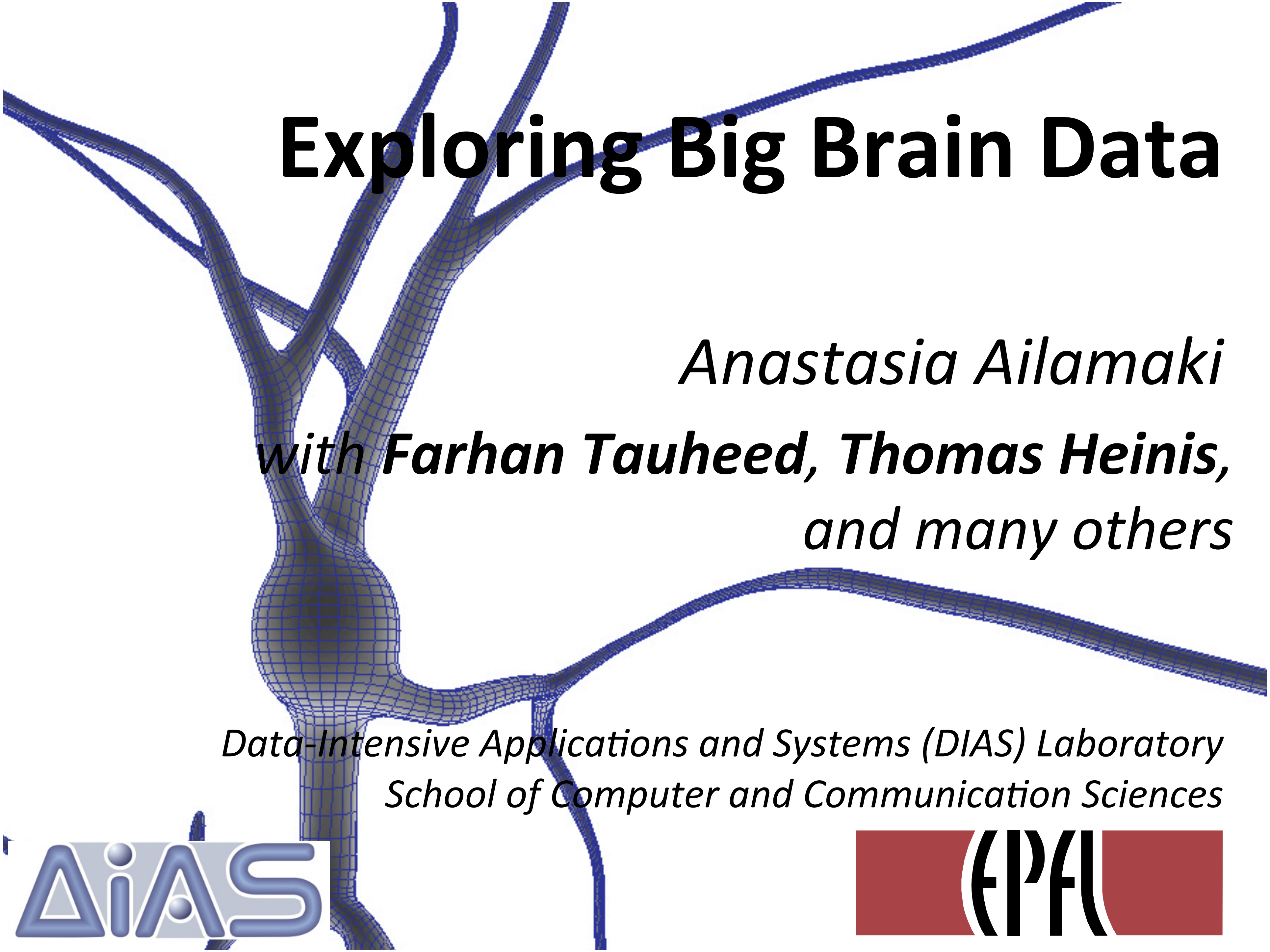


Exploring Big Brain Data

Anastasia Ailamaki

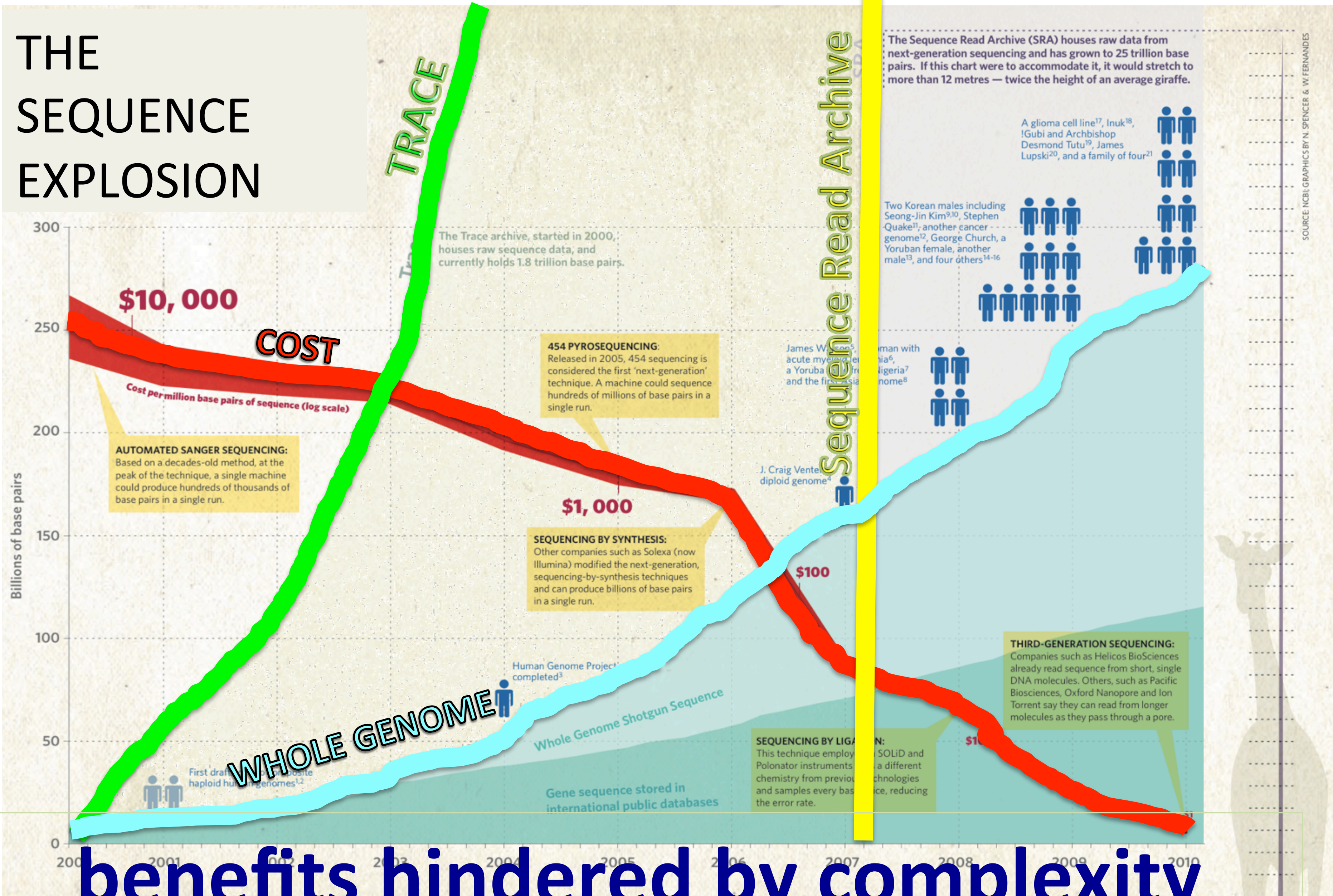
with ***Farhan Tauheed, Thomas Heinis,***

and many others

Data-Intensive Applications and Systems (DIAS) Laboratory

School of Computer and Communication Sciences

the human genome at ten (nature, 4/2010)



benefits hindered by complexity

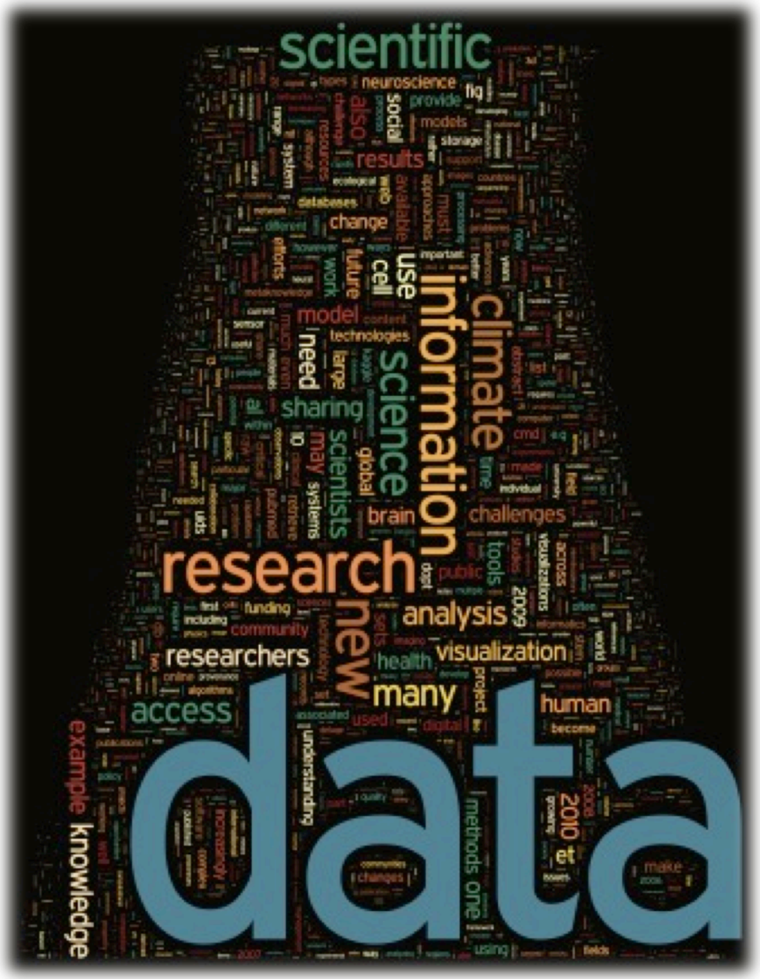
scientific questions and answers

Past three paradigms:

- theory
- simulation
- experimentation

fourth paradigm:

data-driven science



wasted data → worst ROI ever!

questions to answer today

Domain Scientist:

- how to find interesting data?
- how to move quickly through data?
- how to enable scientific discovery?

Computer Scientist:

- What's in it for me?

How to find interesting data:

