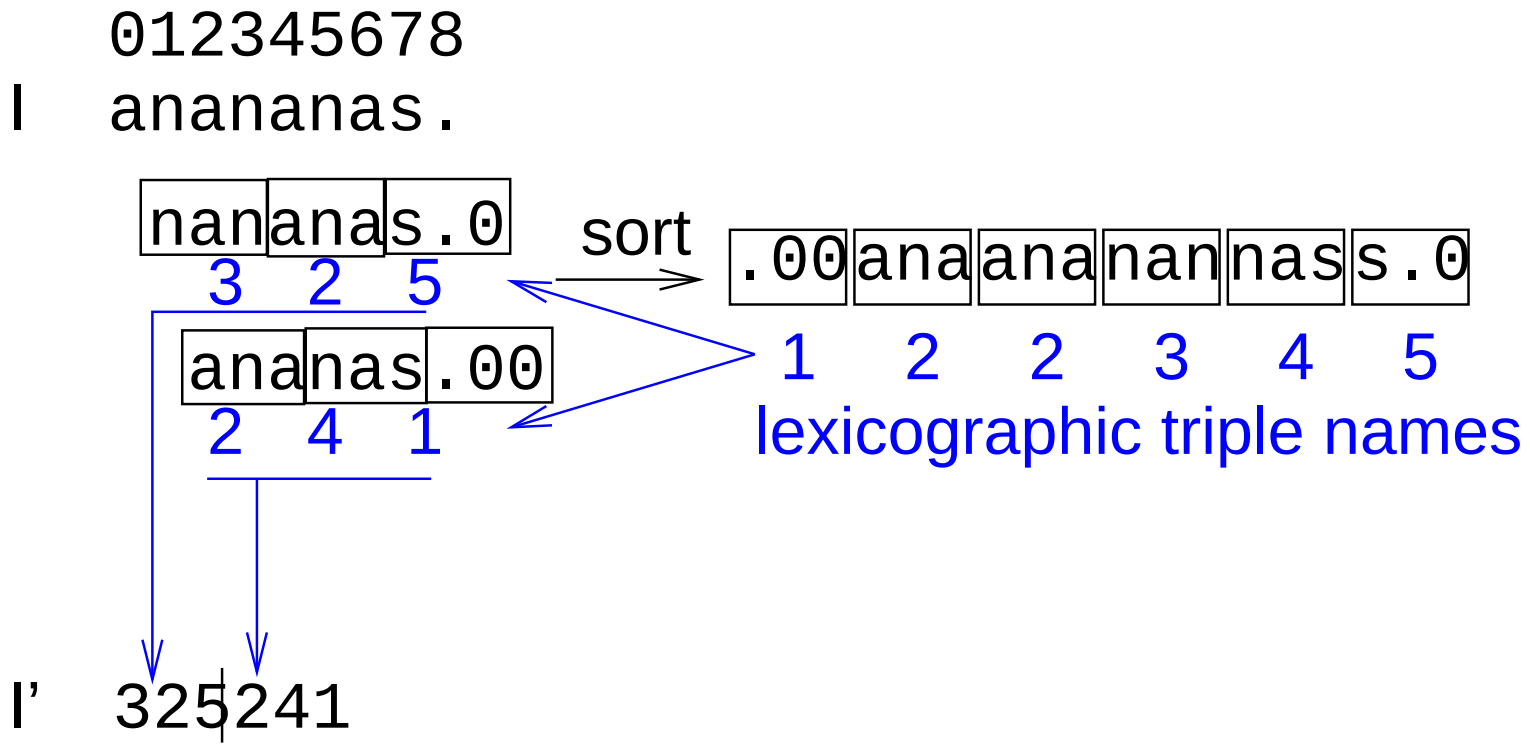
$\rightsquigarrow$ particularly interesting for large data

Linear Work Suffix Sorting

[Kärkäinen-Sanders-Burkhardt JACM 2006]

simple: Radix-Sort + linear recursion + merging.

$\rightsquigarrow$ trivial external, parallel adaptation



Current Work

- distributed memory (external) query
- parallel distributed construction of query data structure (longest common prefixes,. . .)

