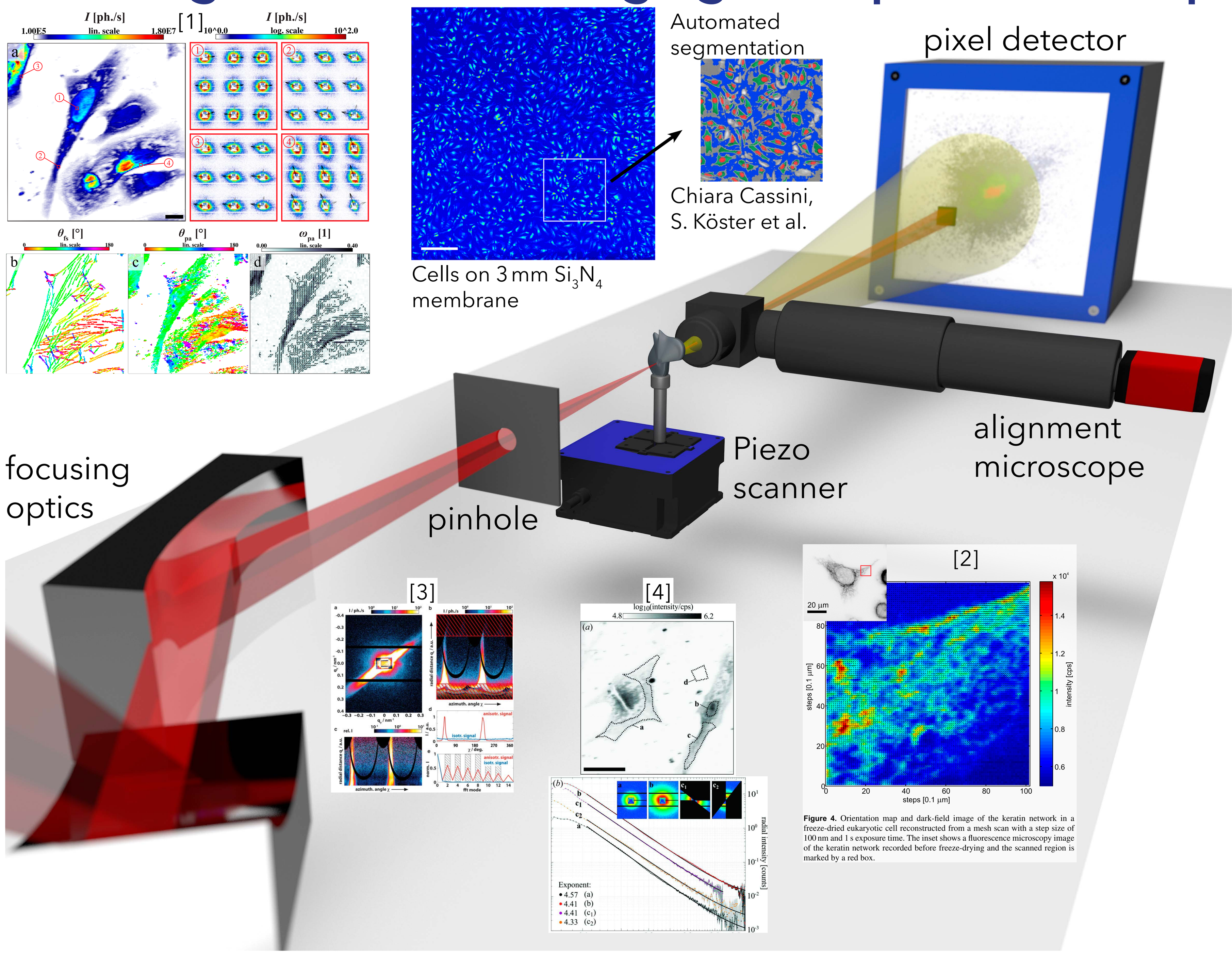
dada-STXM: parallel analysis of Eiger scanning-SAXS data

Markus Osterhoff – Jan Goeman – Sarah Köster – Tim Salditt

Institut für Röntgenphysik – Friedrich-Hund-Platz 1 – 37077 Göttingen

Scanning nano-SAXS: bridging real-space and reciprocal space for biological imaging



