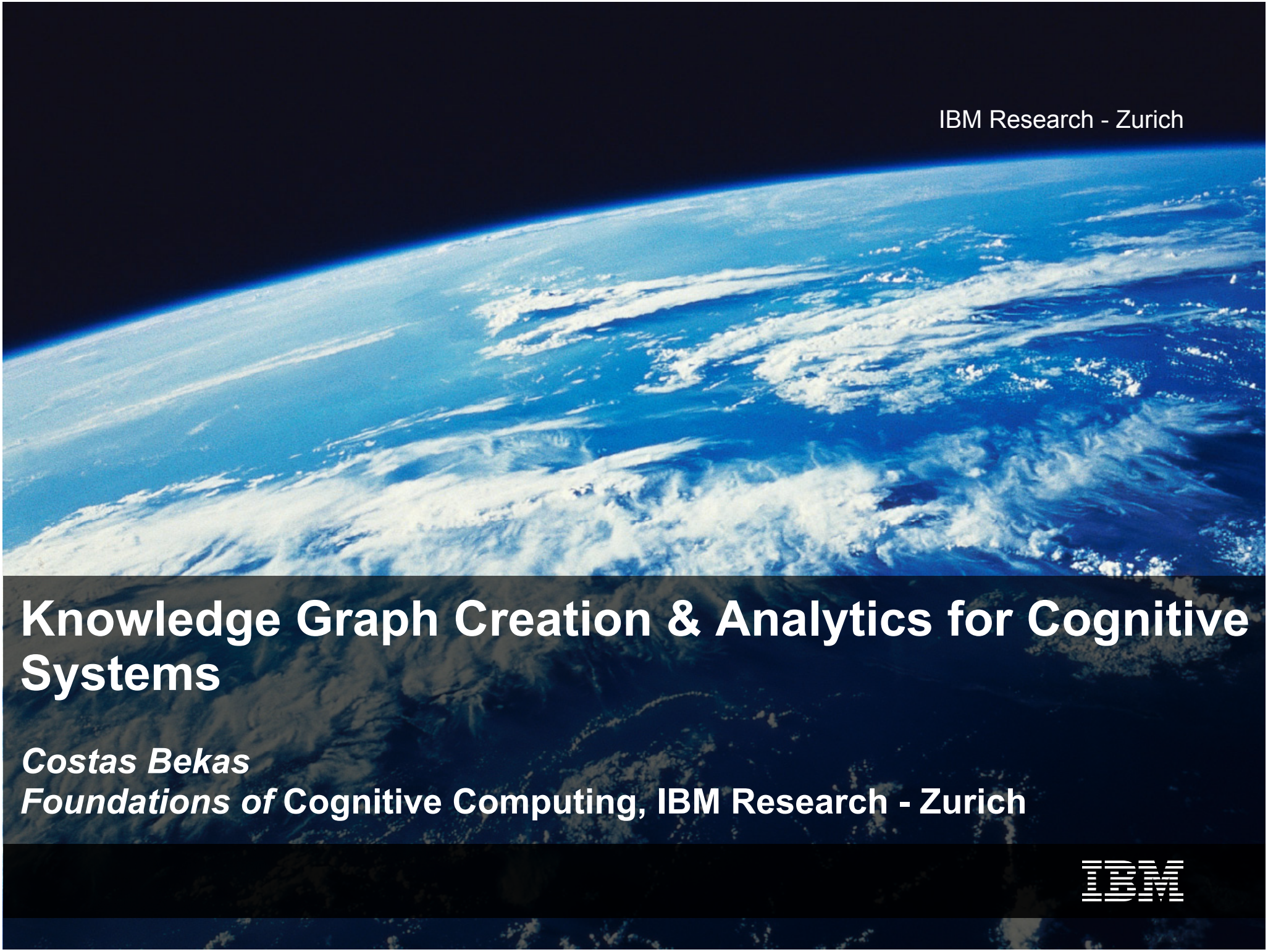
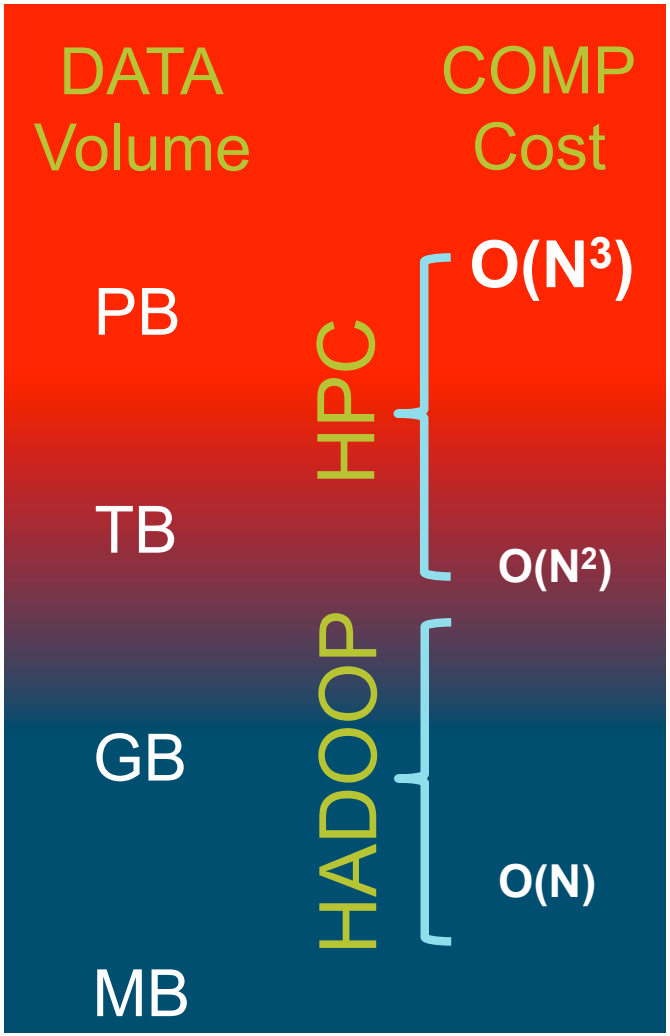