Towards an Energy Efficient Search Engine

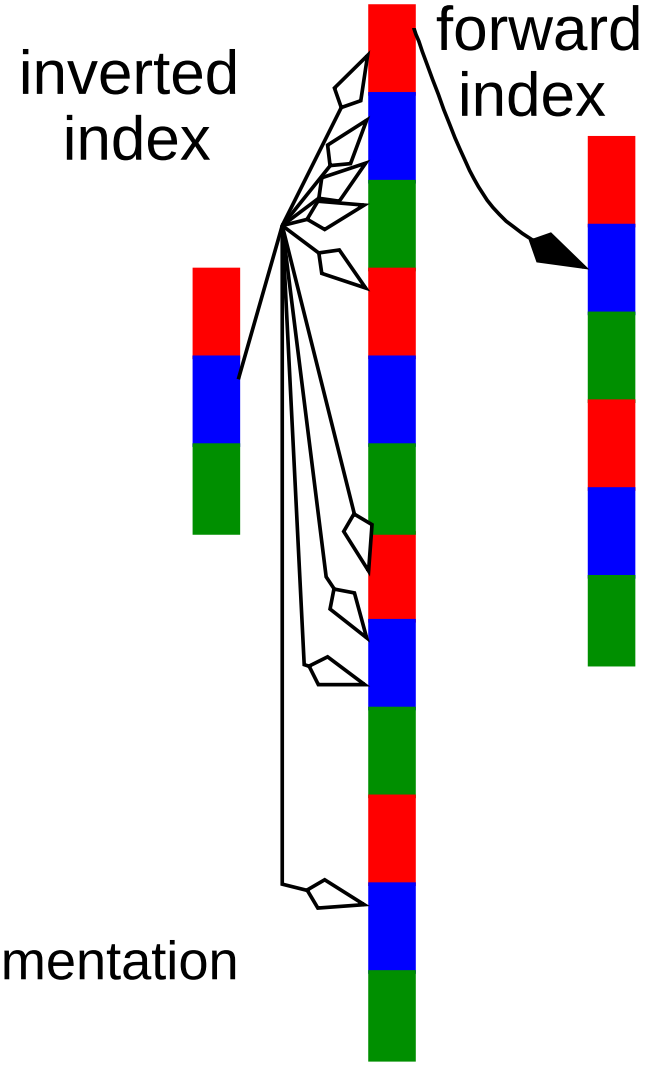
- conjunctive queries
- touch only top ranked results
- fault tolerance

Algorithmic meat: Clustering, data structures, compression, fault tolerance, load balancing

Data Bases – Our Approach

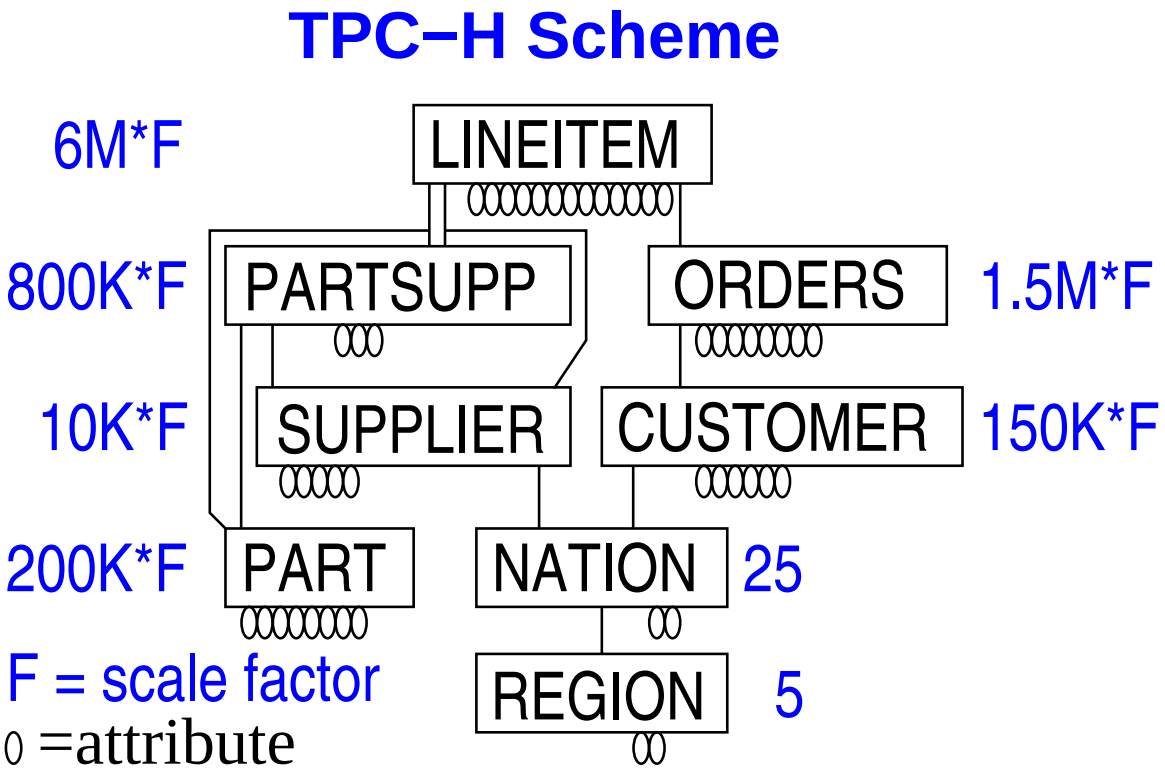
[with SAP HANA team, PhD students Dees, Müller]