SPATIAL QUERIES ON UNSTRUCTURED MESHES

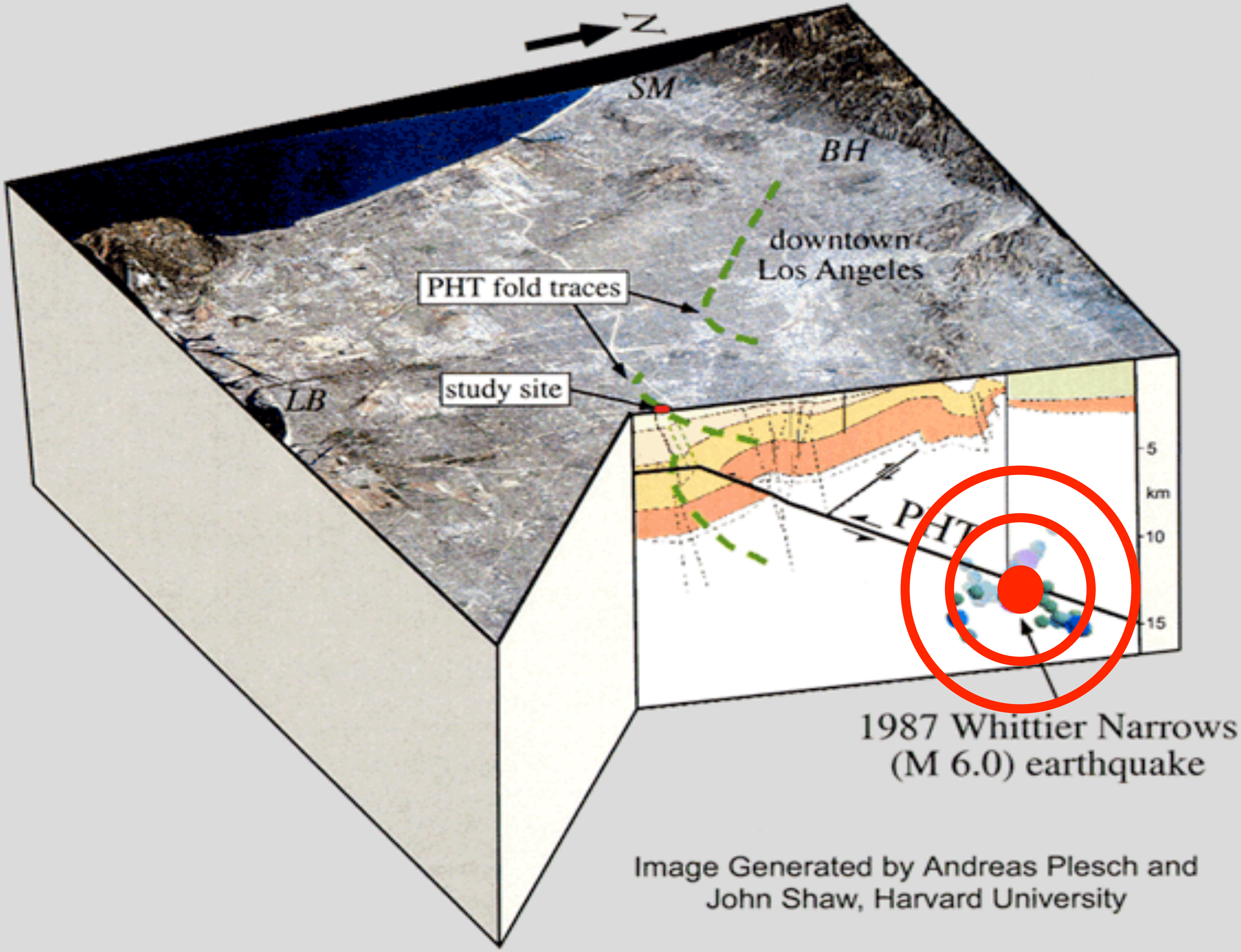
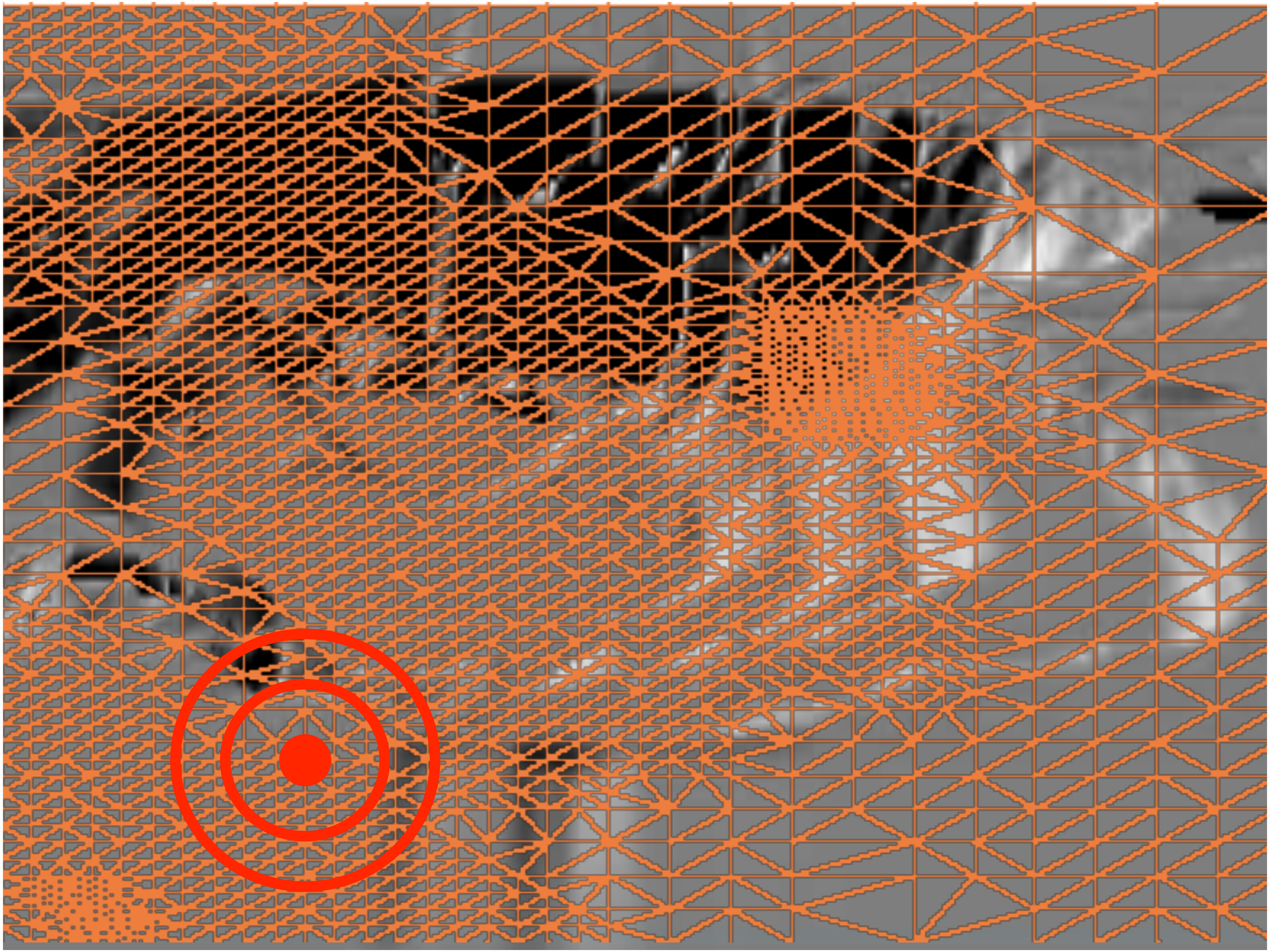
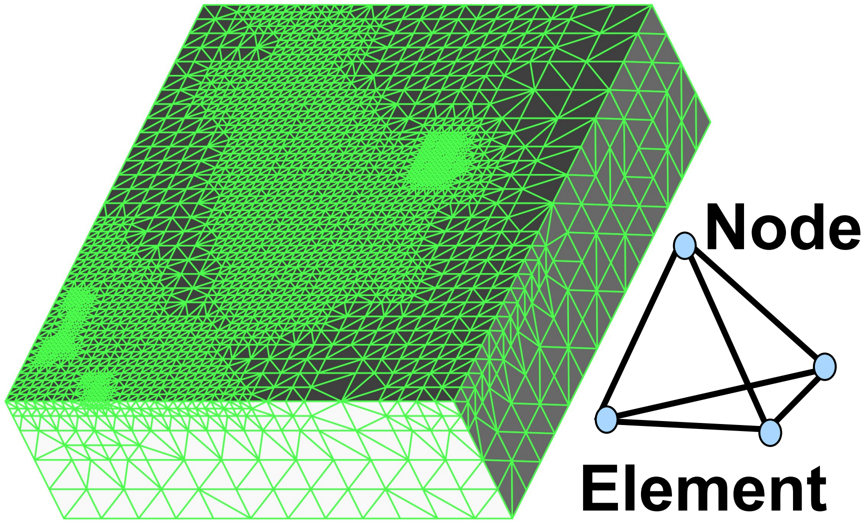
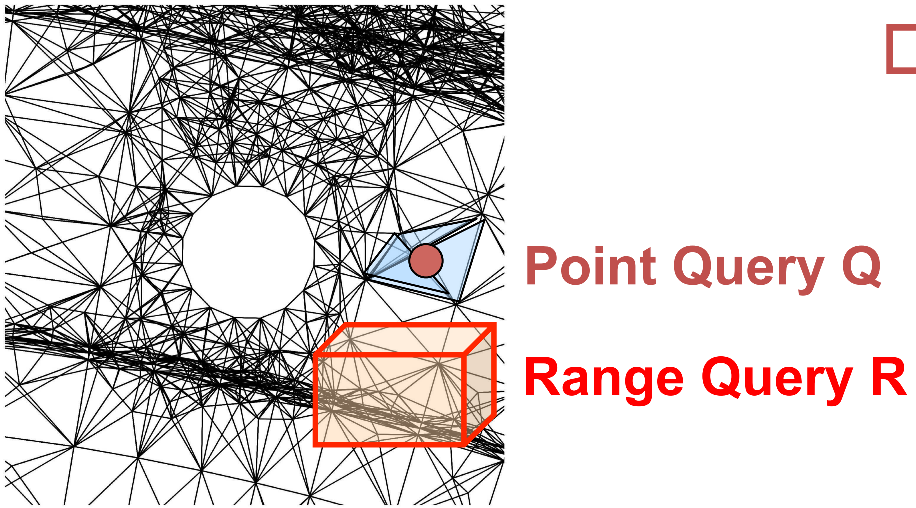


Image Generated by Andreas Plesch and
John Shaw, Harvard University



spatial range queries

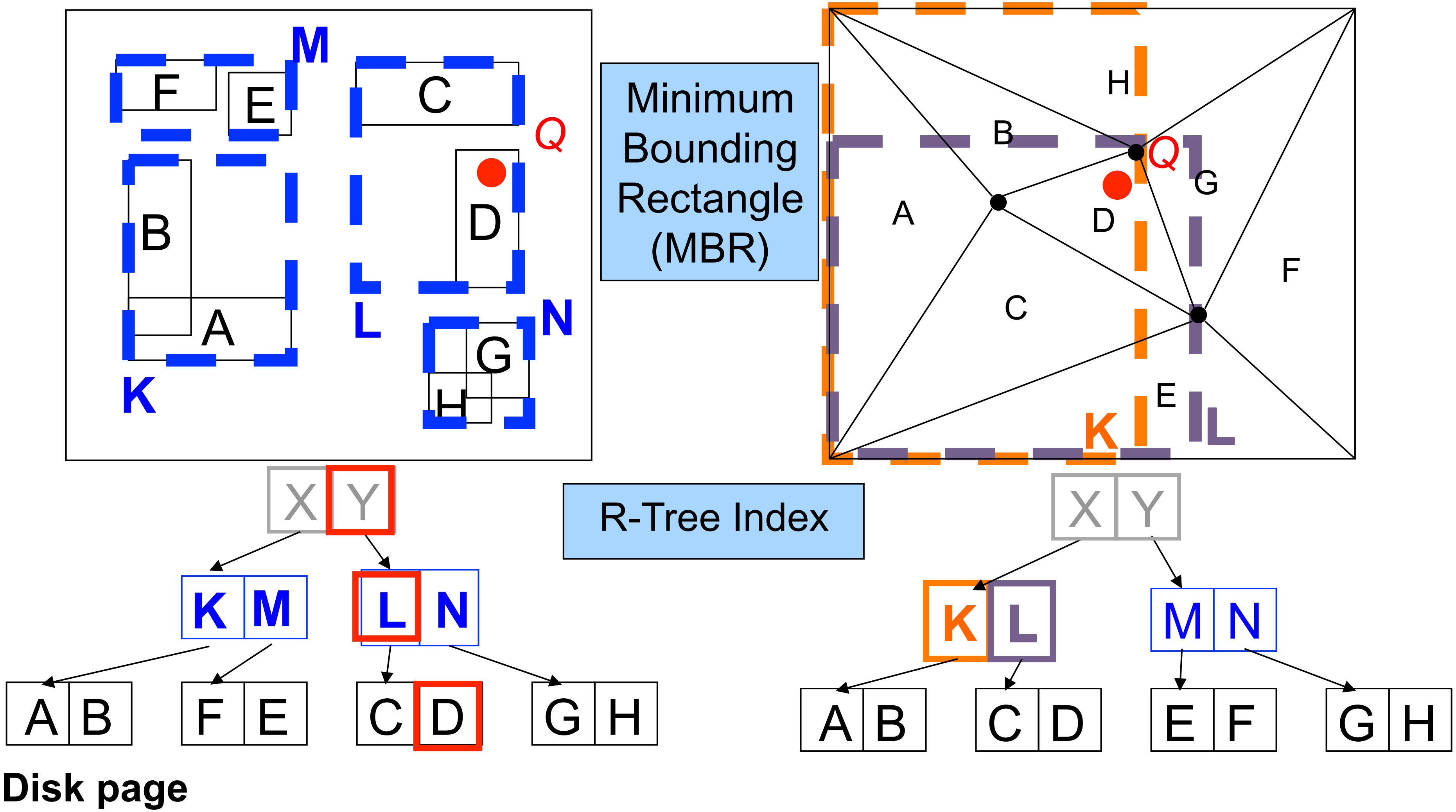




- Queries
  - Point
  - Range
  - Feature
- Use in simulation
  - Show ground velocity at **Q**
  - Draw the temperature of **R**