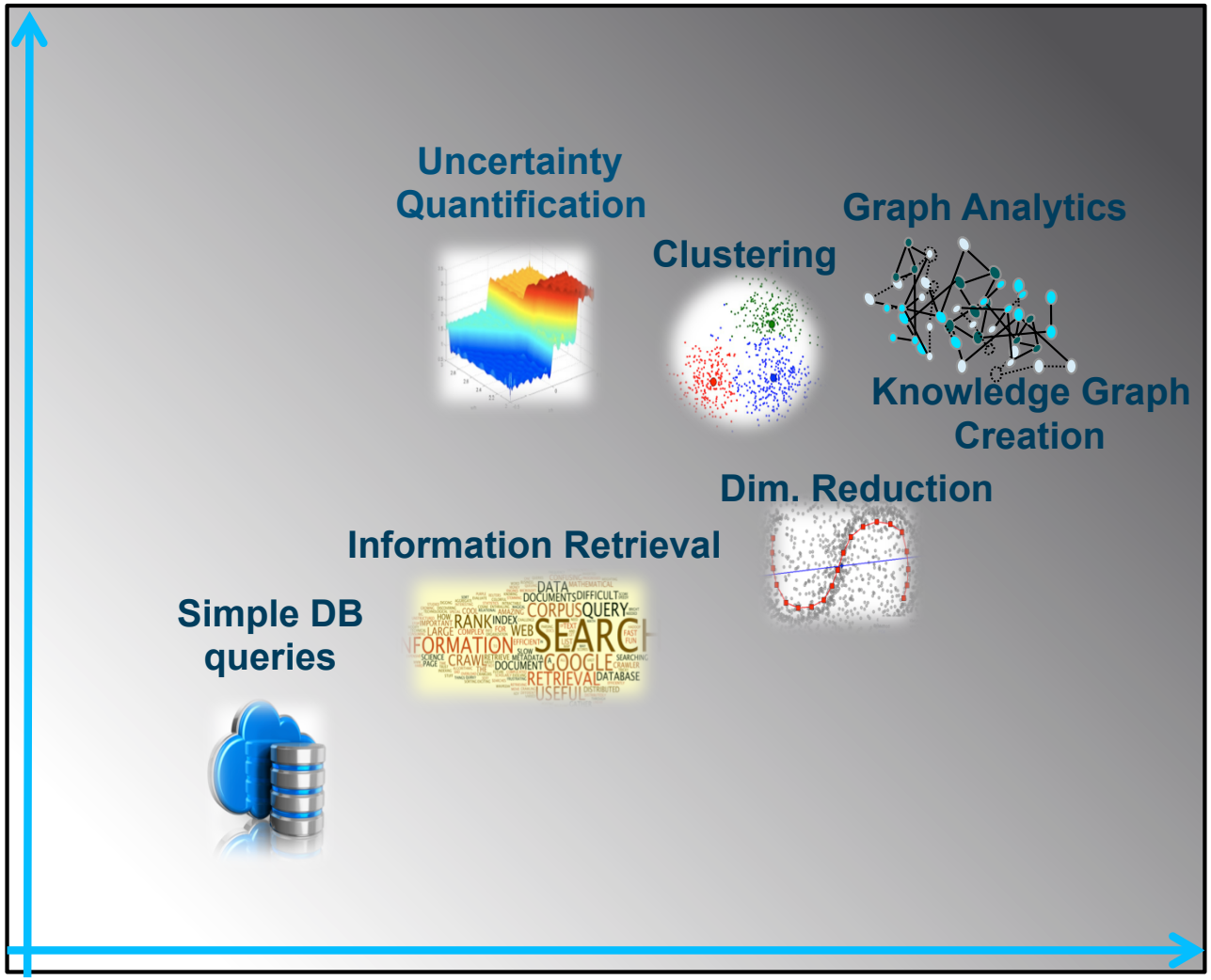
Knowledge Graph Creation & Analytics for Cognitive Systems

Costas Bekas
***Foundations of* Cognitive Computing, IBM Research - Zurich**

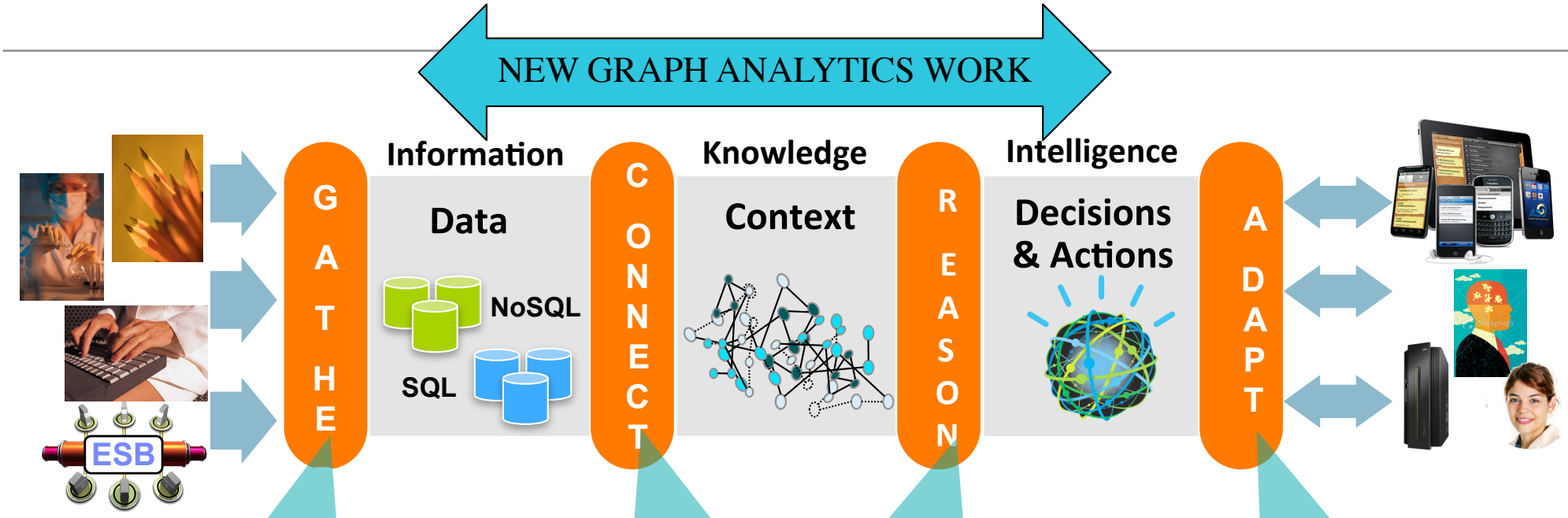
IBM

Data & Computation trends towards Cognitive Computing





Data driven knowledge discovery pipeline



Gather

Collect all relevant data from a variety of sources: **Publications, RSS, APIs, DBs, etc**

Connect

Extract features and build context using multiple **diverse data sources, new sources added at run-time: User defined**

Reason

Analyze data in context to uncover hidden information and find new relationships.

Analytics both add to context **via metadata extraction, and use context to broader information exploited**

Adapt

Compose recommended interactions, use context to deliver to point of action.