- main memory based
- column based
- many-core machines
- NUMA-aware
- no precomputed aggregates
- aggressive indexing
- generate C++ code close to tuned manual implementation



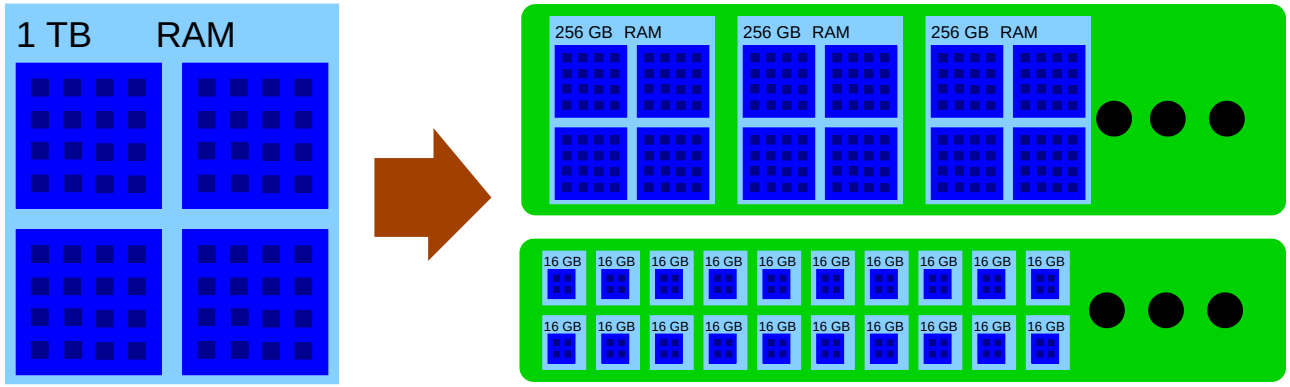
TPC-H Decision Support Benchmark

- $\approx 30\times$ faster than current record in 300GB category (manual implementation)
- compiler: work in progress



Larger Inputs

- Already needed by some large customers of SAP
- Move to clusters
- fault tolerance beyond recovery?
- energy efficiency using many small nodes (ARM)?