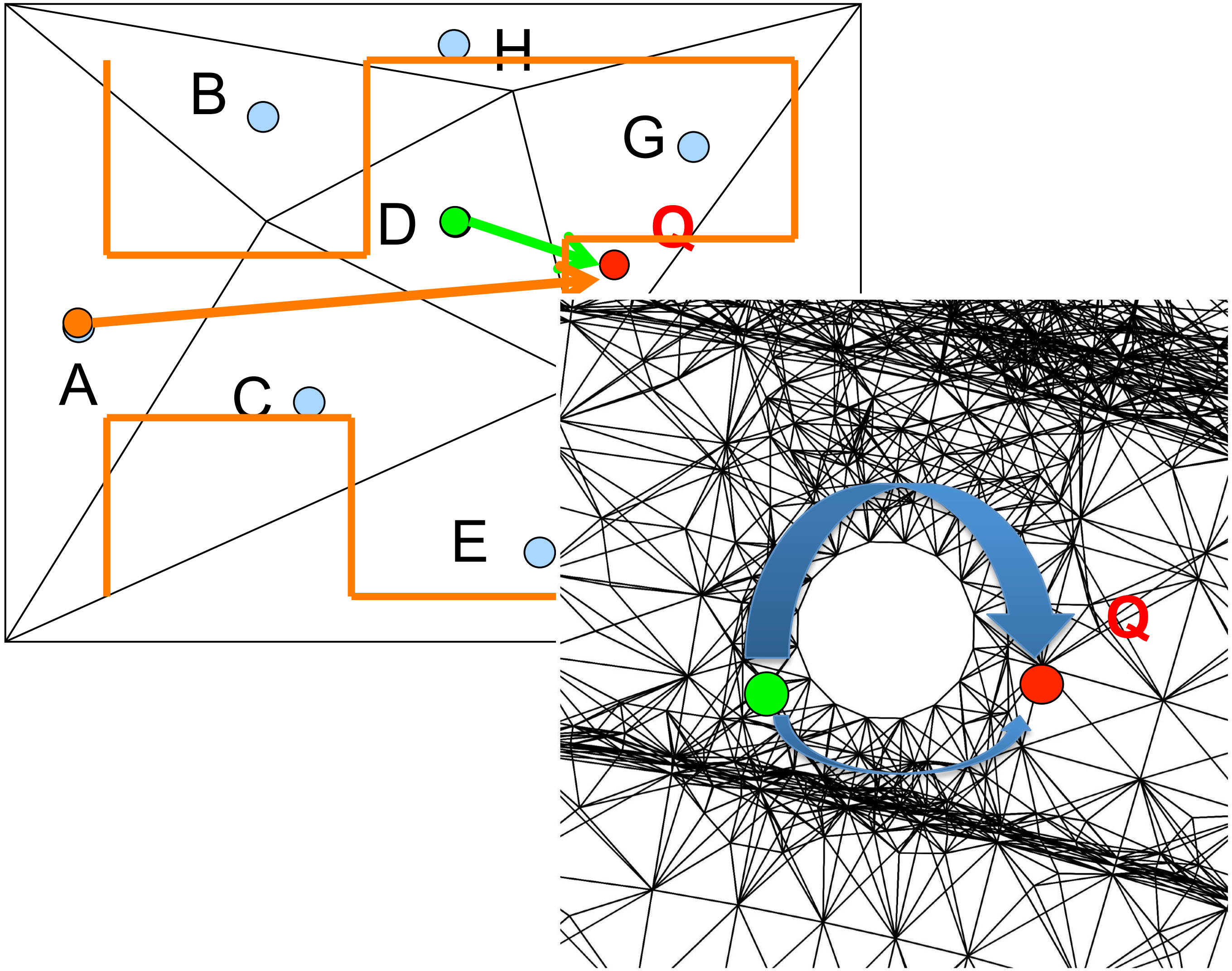
must scale with model complexity

R-Tree indexing



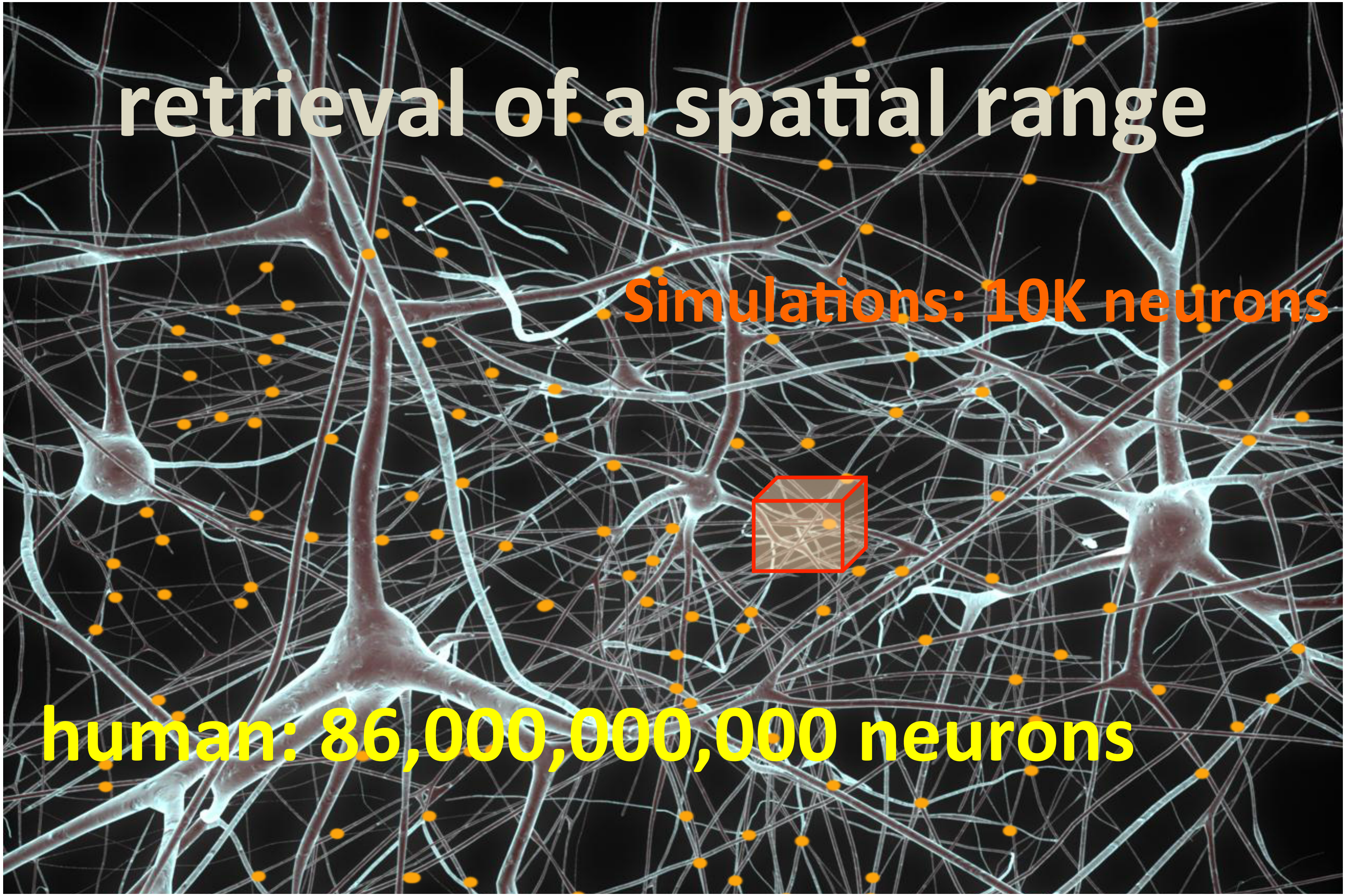
tight connectivity hurts performance

directed local search



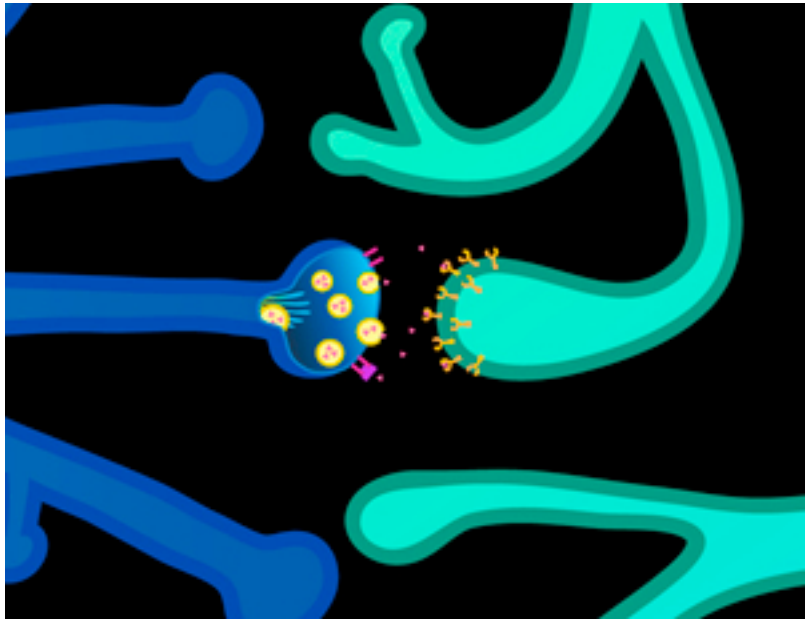
insight:

trade accuracy of target with efficiency



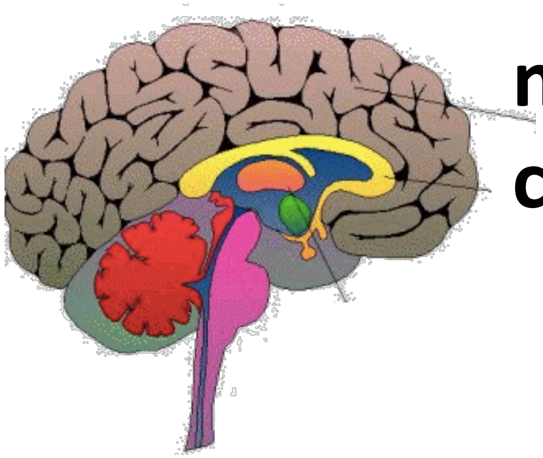
querying increasingly *denser* data

brain diseases in Europe: €800B



?