Research directions and projects engagement

Focus on Knowledge Graph creation and analytics

- Representation of knowledge. Basic foundation of Cognitive Computing

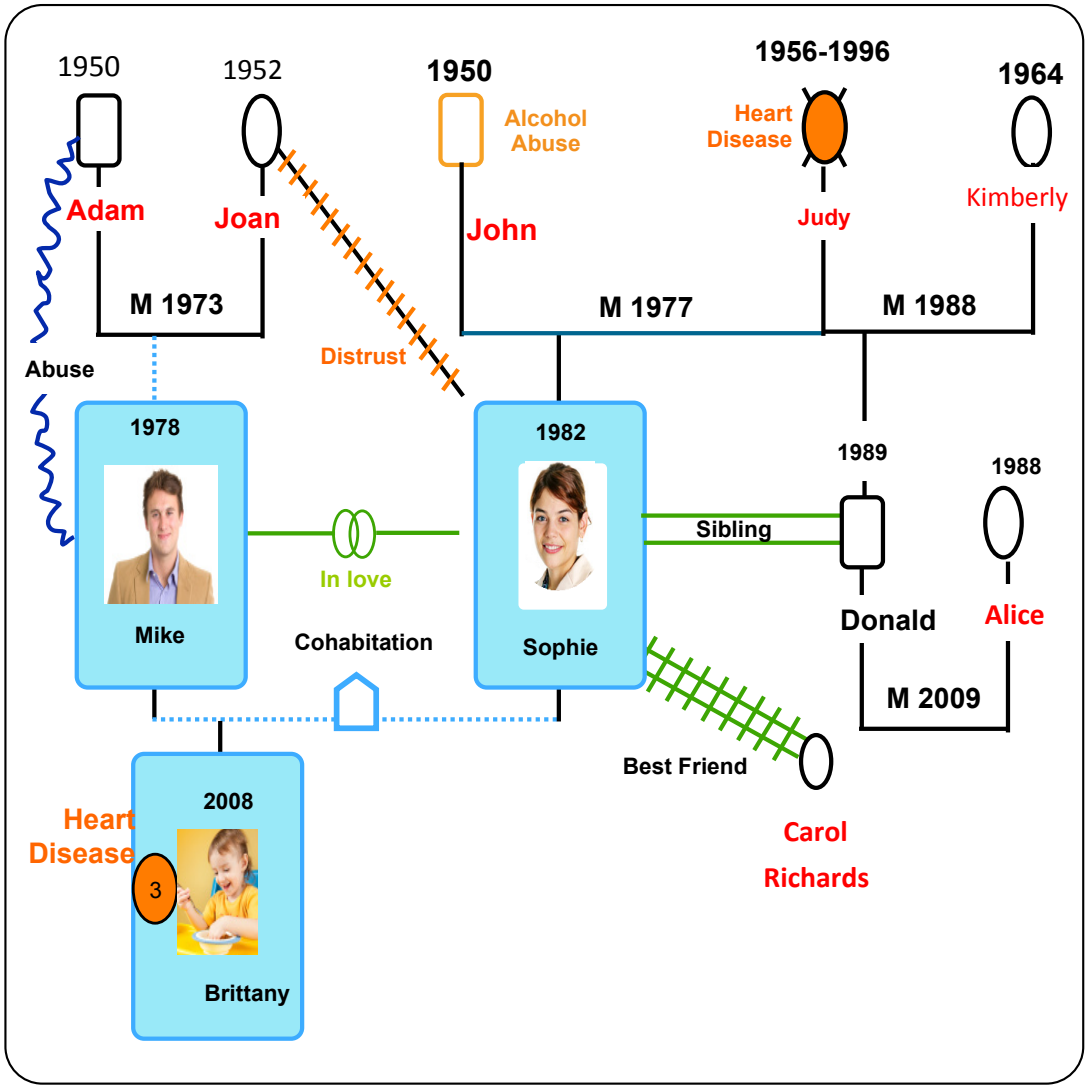
Strategic Projects: Support roadmap for competitive edge

- Materials Analytics: Focus on materials knowledge graphs
  - MARVEL: SWISS NSF. Data driven material discovery
  - Direct client projects and Watson engagement
- Acceleration and Cloud deployment of graph analytics
  - Nanostreams (EU FP7): Focus on accelerators
  - HPC Java (EU FP7), XDATA (DARPA): Focus on Java and algorithms

Novel graph analytics tools

- **Node importance: Sub-graph node centralities**
  - Previous art: $O(N^3)$ cost. Graphs in the millions of nodes require Exascale. Not possible on the Cloud.
  - Our method: close to O(N). Runs on the Cloud and HPC. Cuts time down to minutes from several hours (or days)
- **Graph Comparisons: Spectrograms**
  - Completely new method to compare graphs
  - Standard methods: Combinatorial heuristics, limited to very small graphs
  - Our method: close to O(N) cost, runs on the Cloud & HPC

Knowledge Graphs: Hold data + relationships = Knowledge



Data examples:

- Birthdays, death dates, family lineage, medical events

Relationship examples:

- Mike is in love with Sophie, they live together
- Carol is the best friend of Sophie

Advanced relationships examples:

- Sophie's mother died of heart disease. Sophie's daughter suffers of heart disease

Query Examples:

- Why does Brittany has heart disease? Is there a chance that she is abused? Who should we call if Sophie goes missing?

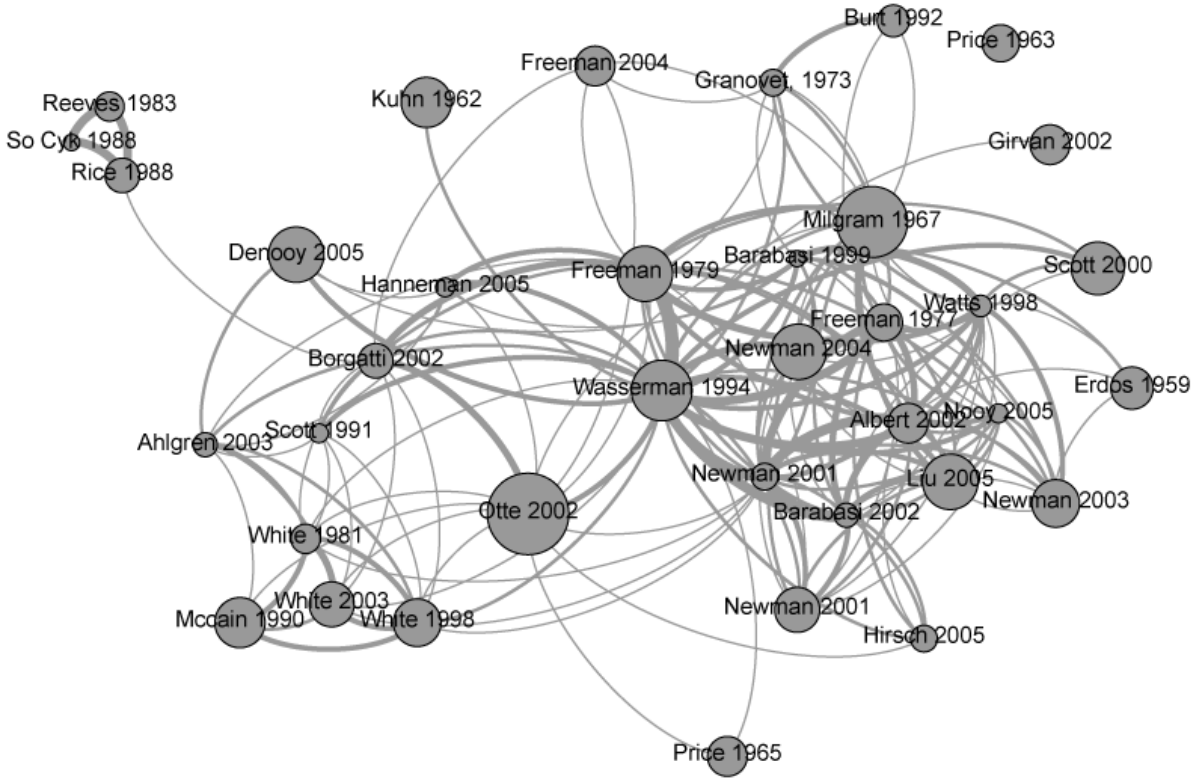
Deeper dive: Knowledge graphs for materials