- Small Angle X-ray Scattering: quantify structure sizes, morphology, orientations
  common approach: large ensemble averages in solutions
  nano-SAXS: focused X-ray beam averages over small volumes
  it becomes possible see *local structures* and orientations
- real-space resolution:
  determined by beam size
  500 nm down to 50 nm
- scanning technique
  Example: 20 ms per point,
  250×250 points, 400 nm steps:
  100 µm square FOV, 40 minutes
- reciprocal-space resolution:
  scattering to largest *q*-vector,
  currently ~ 1 nm^{-1}
- label-free imaging of biological matter on the nano scale
- quantitative information channels available:
  darkfield (how many photons are scattered? – **measure the electron density**),
  differential phase contrast (where are the photons scattered? – gradients),
  azimuthal and radial analyses – quantify **local ordering and structures**:
  e.g. study the actin cytoskeleton and understand the physical parameters of cells

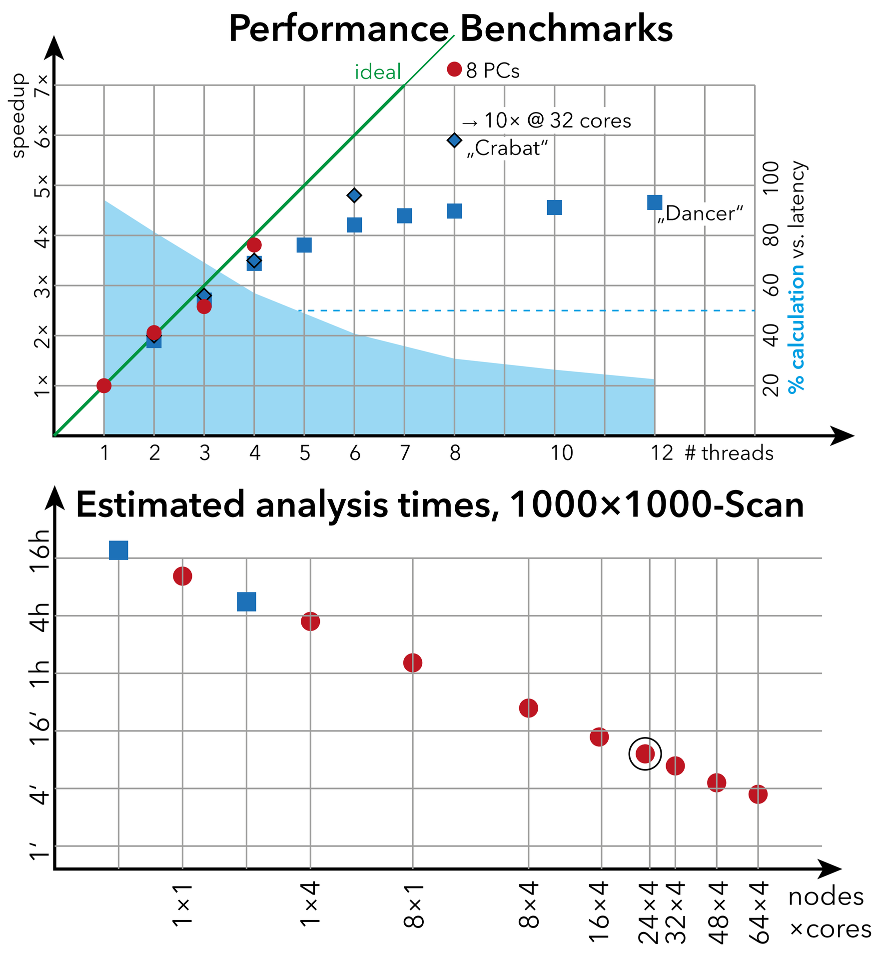
Typical frame and data rates:

- 10 to 100 Hz EigerX 4M
- (up to 750 Hz possible)
- 50 to 500 Mega Pixel per second
- ~ 50 MiB/s (compressed LZ4-HDF5),
- ~ 1 GiB/s (uncompressed for analysis)

Typical scan parameters:

- 50×50 to 250×250 (sometimes 1000×1000)
- from 1 ms to 50 ms per point
- 10^{11} pixels in less than half an hour