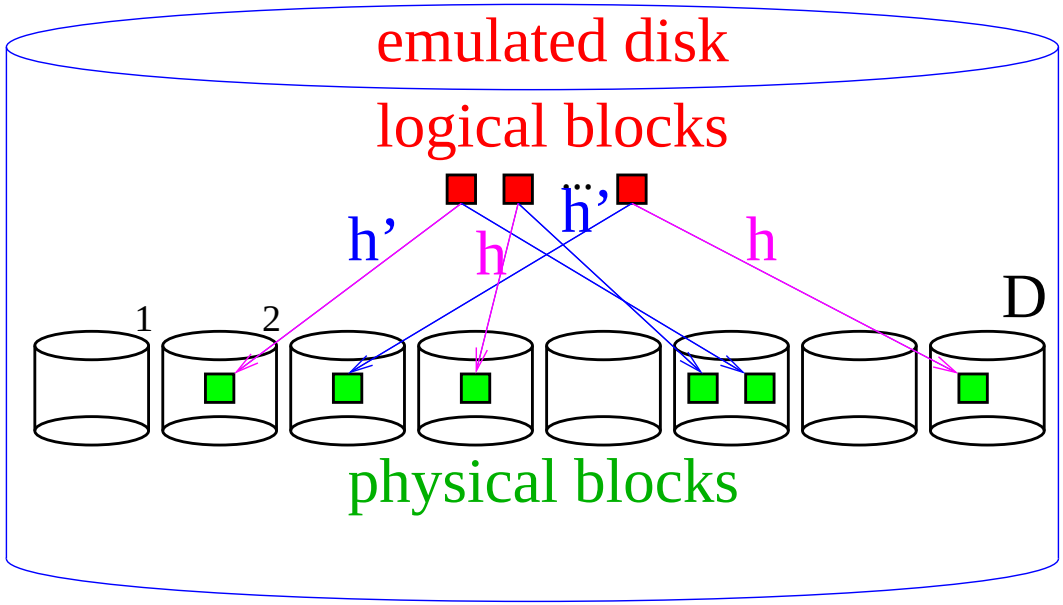


Algorithmic Meat: Randomization, collective communication, communication complexity, sorting, data structures, multi-level memory hierarchies, coding theory

Storage Servers:

Random Redundant Allocation

- Idea: flexibility by redundant storage
- E.g. two copies for each logical block
- Two hash functions h und h'

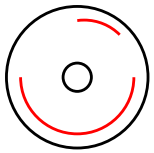


planning problem: read N blocks from D disks.

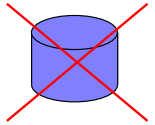
Optimal schedule: $\left\lceil \frac{N}{D} \right\rceil + 1$ steps.

Generalizations

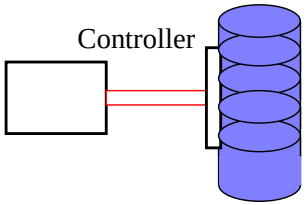
- Variable block lengths



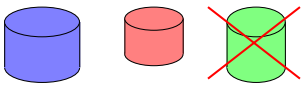
- Disk failures



- Communication bottlenecks



- Asynchronous parallel accesses from several applications
- Inhomogeneous changing pool of disks



Graph Partitionierung

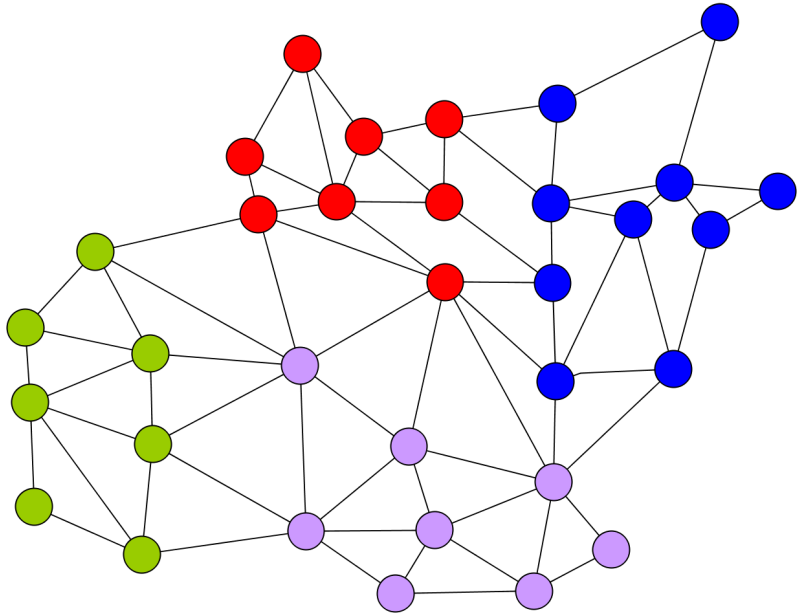
Input: Graph (V, E) (possibly with node and edge weights), ϵ, k

Output: $V_1 \dot\cup \cdots \dot\cup V_k$ mit $|V_i| \leq (1+\epsilon)\left\lceil \frac{|V|}{k} \right\rceil$

Objective Function: minimize cut

Applications: finite element simulations, VLSI-design, route planning,...

Variants: hypergraphs, clustering, different objective functions,...