Citation graphs on documents: papers/patents

- Nodes are documents
- Two nodes are connected (there exists and edge between them) if there is citation pointer between the documents

What kind of analytics can we do with our tools?

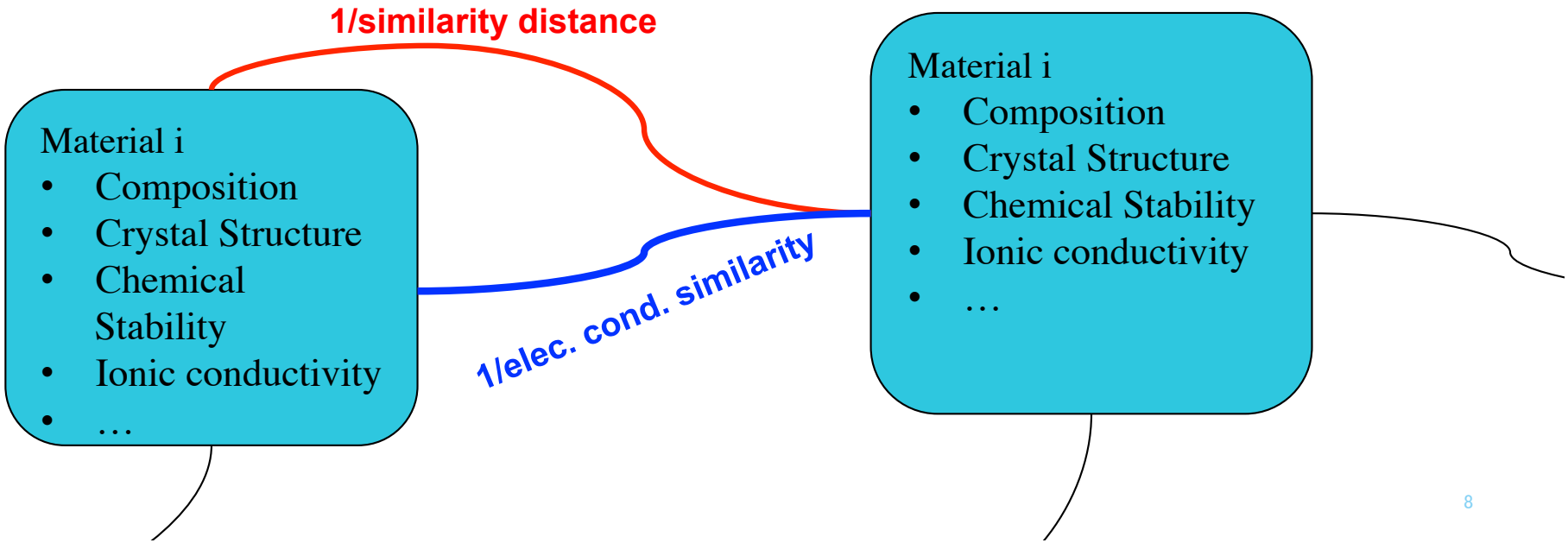
- Find influential patents (obviously not simply the most cited ones)
- Find groups of similar patents (wrt impact/ influence)
- Find similar patent categories



Knowledge graph creation example: focus on similarities
An example from materials science

Connections (edges) have weights to indicate degree of similarity

- Multiple similarities can be expressed at the same time and can be chosen on the fly
- There is a learning phase for knowledge graph creation. Similarities and connectivity needs to represent ground truth and empirical knowledge
- This calibration phase requires the systematic comparison of graphs



A Metallurgical Knowledge Graph Data model

We have various node types

- Alloy node type: according to standard numbering/classification
  - Node properties: composition, form, etc
- Basic doping element type
- Processing type
  - Node properties: basic forming steps and order
- Document node type:
  - Node properties: extracted text based on industry provided ontologies

The Knowledge Graph Data model

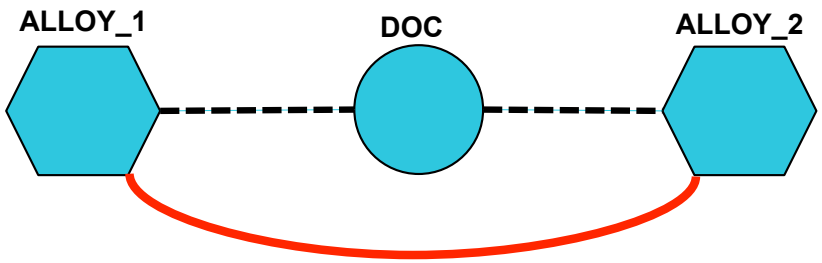
Let us consider 2 alloys: Alloy_1, Alloy_2

- We conduct text extraction from a set of documents and get the document type nodes

Consider the following cases:

1. A document node refers to both alloys:

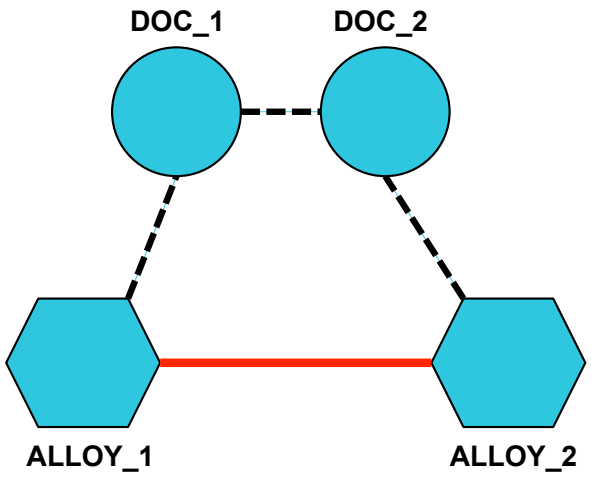
   Then the alloys nodes are connected



2. Two documents refer to alloys separately

   but the documents are linked by citation:

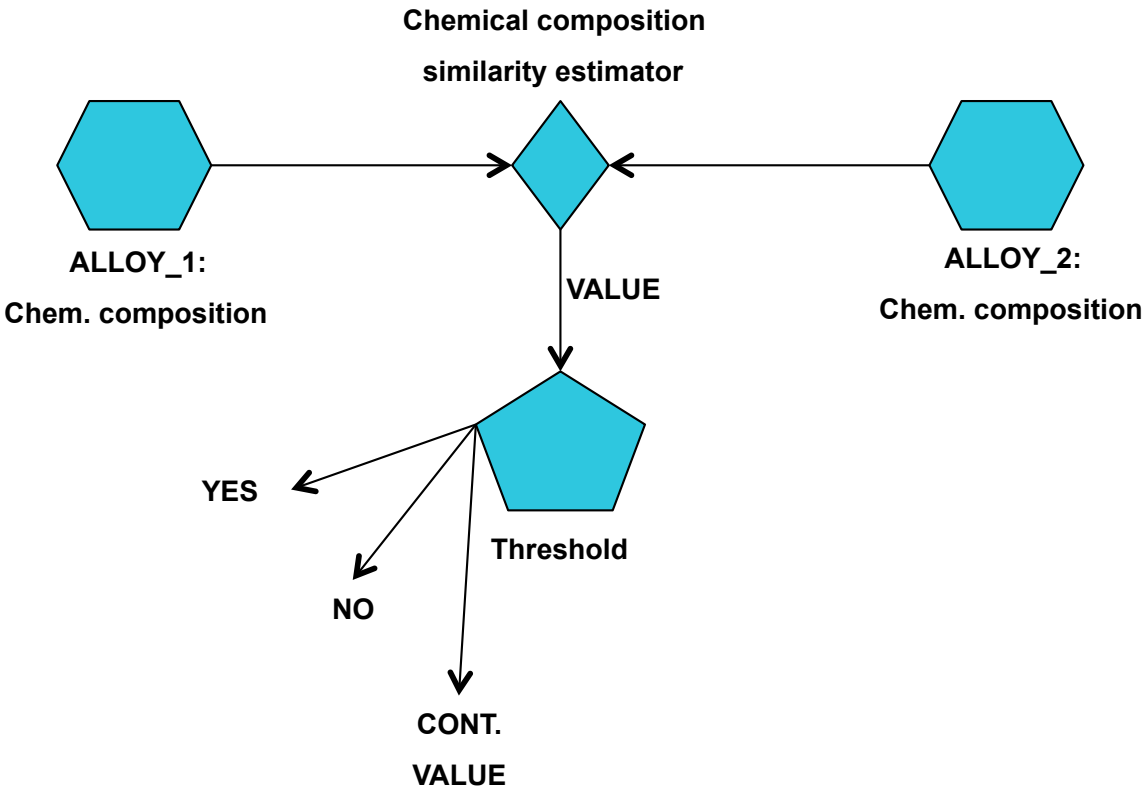
   Then the alloy nodes are connected



The Knowledge Graph Data model

Let us consider 2 alloys: Alloy_1, Alloy_2

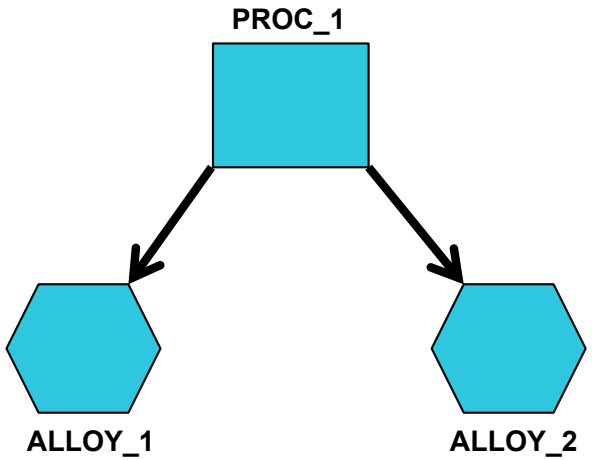
- Can we extract similarities beyond extracted text?



The Knowledge Graph Data model

Let us consider 2 alloys: Alloy_1, Alloy_2 and a Process Proc_1

- We conduct text extraction from a set of documents and get the document type nodes. The document nodes link process Proc_1 to each alloy separately:



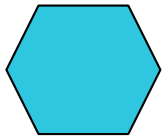
Query: “Find all alloys for which Proc_1 is used and certain properties need to hold for the alloys”

Action on graph:

- Start from the node Proc_1 and visit its neighbors
- Those nodes you find that are of the alloy type that fulfill the user defined criteria are your answers

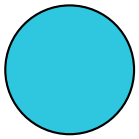
The Knowledge Graph Data model

These processes create a complex Knowledge Graph that captures all the knowledge in the text, in the practical experience & from physics/chemistry principles.