Poor scaling on multi-core-systems

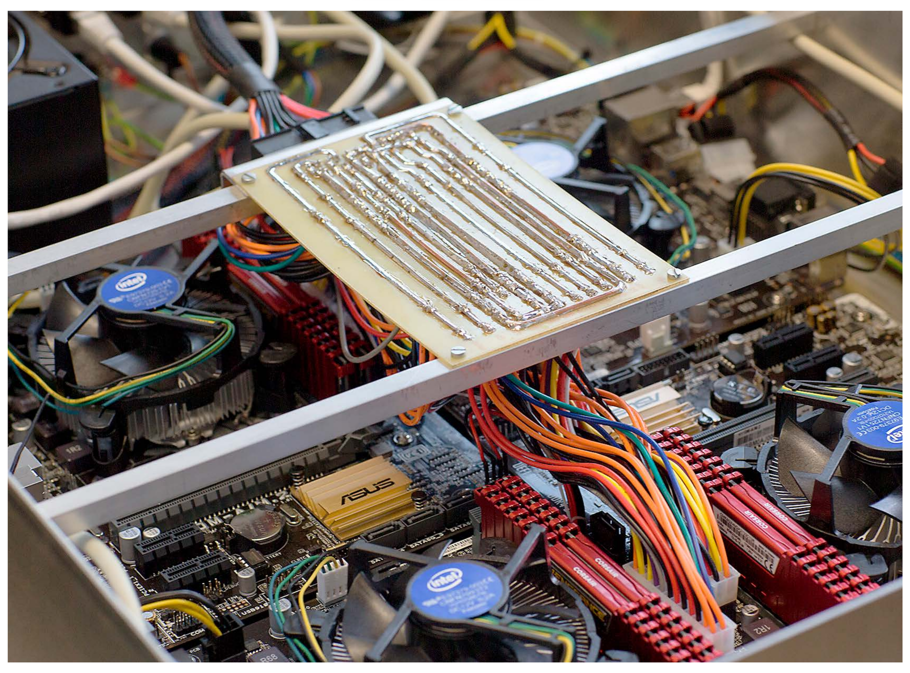


- current trend: more cores per CPU,
  but ≤ 2 MiB cache per core
- full EigerX 4M image:
  8 MiB @ int16_t
- limited speed-up on multi-core
- calculation time vs. latency decreases
- Bottleneck: CPU cache

  throughput from RAM okay,
  but latency too high

  multi-core analysis faster than
  data can flow into CPU
- 24 Heinzelmännchen: analysing 1000×1000 images in ten minutes

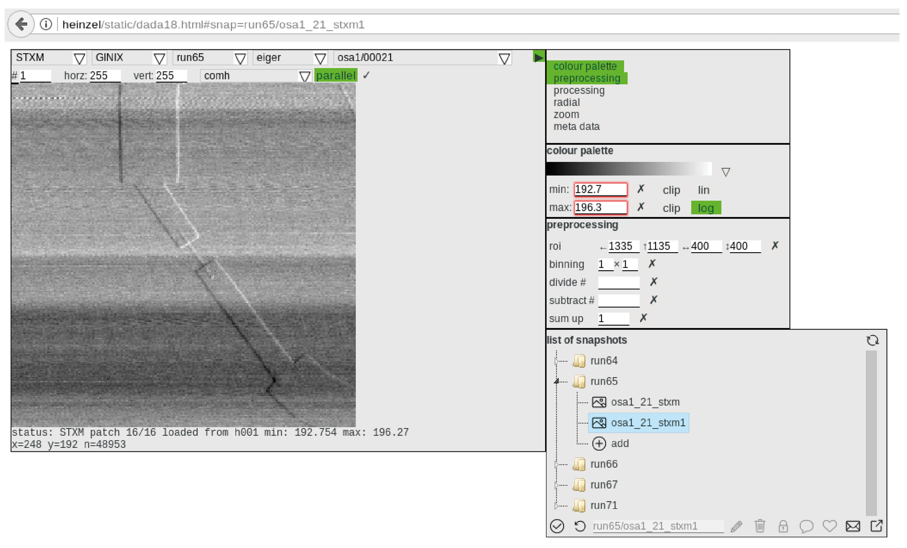
Remedy: more cache = more CPUs = more boards



- STXM-cluster of individual systems
- 24 systems ("Heinzelmännchen"):
  Boards: ASUS H110M-A M.2
  CPUs: Intel Core i7-7700 @ 3.60 GHz
  RAM: 8 GiB per node
- 1 control system ("Heinzelfrau"):
  CPU: Intel Core i7-8700 @ 3.20 GHz
  RAM: 64 GiB
  SSD: 1×Samsung 850 PRO 256 GiB
  cache: 5×Sandisk Ultra II 960 GiB
- network:
  2×10G uplink from Heinzelfrau,
  1G per node;
  10G extern to NFS server
- 96 cores, 192 MiB L3-Cache
  "cheap, but many"

  under construction *now*

dada18: web GUI and re-worked C-backend



- GUI to generate URLs for analysis
- browsing & pre-processing
  composites (2D array of 2D images)
  STXM analysis, different algorithms
- new: snapshots and parallel jobs