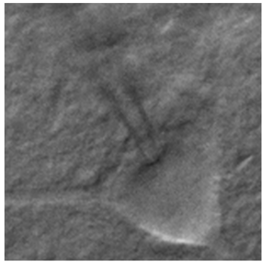
the Blue Brain Project



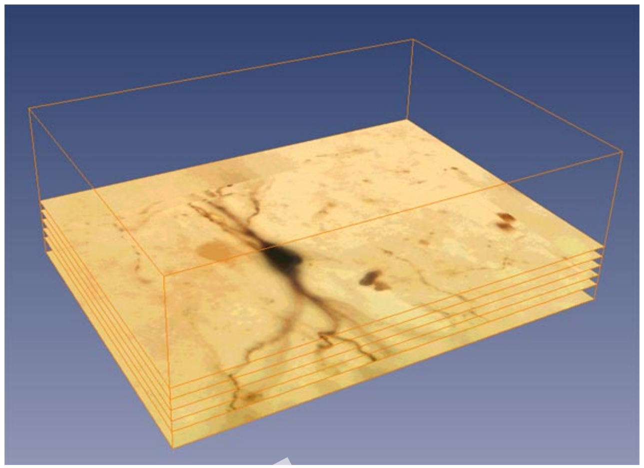
neo cortex

images courtesy of the Blue Brain Project

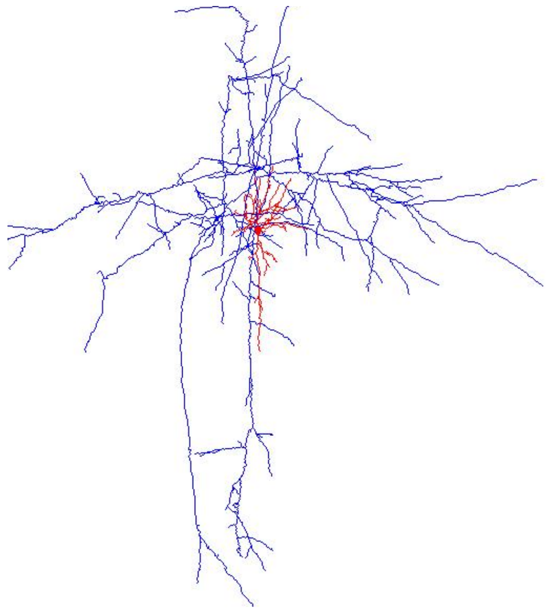
dye loading & raster scanning



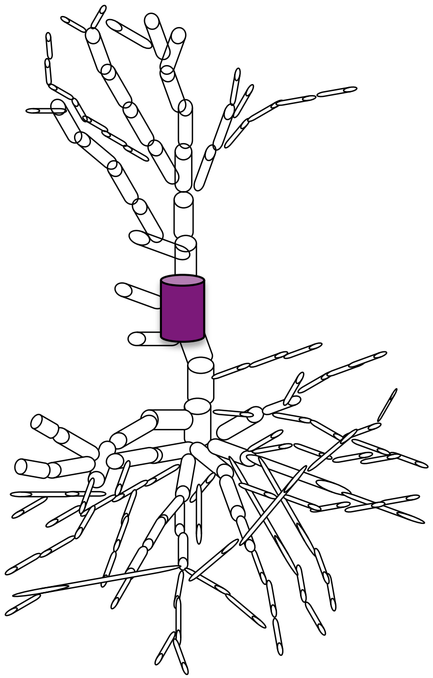
3D reconstruction



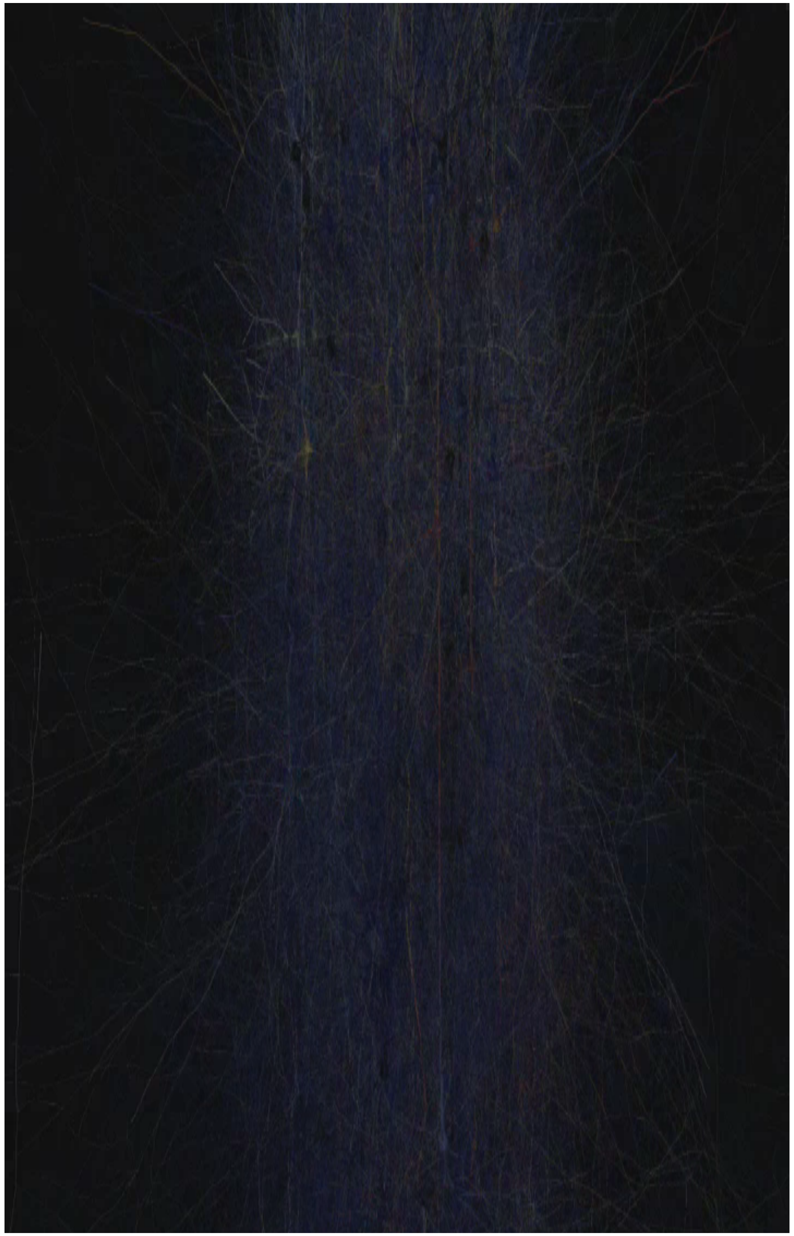
a neuron



brain simulation

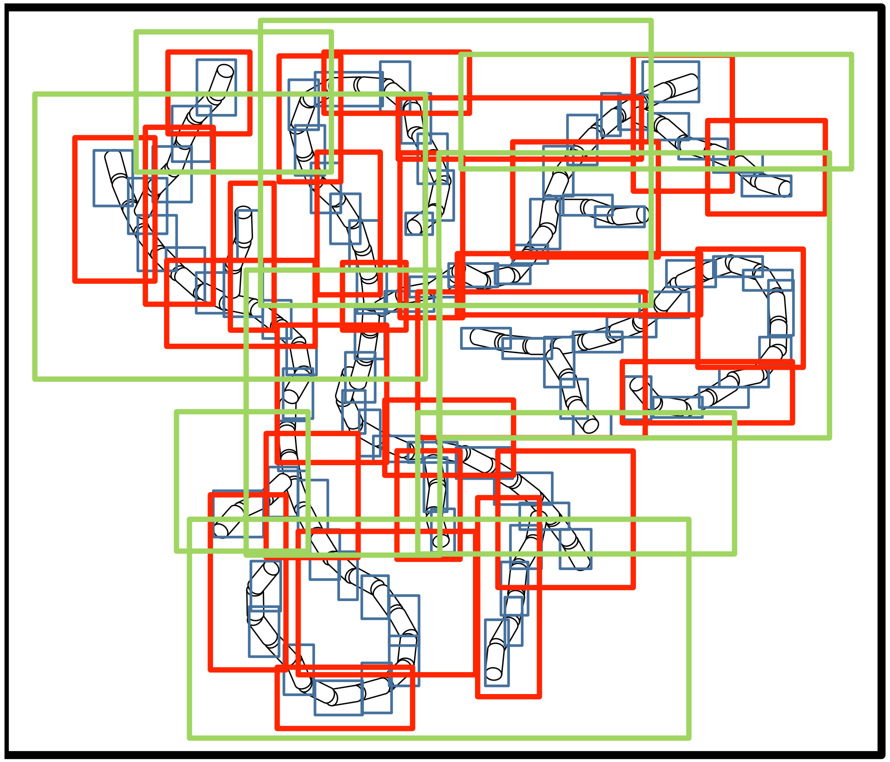


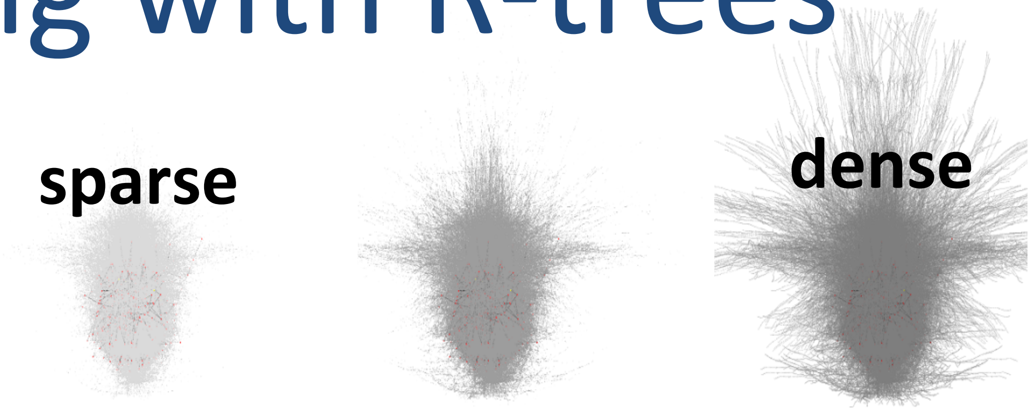
single neuron,
modeled with 3D cylinders

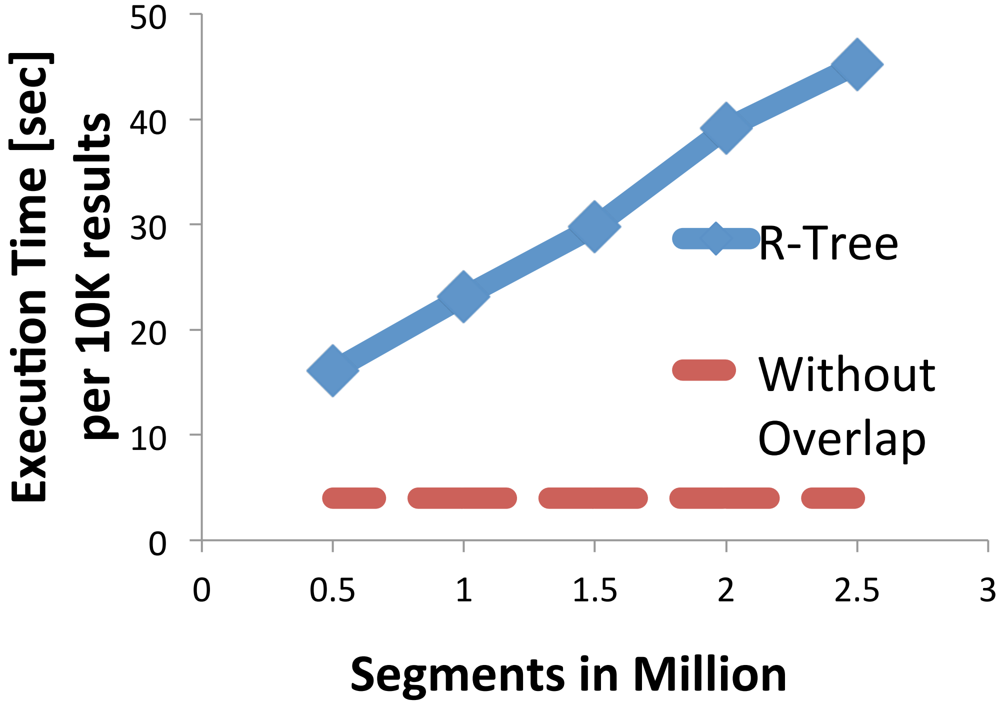


idea: index the brain

brain indexing with R-trees







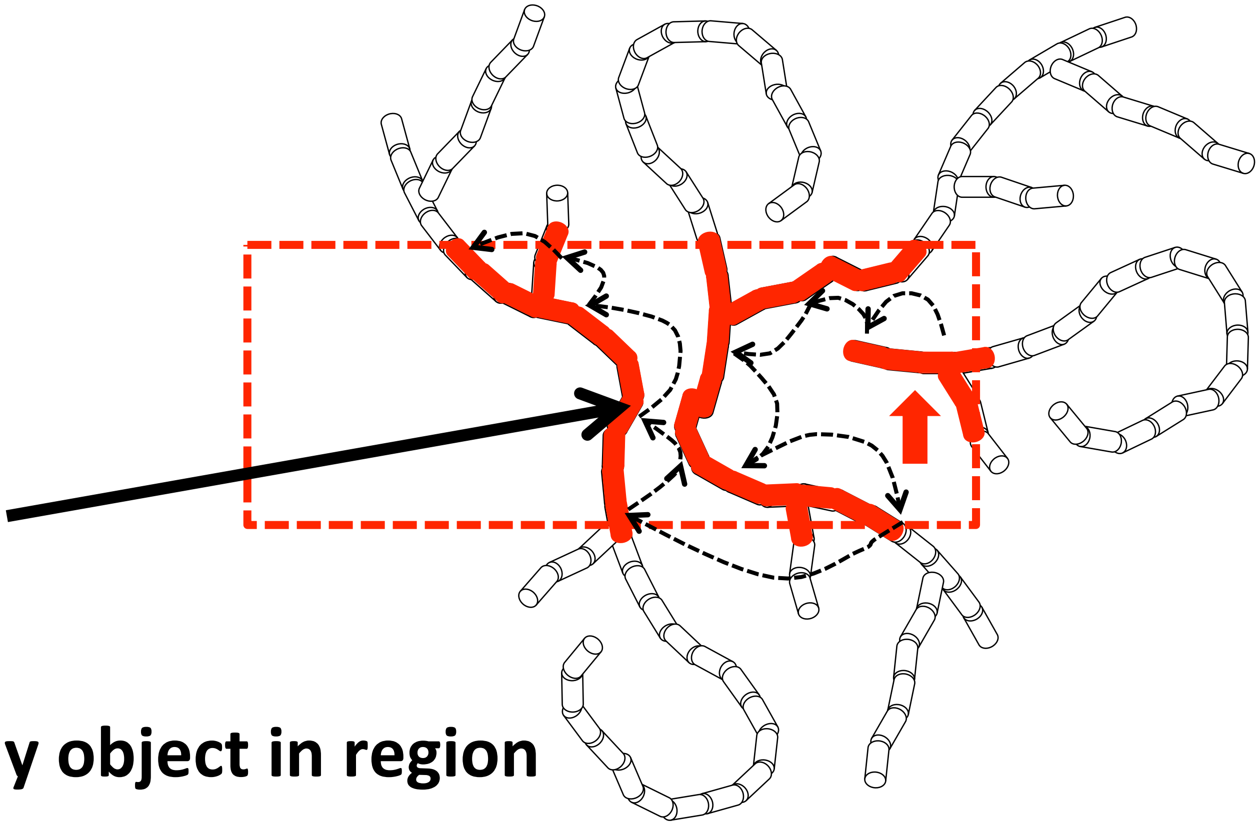
Bulk Loaded R-Trees:

Hilbert packed R-Tree,

need to scale with density

FLAT! idea: seed-then-crawl

Reachability?