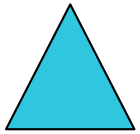


ALLOY NODE TYPE

PROCESS NODE TYPE

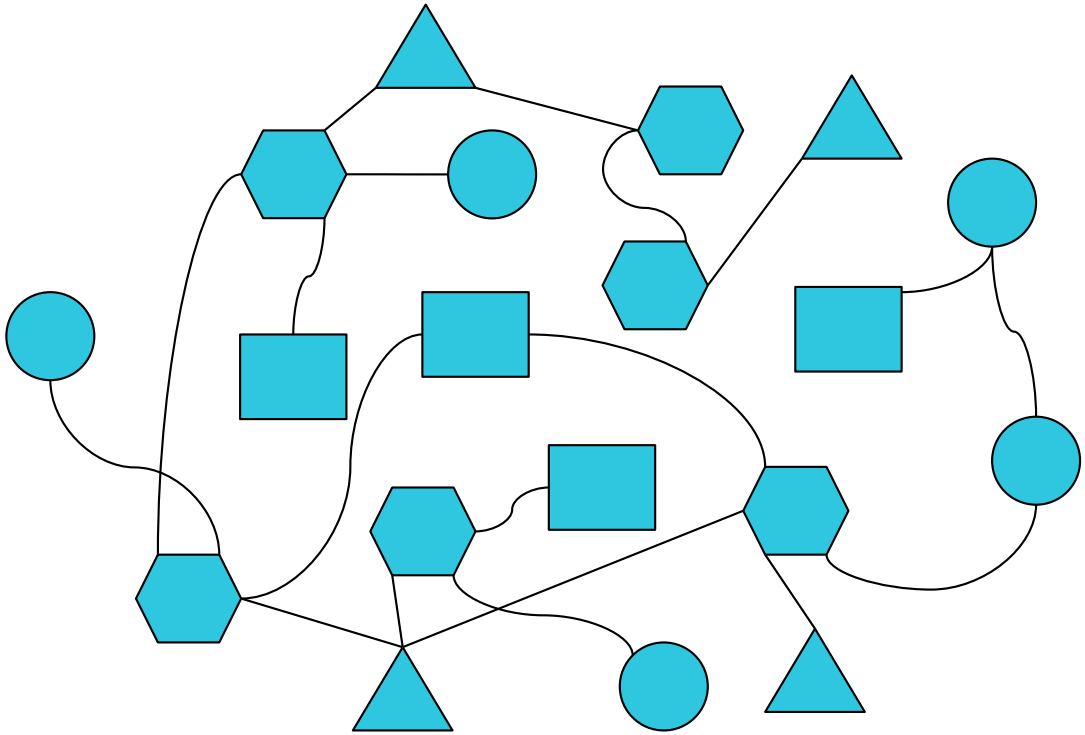


DOCUMENT NODE TYPE



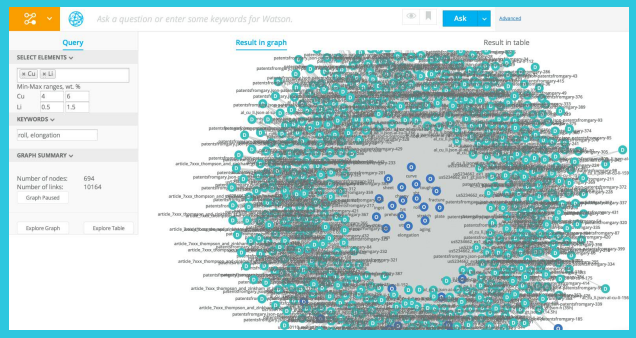
ELEMENT NODE TYPE

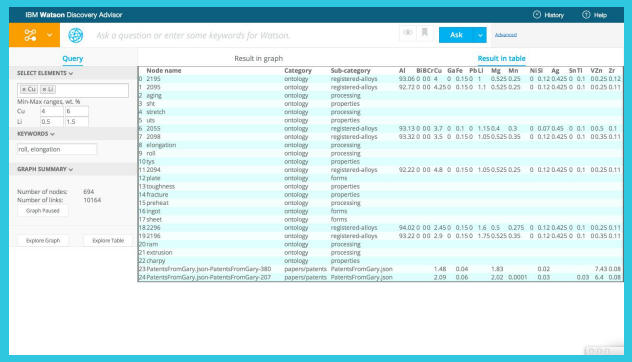
KNOWLEDGE GRAPH



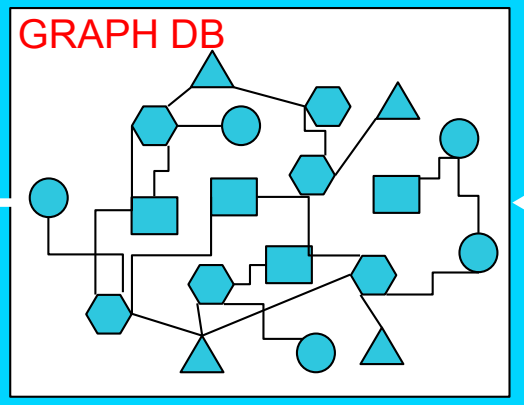
System Architecture

WDA front end system



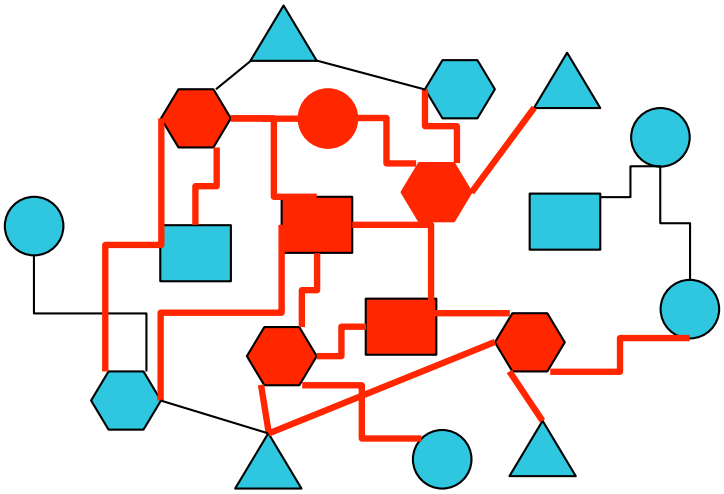


Back end system

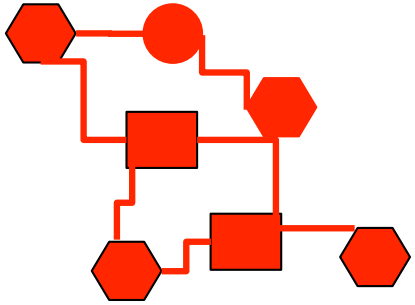


Query work flow

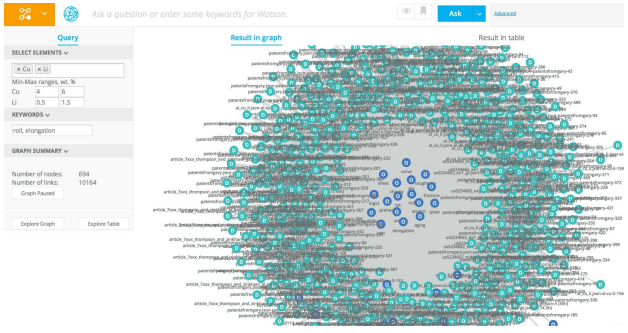
TRANSLATE QUERY TO SUBGRAPH SELECTION



COMPUTE RANK (IMPORTANCE) OF NODES: NODE CENTRALITIES

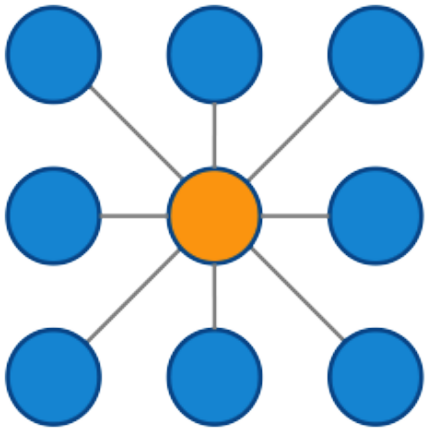


VISUALIZE AND EXPLORE



Node importance: Sub-graph node centralities

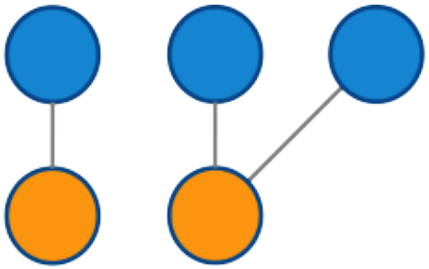
The subgraph centrality measures the participation of each node in all subgraphs in a network, where smaller subgraphs (with same start and end point) carry more weight than larger ones.