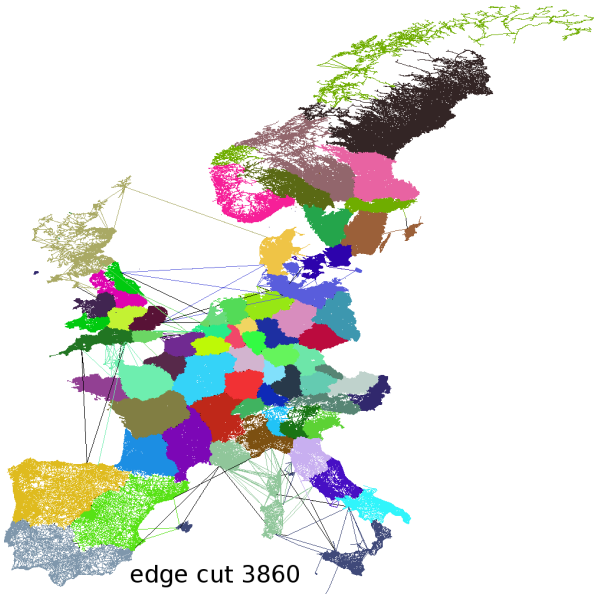
Large Data Graph Partitioning

- difficult inputs: social networks, WWW, 3D/4D models, VLSI
- more difficult parallelization

Our Contribution

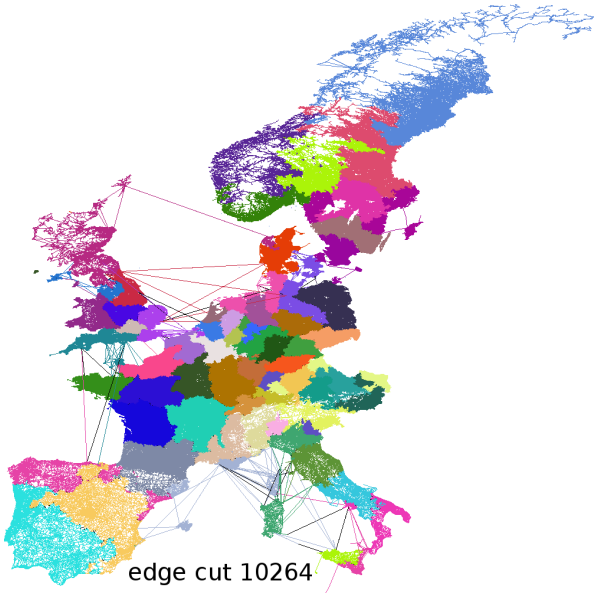
- scalable parallelization KaPPa
  (matching, edge coloring, evolutionary)
- thorough reengineering of multilevel approch
  (use flows, SCCs, augmenting paths, . . .)

⇝ high quality (e.g. 90–99%
entries in Walshaw's benchmark)



Future Work

- parallel external
- other variants
- fault tolerant
- component of a graph processing framework