- dada, the *data daemon* [5]:
  centralised entry node to data
- do not worry about file name,
  folder structure, data format,
  compression et al. any longer
- reducing obstacles for new students

  centralised entry node to analysis
- collecting our student's developments,
  so the next generation can
  use old methods on new data
- after testing: optimising performance
- web GUI and for the user,
  HTTP interface for software

dadafs: network filesystem with multiple caches

h001 … h024 — 128 MiB RAM cache for metadata + recent data

2×10 G to 24×1 G — fuse based client, TCP + Jumbo frames, read-only

heinzelfrau — 3.4 TiB data cache, 128 GiB results cache, 60 GiB page cache

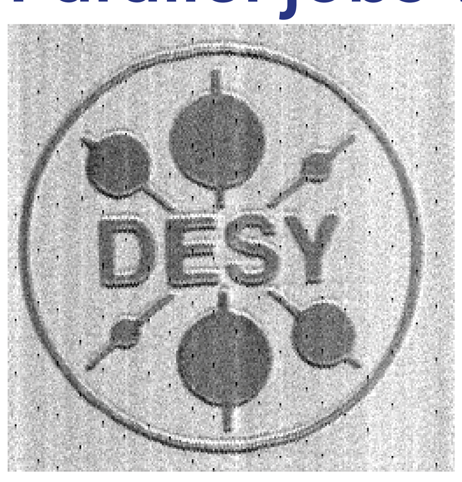
10 G Ether — NFS

File Server homer4b ← 2×8 G SAN — NetApp ≥ 150 TiB

- raw data accessible via NFS
  only visible on Heinzelfrau
- Heinzelmännchen mount via
  fuse-based dadafs-client
- local caching:
  recently accessed data,
  meta data, i.e. calls to stat(2)
- Heinzelfrau caching:
  dadafsd-server caches accessed
  file fragments on SSD (ZFS pool)

  results: cached in aerospike DB
  (distributed caching database)
- faster than NFS and Samba

Parallel jobs via HTTP proxy



single 255×255 STXM scan

```
http://heinzel/stxm/GINIX/
  run60/eiger/desylo/1/1
  ?horz=255&vert=255
```

+ parameter
+ colour map etc.

parallel regions, e.g.

```
…/1/    1?horz=255&vert=10,
…/1/2551?horz=255&vert=10,
…/1/5101?horz=255&vert=10,
…/1/7651?horz=255&vert=10
```

User	Proxy	dada18
web browser	lighttpd as load balancer	running on all Heinzelmännchen
dada18.html GUI to define analysis and parameters	requests to h001 … h024	load / process and analyse
dada18.js generate URL	caching database	sends results – to user – to cache
	aerospike	

- parallelisation is "easy":
  many independent Eiger images
- optimal strategy depends on
  scan geometry; e.g. stitching-STXM
- full scan is broken down into
  patches with individual URLs
- multiple requests to HTTP proxy,
  lighttpd acts as load balancer
- patches are "glued" to full dataset
- all data can be imported from e.g.
  Matlab / Python / whatever by URLs

  URLs are "human-readable"

References

[1] M. Bernhardt, J.D. Nicolas, M. Eckermann, B. Eltzner, F. Rehfeldt, T. Salditt:
Anisotropic x-ray scattering and orientation fields in cardiac tissue cells, New Journal of Physics 19, 2017.

[2] B. Weinhausen, J.F. Nolting, C. Olendrowitz, J. Langfahl-Klabes, M. Reynolds, T. Salditt, S. Köster:
X-ray nano-diffraction on cytoskeletal networks, New Journal of Physics 14, 2012.

[3] M. Priebe, M. Bernhardt, C. Blum, M. Tarantola, E. Bodenschatz, T. Salditt:
Scanning X-Ray Nanodiffraction on Dictyostelium discoideum, Biophysical Journal 107, 2014).

[4] J.-D. Nicolas, M. Bernhardt, M. Krenkel, C. Richter, S. Luther and T. Salditt:
Combined scanning X-ray diffraction and holographic imaging of cardiomyocytes, J. Appl. Crystallogr. 50, 2017.

[5] M. Osterhoff: dada – a web-based 2D detector analysis tool, J. Phys: Conf. Ser. 849, 2017 (XRM 2016).

Acknowledgements

We thank our workshops (electrical and mechanical), especially Thomas Pingel, for support and construction.

We thank DESY Photon Science, especially André Rothkirch, for discussion and support during beamtime.