Star graph, center node has largest subgraph centrality value.

Subgraphs of size 2

Subgraphs of size 4



- In a star graph the center participates in all (8) subgraphs of size 2 (a line)
- Each other nodes participate in all (8) subgraphs of size 4, where the center participates in 64 subgraphs of size 4

Computing graph centralities

Counting the number of paths in a graph boils down to simple algebra with the adjacency matrix:

$$C = I + A + A^2/2! + A^3/3! + A^4/4! + \ldots$$

- The power k counts how many paths of length k there are between nodes i-j
- We would like to penalize (weight) very long paths over shorter ones. Thus, we use the factorial scaling (Estrada index)
- Thus the diagonal of matrix C is exactly what we need.
- But: basic matrix algebra shows that C=expm(A), i.e. the matrix exponential
- Problem: This is a $O(N^3)$ problem using standard techniques. Exascale needed for graphs of size 1M
- We developed an O(N) method to compute graph centralities. (patent pending)

How do we compute node centralities at O(N) cost

Remember: we are looking for the diagonal of the matrix exponential function

Key point: we do not need all of the matrix function. Just selected elements of it!

$$\mathcal{D}_s(P(A)) = [\sum_{i=1}^{s}(z_i \odot P(A, z_i))] \oslash [\sum_{i=1}^{s} z_i \odot z_i]$$

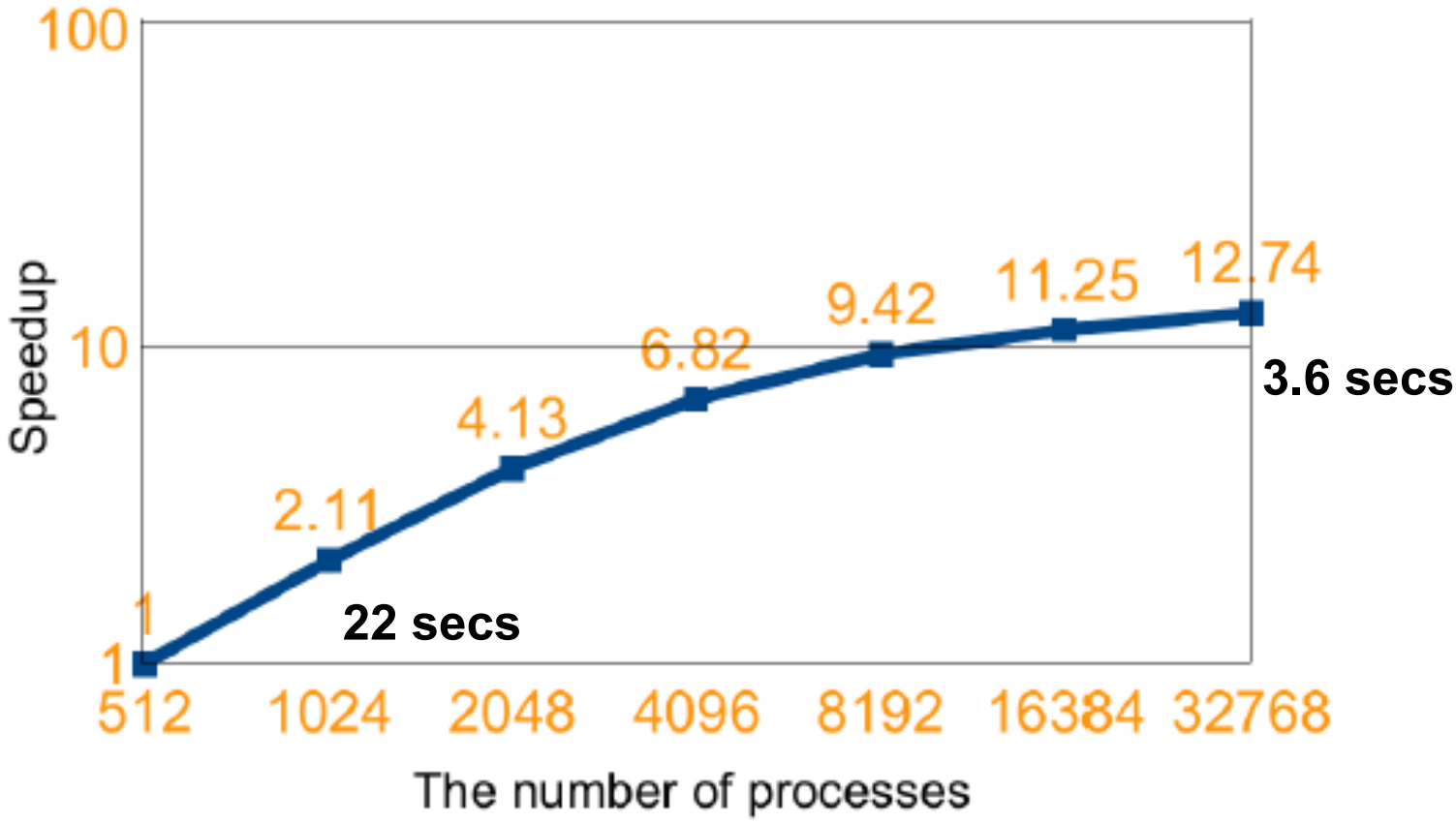
Solution:

- ✓ Stochastic estimation: Use $s<<n$ carefully designed vectors v_i and estimate
  - o where $\otimes$ is Hadamard (entry-wise) multiplication and $\oslash$ is Hadamard division
- ✓ P(A,v_i) is approximating the multiplication of matrix exponential with a vector (Lanczos)
- ✓ Since s<<n and P(A,v_i) costs O(N) (Lanczos).

Total cost: O(sn). Memory: O(N)

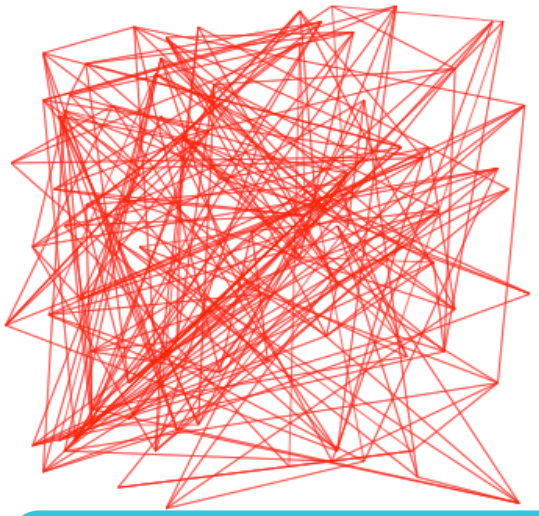
Scalability of node importance calculator: Speedup