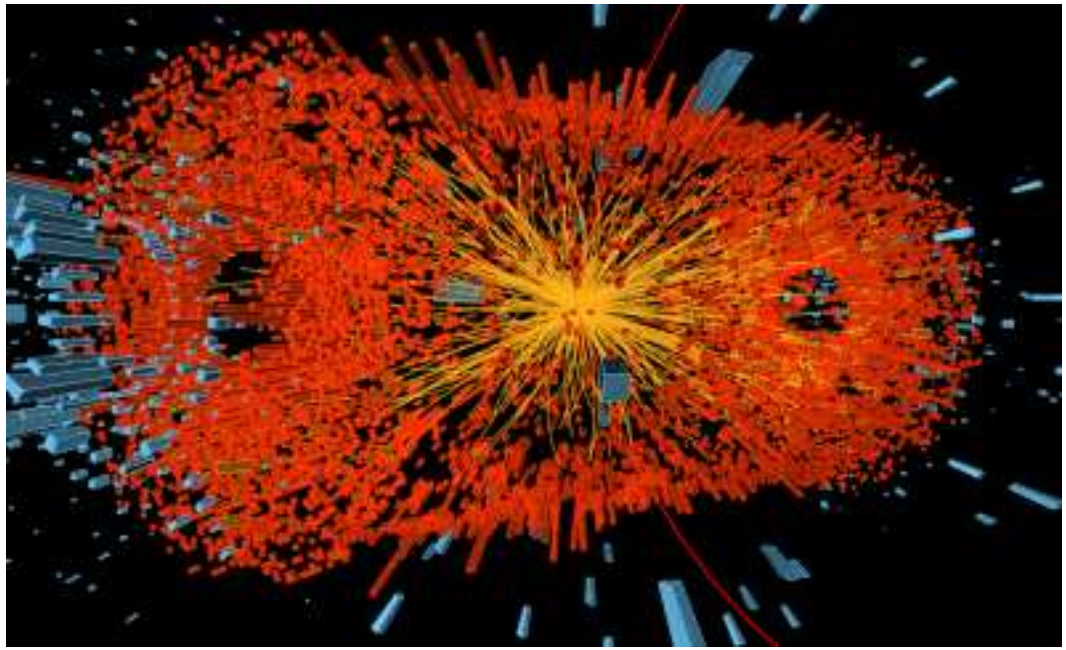
Track reconstruction

[beginning cooperation with Günter Quast and CERN]

Input: clouds of $\approx 10^4$ 3D points

Output: $< 10^3$ spiral tracks of high energy particles

Also cluster tracks by emergence point



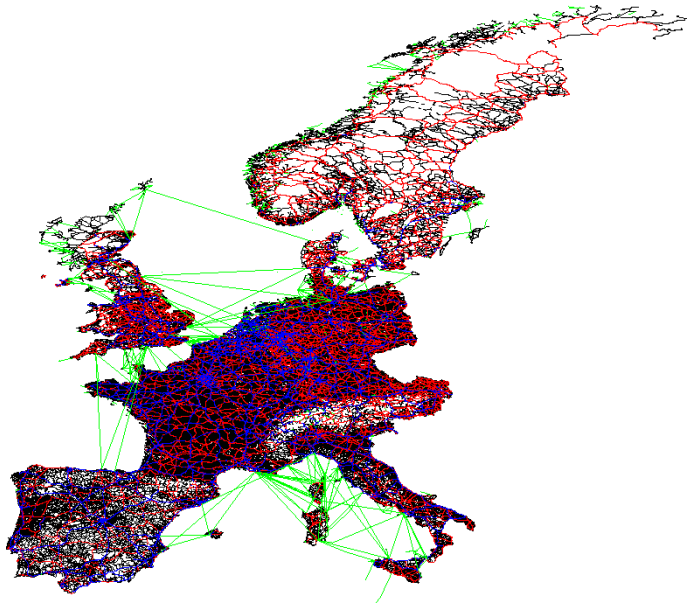
Large Data???

- up to 10^5 instances / s
- cost of processors / energy
- memory constrained
- exploit SIMD/GPU parallelism?

Algorithmic Meat:

Geometric data structures, parallelization, clustering

Route Planning



Large Data 2004: Western European network (18M nodes).

Dijkstra's algorithm needs $6s$.

- too much time for servers
- too much memory for mobile devices

$\rightsquigarrow$ inaccurate heuristics with tedious "manual preprocessing"

Our contribution: Automatic preprocessing techniques

- 10^4–10^6 times faster exact query on servers
- still "instantaneous" on mobile devices (external implementation)

Large Data 2012

- 1.6G nodes OpenStreetMap routing graph (edge based)
- billions of GPS traces
  (+ road based sensors + elevation data)
- public transportation

Potential use:

time-dependent edge weights	(OK)
detailed traffic jam detection	Google, TomTom,. . .
multi-modal route planning	first results
probabilistic route planning	attempts
really useful detours around traffic jams	???

use real time traffic simulation??

Genome Sequencing

[Venter et al. Celera 2000]: 20 000 CPU hours for shotgun sequencing of the human genome ($3 \cdot 10^9$ base pairs, 5–10 times oversampling.

Prototypical large data problem?

Today: a few minutes on a work station [ZieglerDFMS SAP 2012]

(use template, modern hardware, AE $+$ cheap sequencing)

$\rightsquigarrow$ routine use for personal medicine

New Challenge:

processing many sequences

Phylogenetic Tree Reconstruction

[planned cooperation with A. Stamatakis (HITS/ITI)]

Image Processing

[PhD Wassenberg] Gigapixel aerial images.

Filters, Segmentation, Change detection

Algorithmic meat: Graph algorithms, parallelization, memory hierarchies, range minimum data structures,. . .

Future Work

- see above
- find more algorithmic application problems in LSDMA, KIT,. . .
- algorithmic cores of application independent libraries and tools
  data structures, MapReduce, graphs, data bases,. . .
- distributed memory external algorithms
- back to massive parallelism including exascale
- fault tolerance