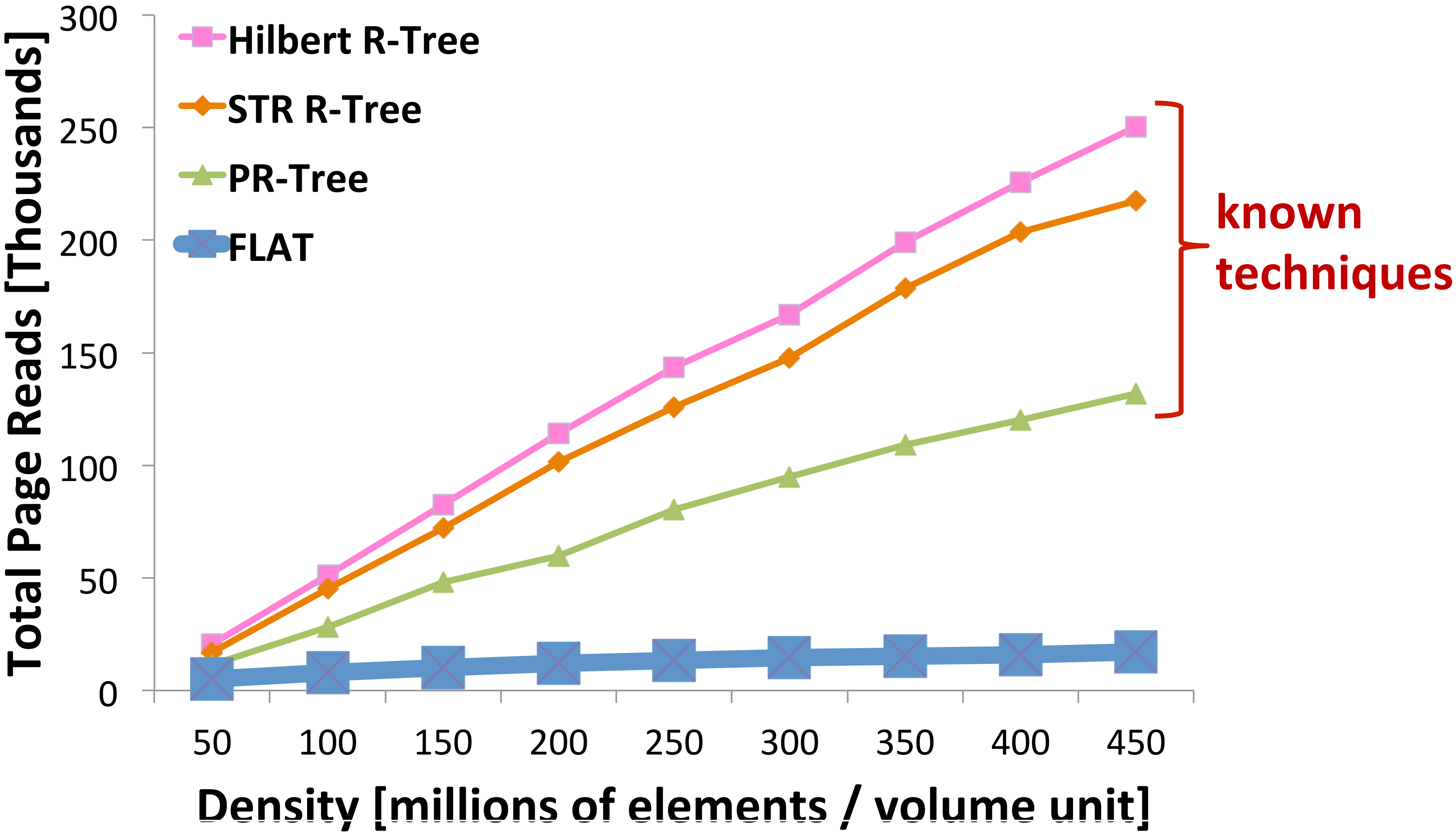


1) SEEDING: find any object in region

2) CRAWLING: traverse and retrieve remaining objects

tesselation + linking enable crawling

FLAT scales with data density



2012: 1 million neurons

How to move quickly through data:

FAST DATA EXPLORATION THROUGH INTERACTIVE QUERIES

structural analysis: navigation

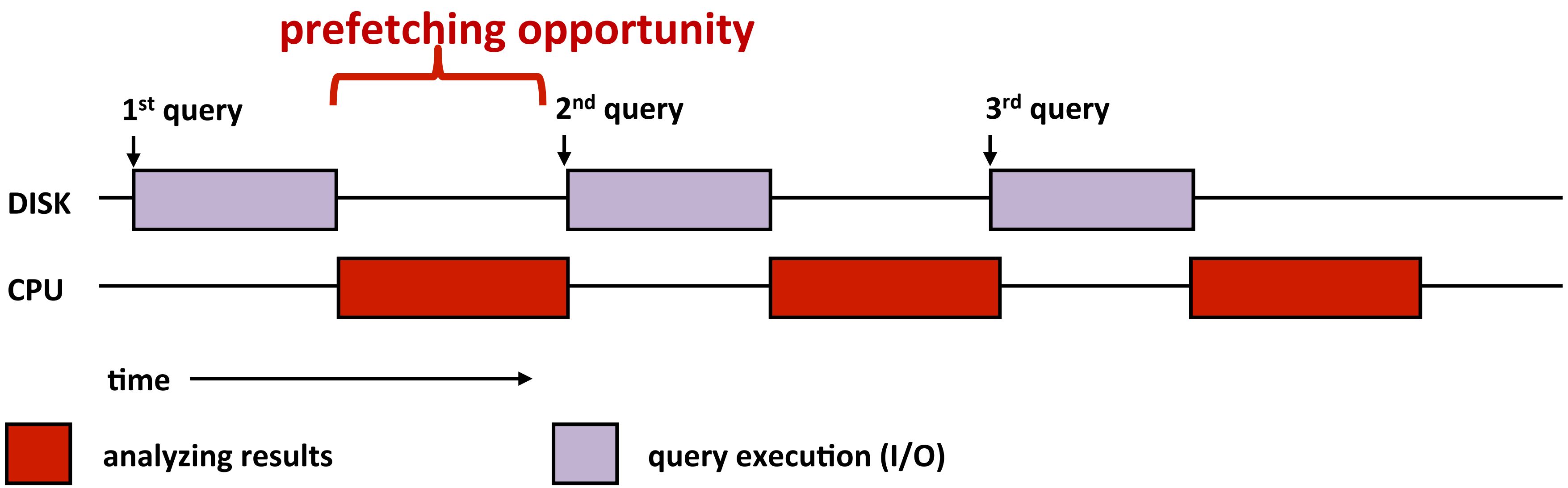
use cases:

- ad-hoc analysis
- visualization
- model refinement

sequences follow a **latent** guiding structure

idea: prefetch next query

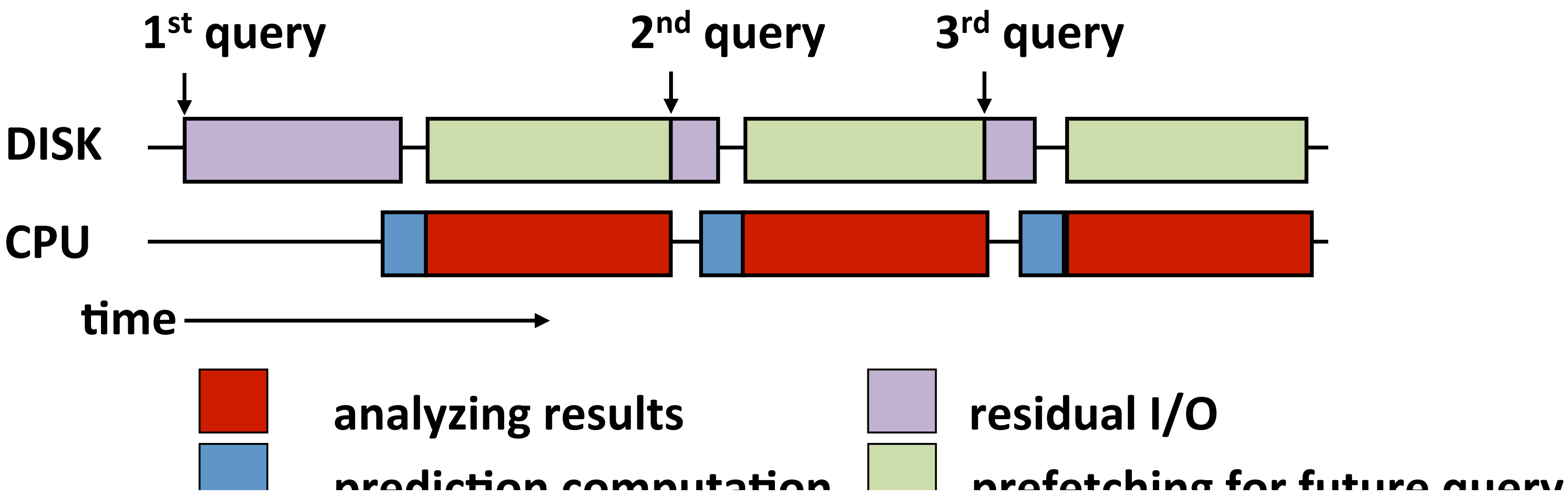
sequence of queries issued **interactively**



known techniques do not scale

SCOUT: **content-aware** prefetching

- model structures in graph, then traverse
- identify guiding structure: iterative pruning
- max accuracy: incremental prefetching



70-98% hit rate, 4x-15x speedup

How to enable scientific discovery:

QUERY PROCESSING ALGORITHMS

touch detection

Model Synapses

electrical connections betw. axons and dendrites

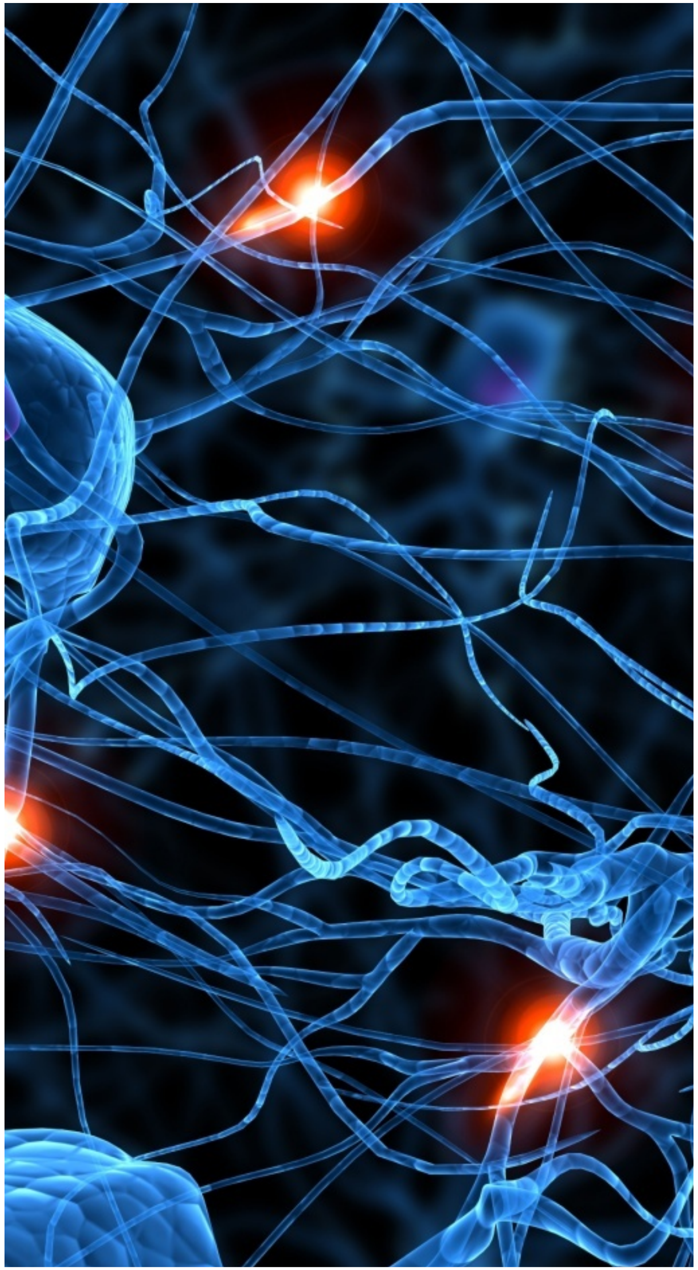
Data Challenge

100K neurons => 5B synapses

30GB addl space to store synapse data

Human Brain => 100B neurons ~PBs

- efficient spatial proximity queries
- precise distance calculation

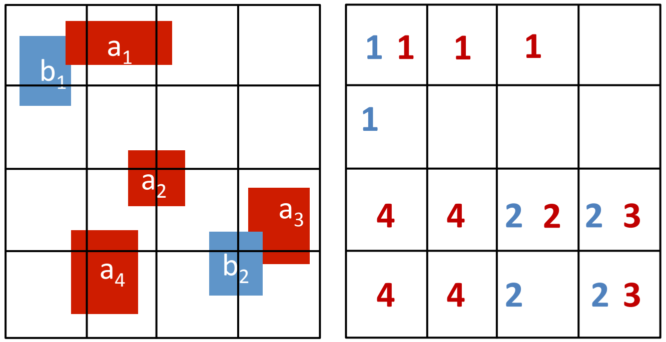


a major bottleneck in brain simulation

disk-based spatial join

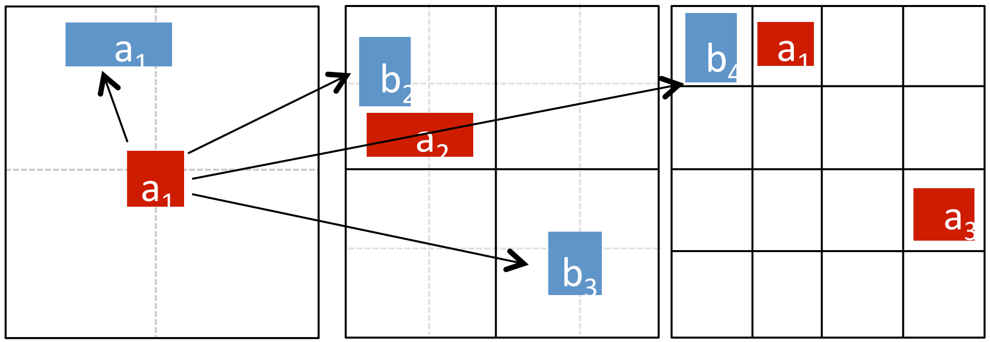
Dataset A ⋈ **Dataset B**

data partitioning



Multiple Assignment : **PBSM**

space partitioning



Multiple Matching : **S3**

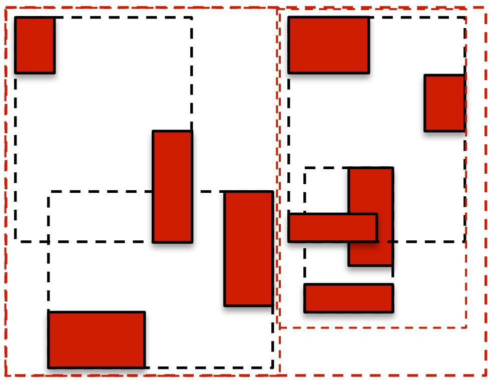
Algorithm	Object Replication	Sensitive to Distribution	Duplicate Results	Filtering
PBSM	☹	☺	☹	☺
S3	☺	☹	☺	☹

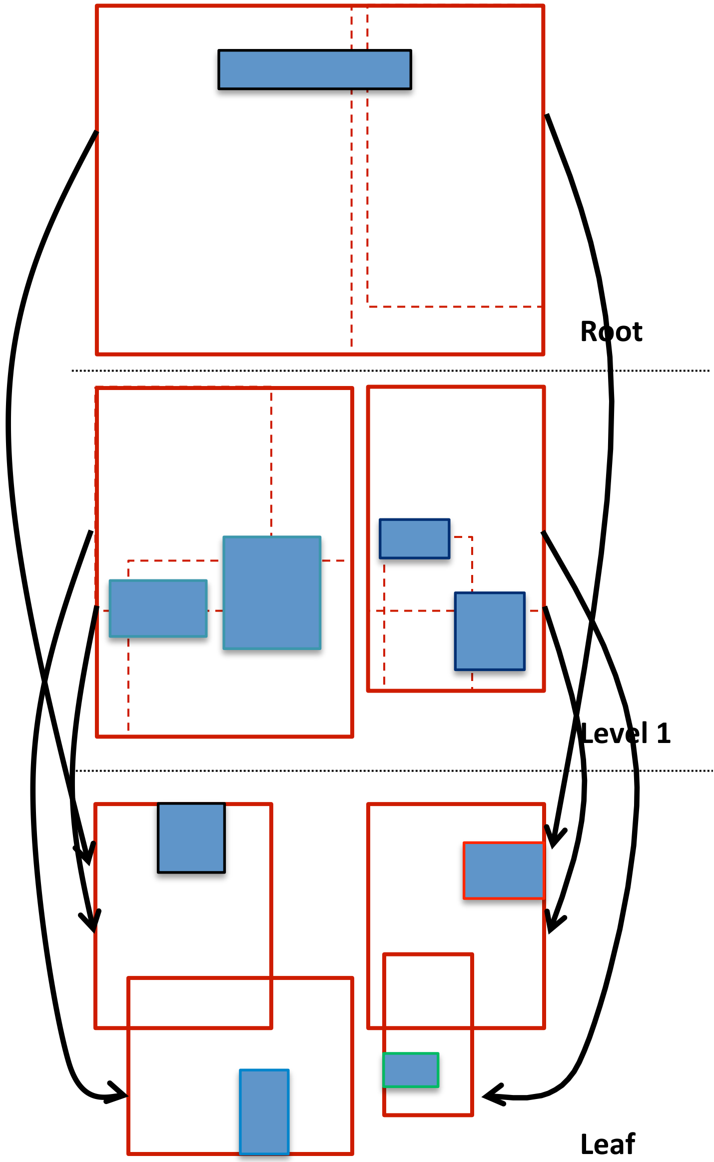
TOUCH: in-memory join

Phases:

1) **Building** (A)
2) **Assignment** (B)
3) **Join** (Plane Sweep)

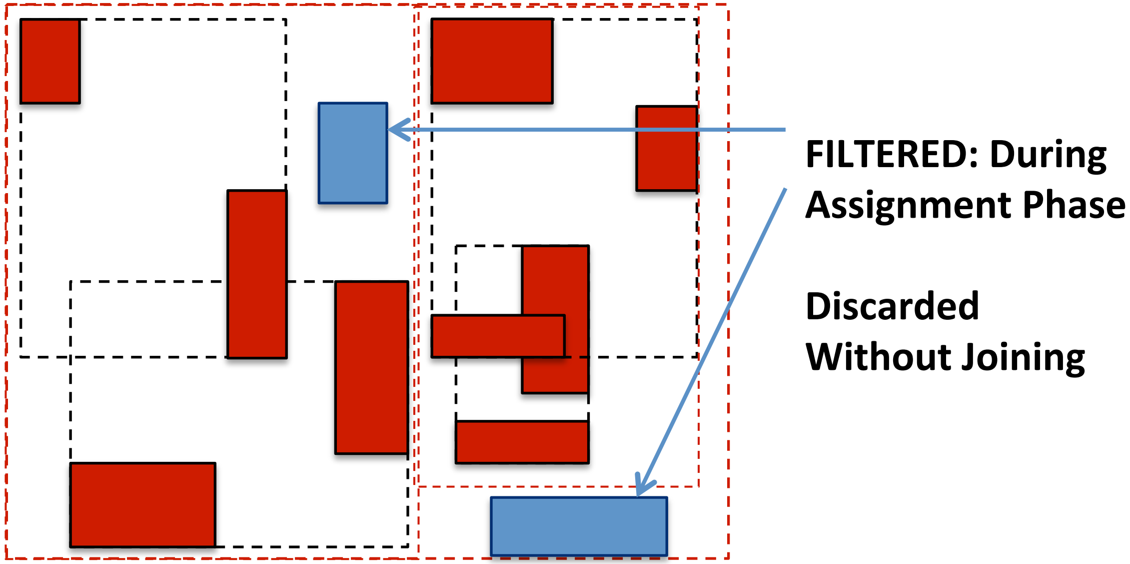
Dataset A ⋈ Dataset B





A-data, B-space

TOUCH is a distribution-aware join



Algorithm	Object Replication	Sensitive to Distribution	Duplicate Results	Filtering
PBSM	☹	☺	☹	☺
S3	☺	☹	☺	☹
TOUCH	☺	☺	☺	☺

simulation trace analysis

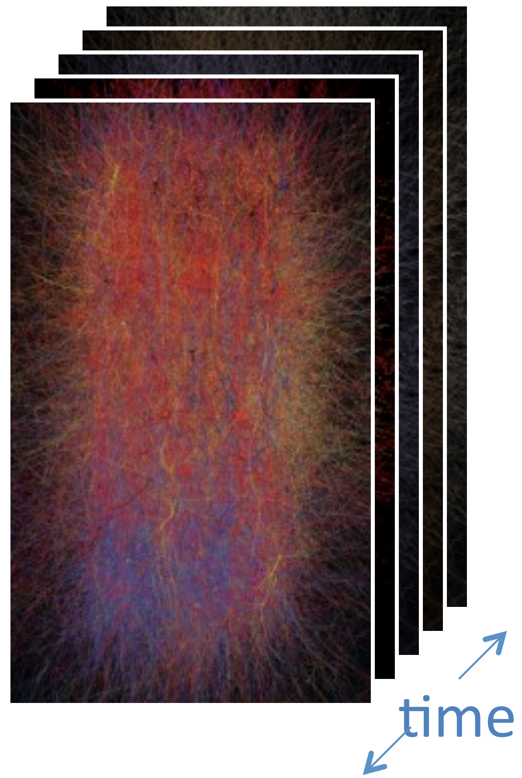
Need *accurate and fast* queries to

- Discover and explore neuro-circuit behavior
- Compare to behavior of biological tissue
- Understand plasticity

Typical trace file ~0.8TB

for 100K neurons

for only 1 second of simulation



In-memory efficient access method limit use of complex query analysis

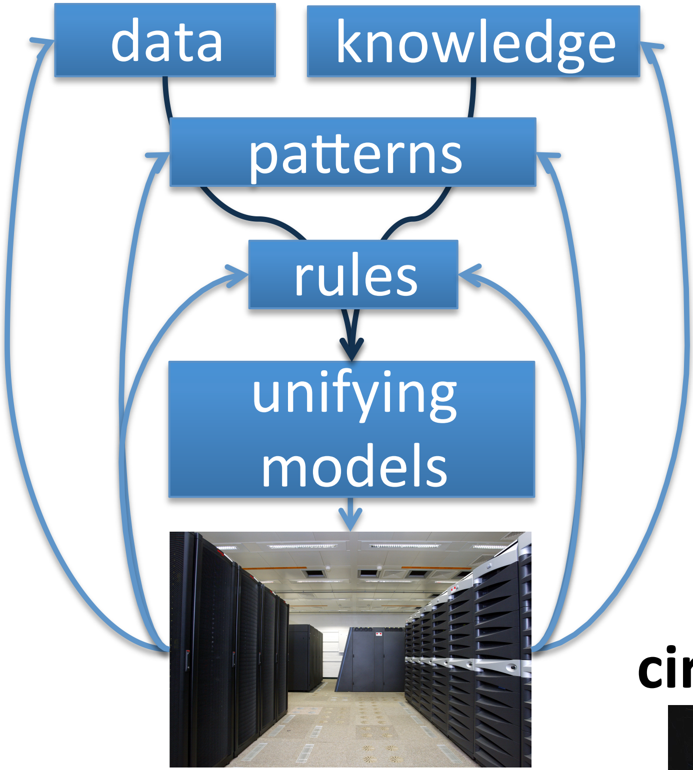
Storage capacity limits longer simulation time

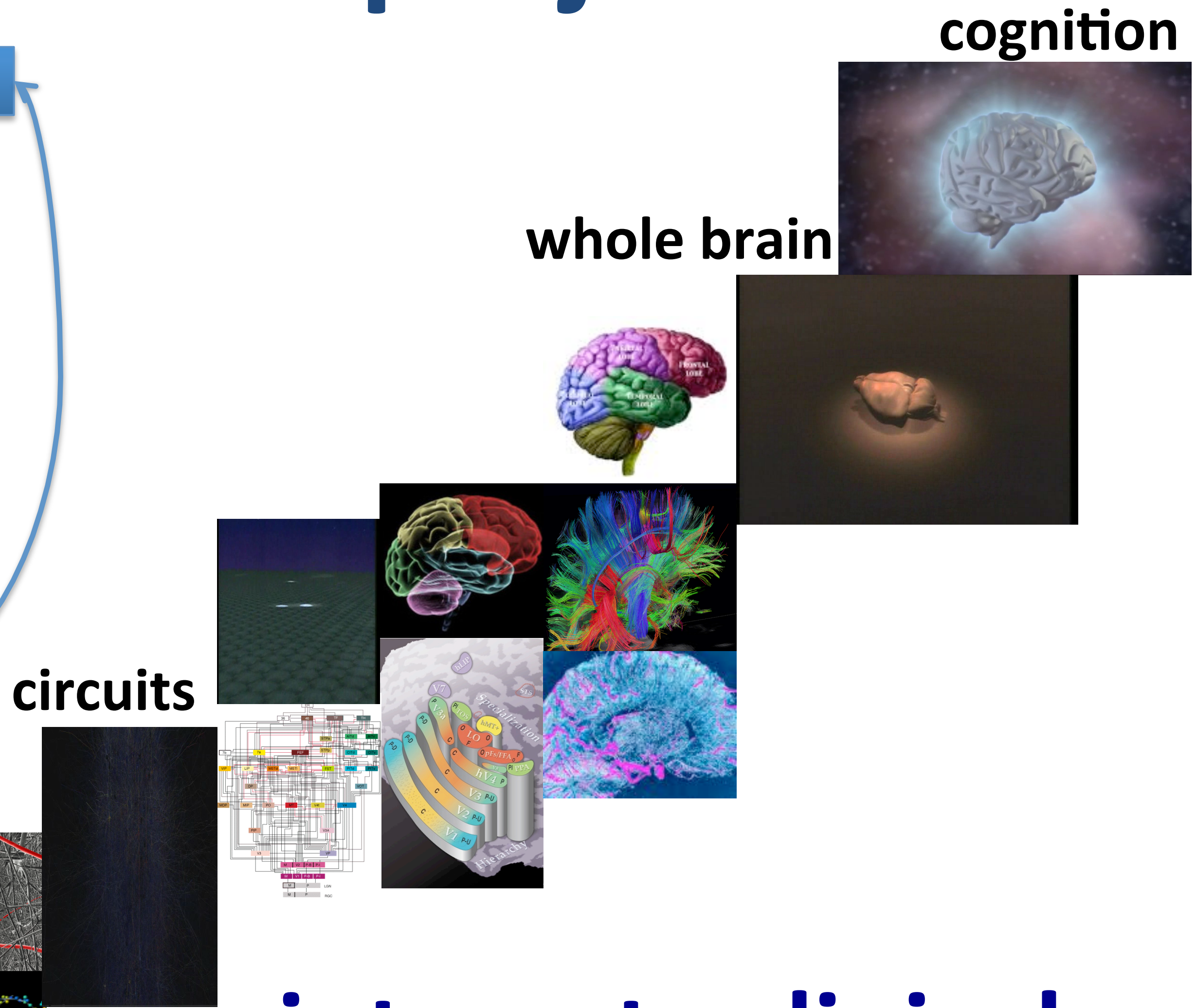
on-demand tracking moving data

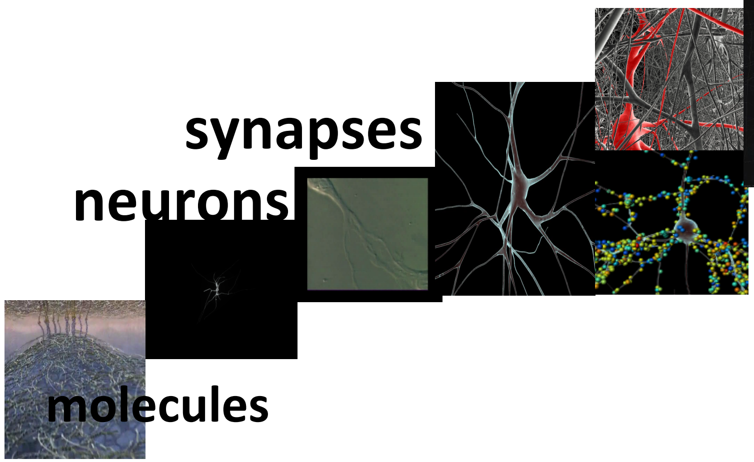
What’s in it for computer science:

NEXT-GENERATION QUERY ENGINES

the human brain project



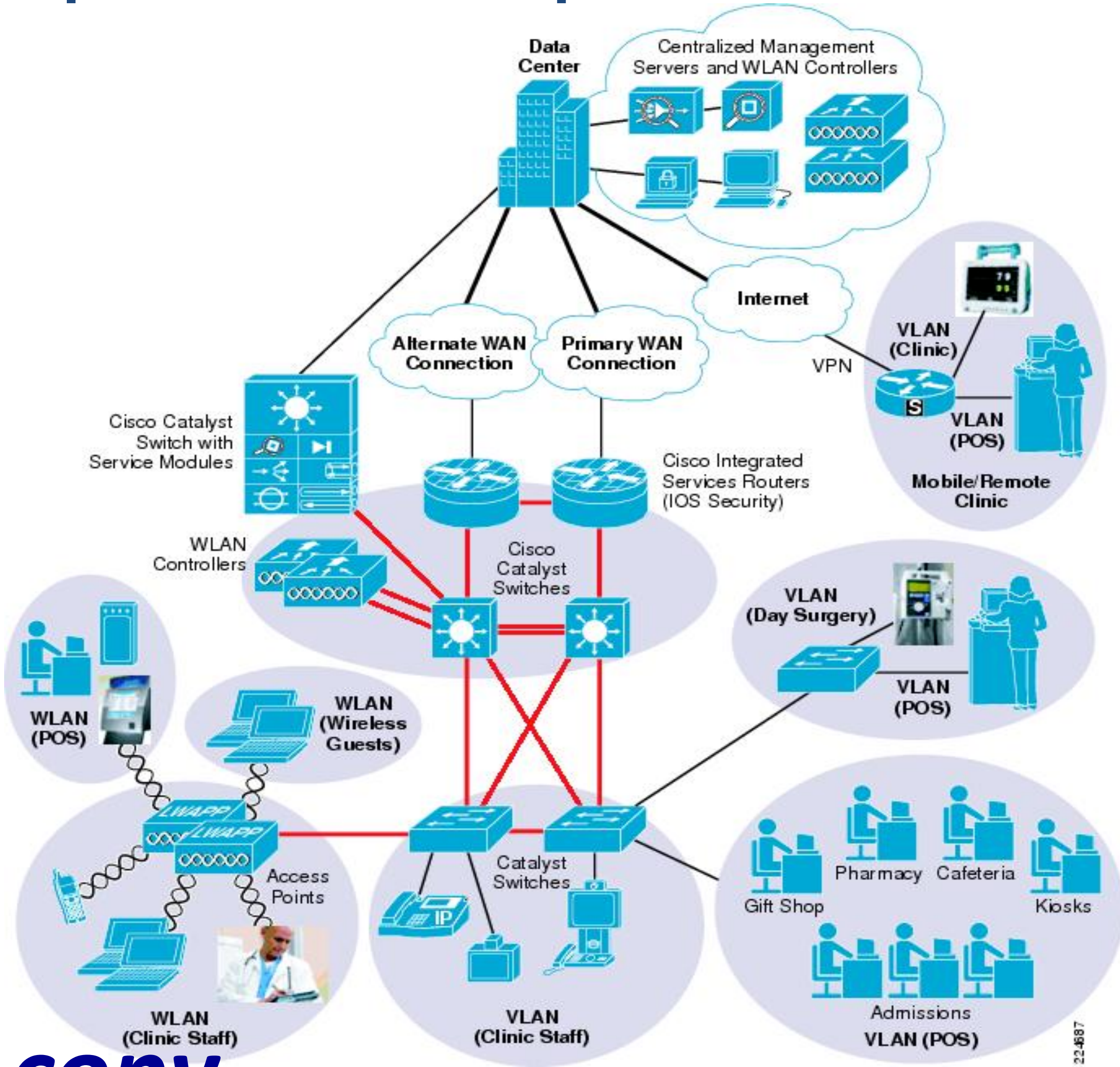




integrate clinical and simulation data

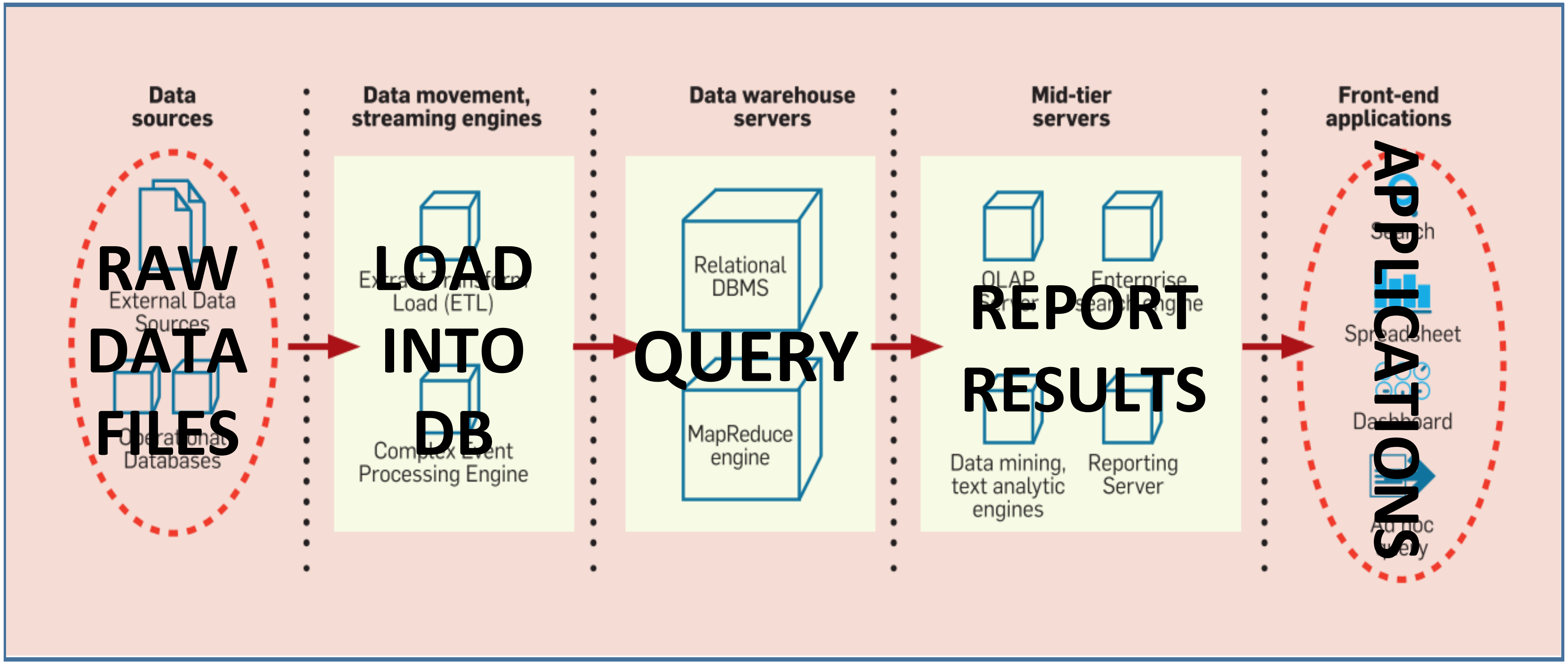
images from the Blue Brain Project, EPFL

access to private hospital data



no move, no copy

BI: create database to run queries



Source: ***"An Overview of Business Intelligence Technology".***
S. Chaudhuri, U. Dayal, V. Narasayya. CACM August 2011

data "locked" in DB for performance

NoDB: In-situ queries over never-before-seen data

Positional Maps

Caching

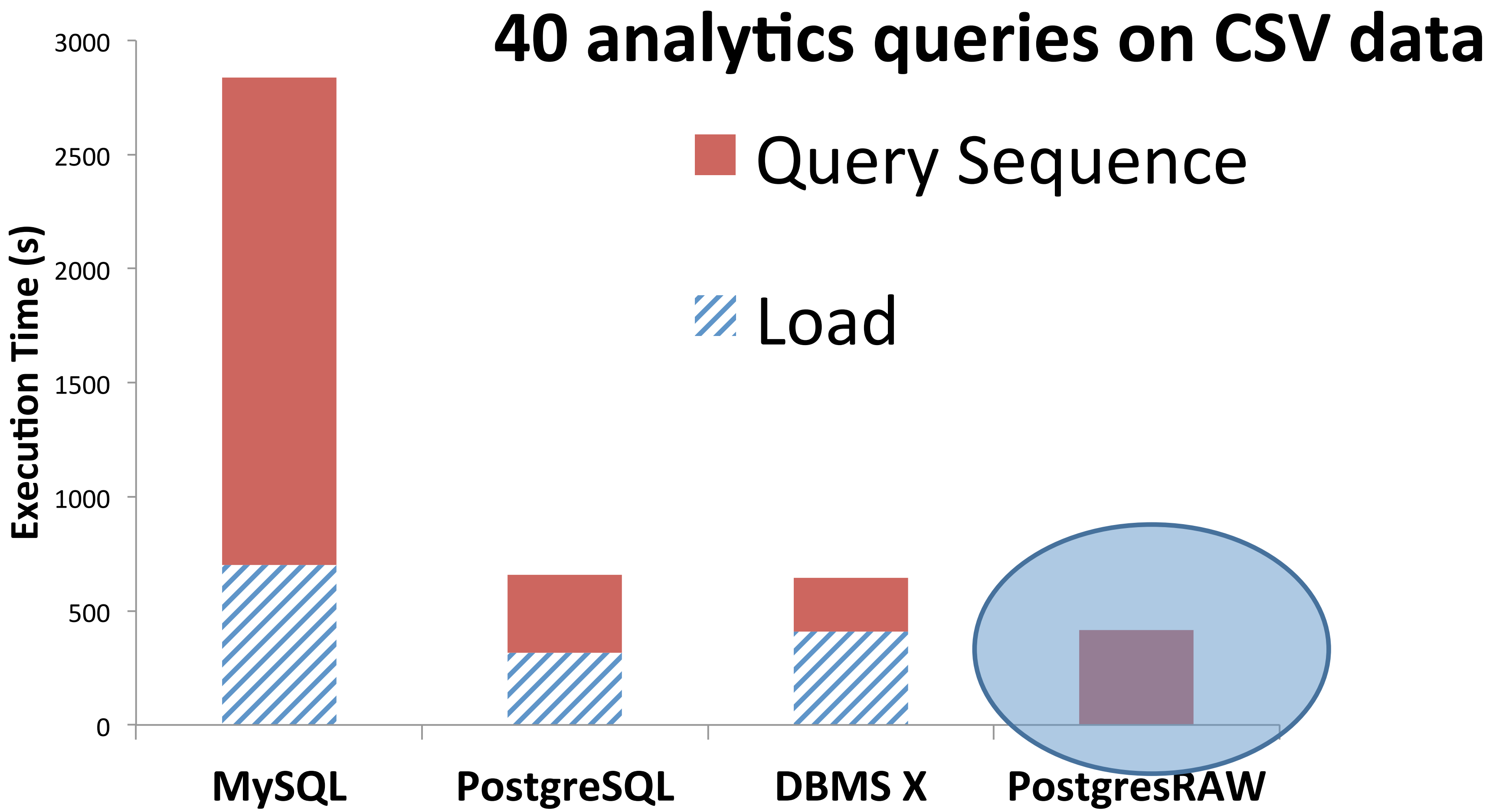
User *does not need to control* when, what, how or where data is cached

!= Classical Data Loading

also: indexing, file system integration

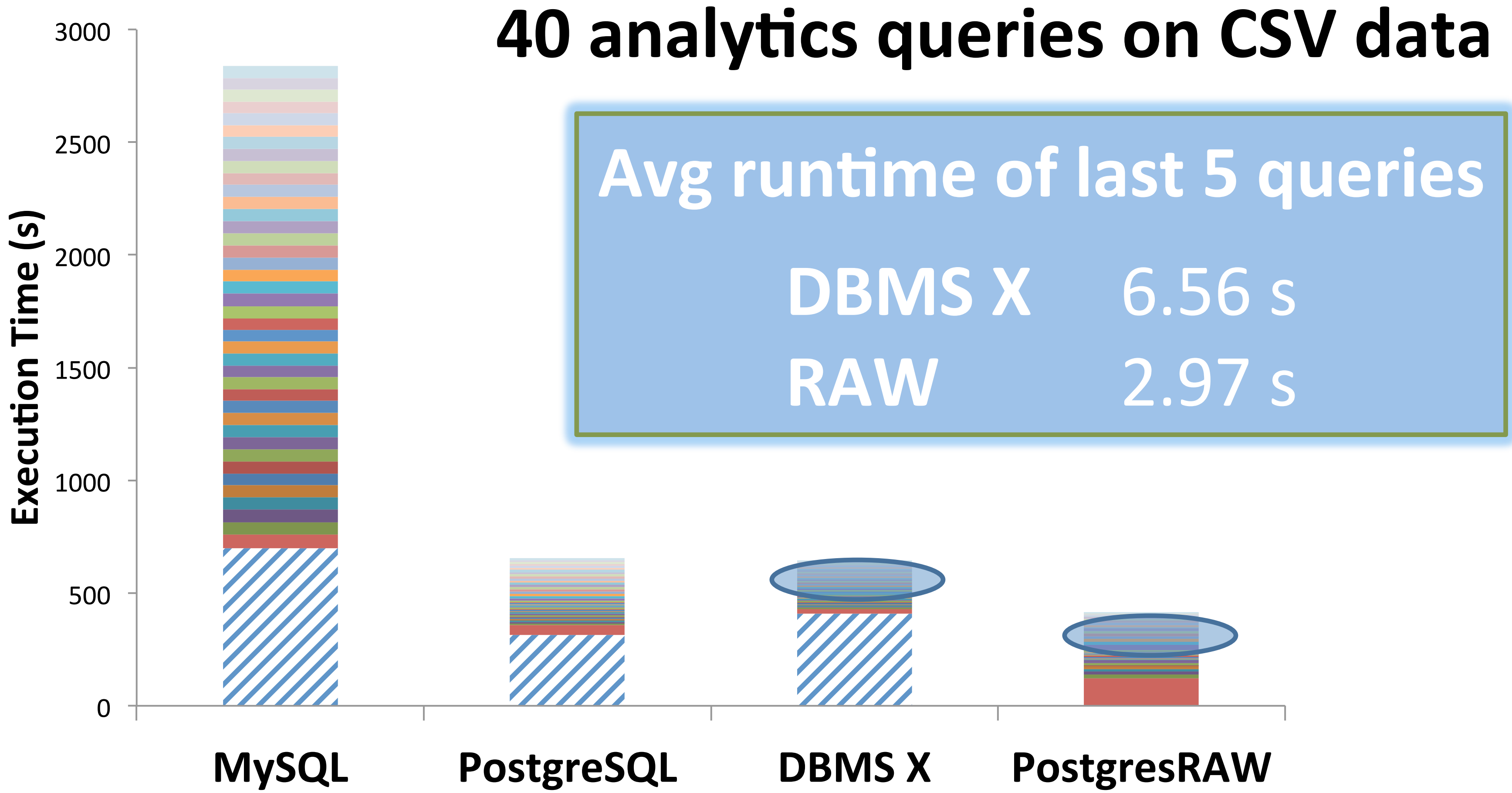
data-to-query time = 0

PostgreSQL + NoDB = PostgresRaw



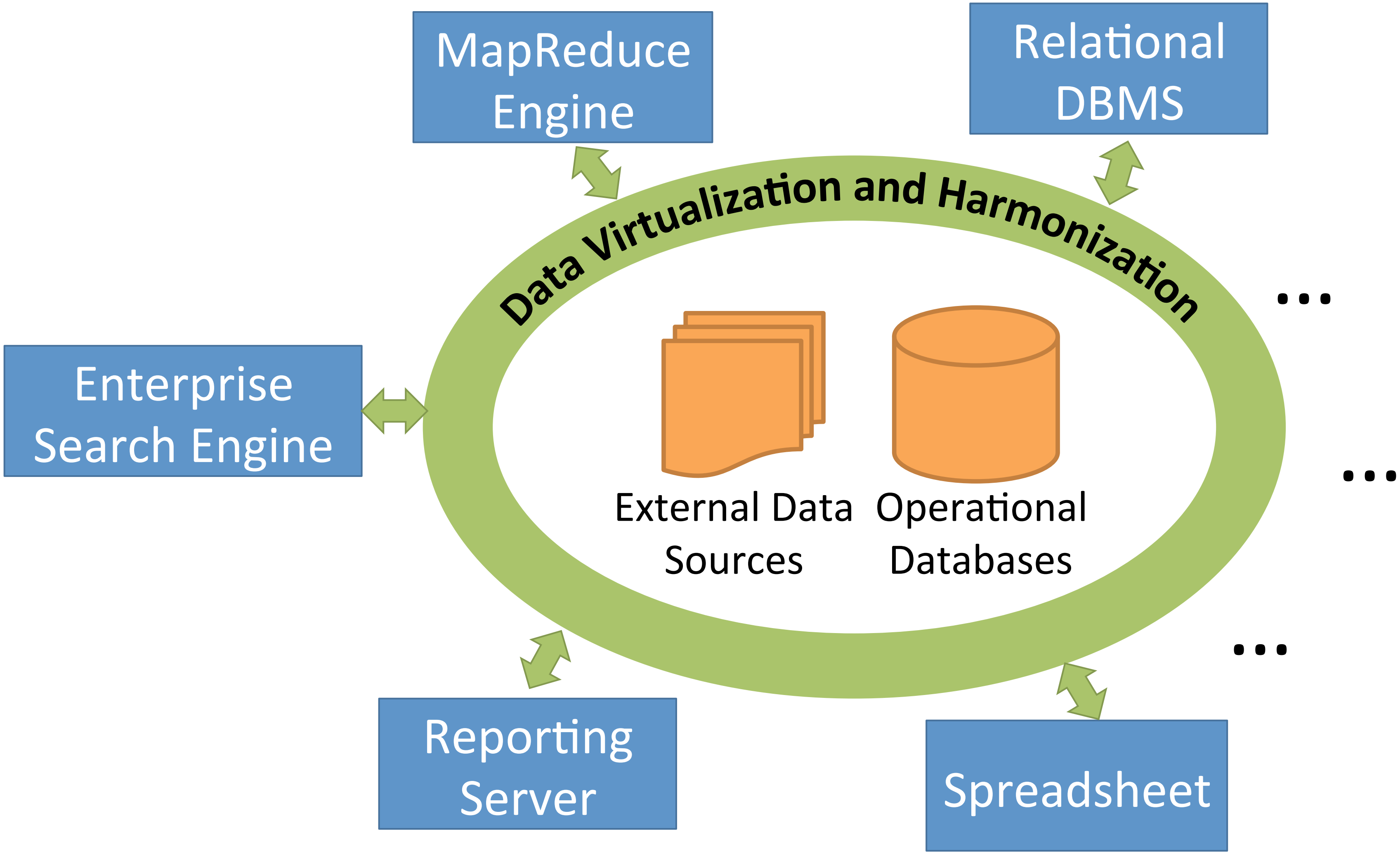
comparable/competitive performance

PostgreSQL + NoDB = PostgresRaw



NoDB gets *progressively* faster

now: run queries to create database



ViDa: in-situ query engine

making a difference

- **Solve the domain scientist's problems**
  - **not ours**
- **One* solution does not fit all**
  - ***query language, data model, data type, index**
- **Go back to the lab and apply findings**
  - **then write computer science papers**

- **Build *multiple* bridges to sciences**