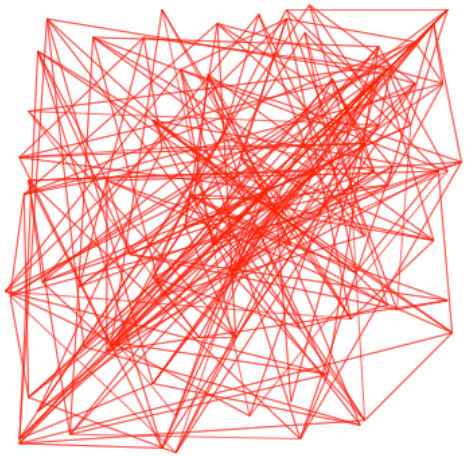
Scaling to HPC resources



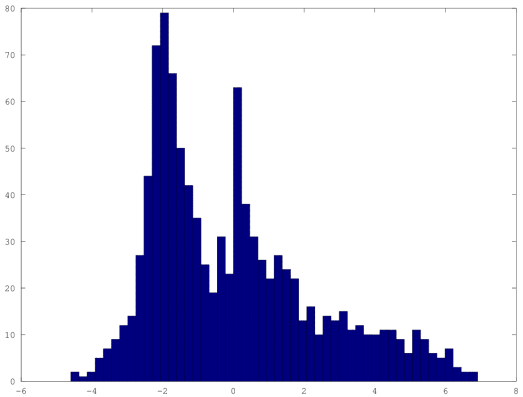
Road network of Europe. 50M nodes, 150M edges.

Graph Similarities

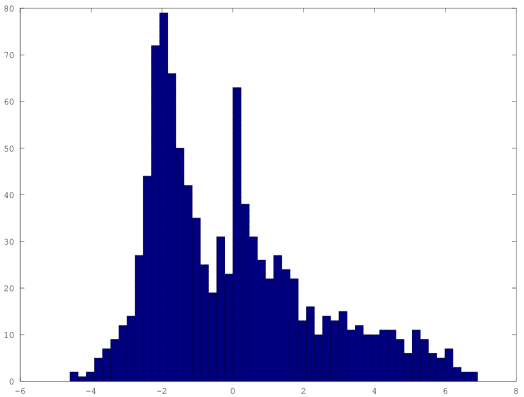




GRAPH SPECTROGRAMS



1D VECTOR CORR.



O(N) Method for Graph Spectrograms

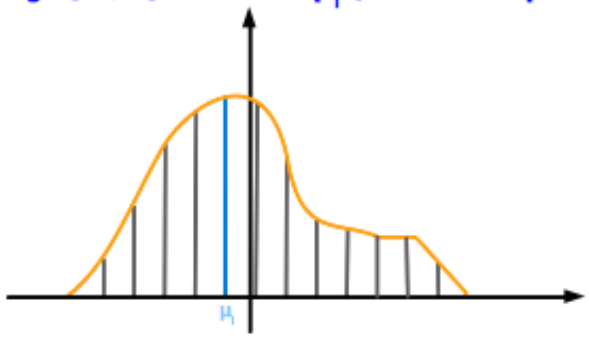
METHOD

1. SCALE AND SHIFT EIGENVALUES OF A

For any real symmetric sparse matrix $A \in \mathbb{R}^n \times \mathbb{R}^n$ we start by **estimating the λ_{min} and λ_{max} eigenvalue** of the matrix in order to shift and scale A to have eigenvalues in the interval [-1, 1]:

2. PARTITION IN B BINS

Divide the range [-1, 1] in **b bins μ_i (inflection points)**:

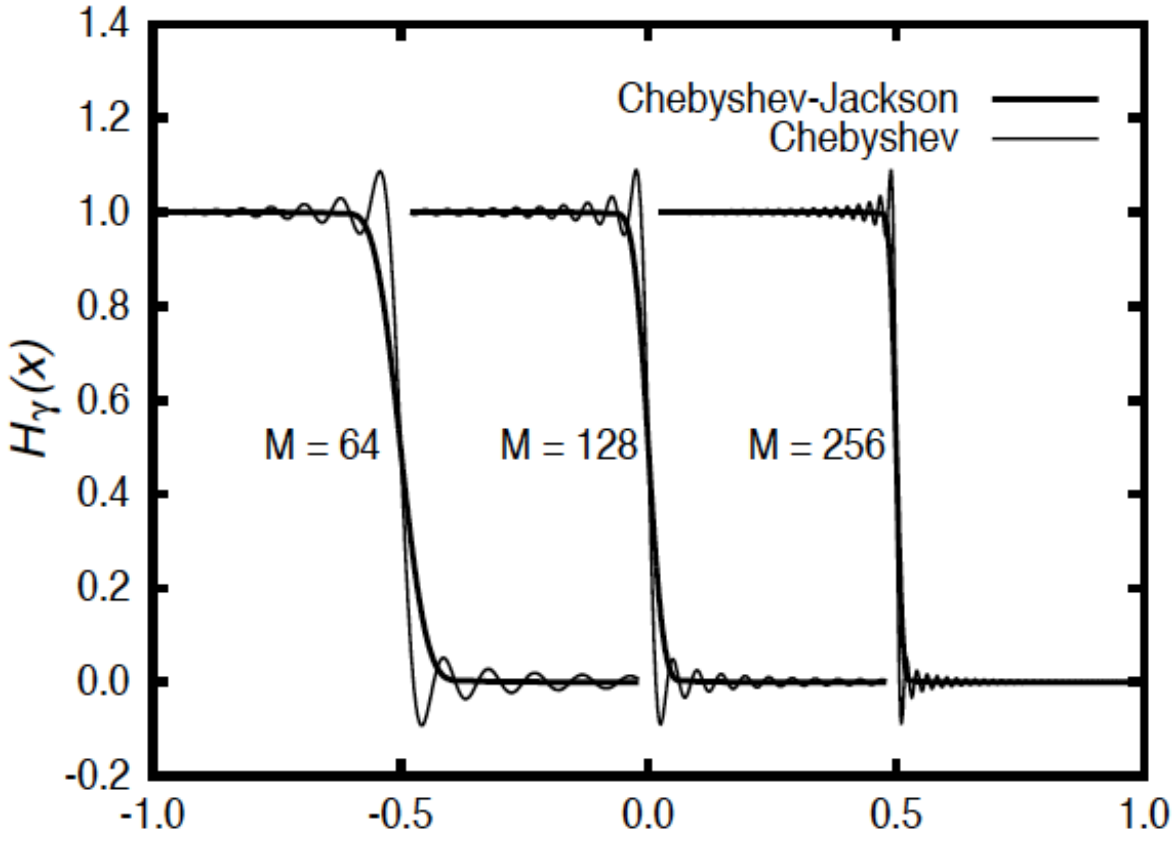


- Then P(A,v)'s is nothing but filtering functions that map all eigenvalues less than the inflection points to 1 and the rest to 0.

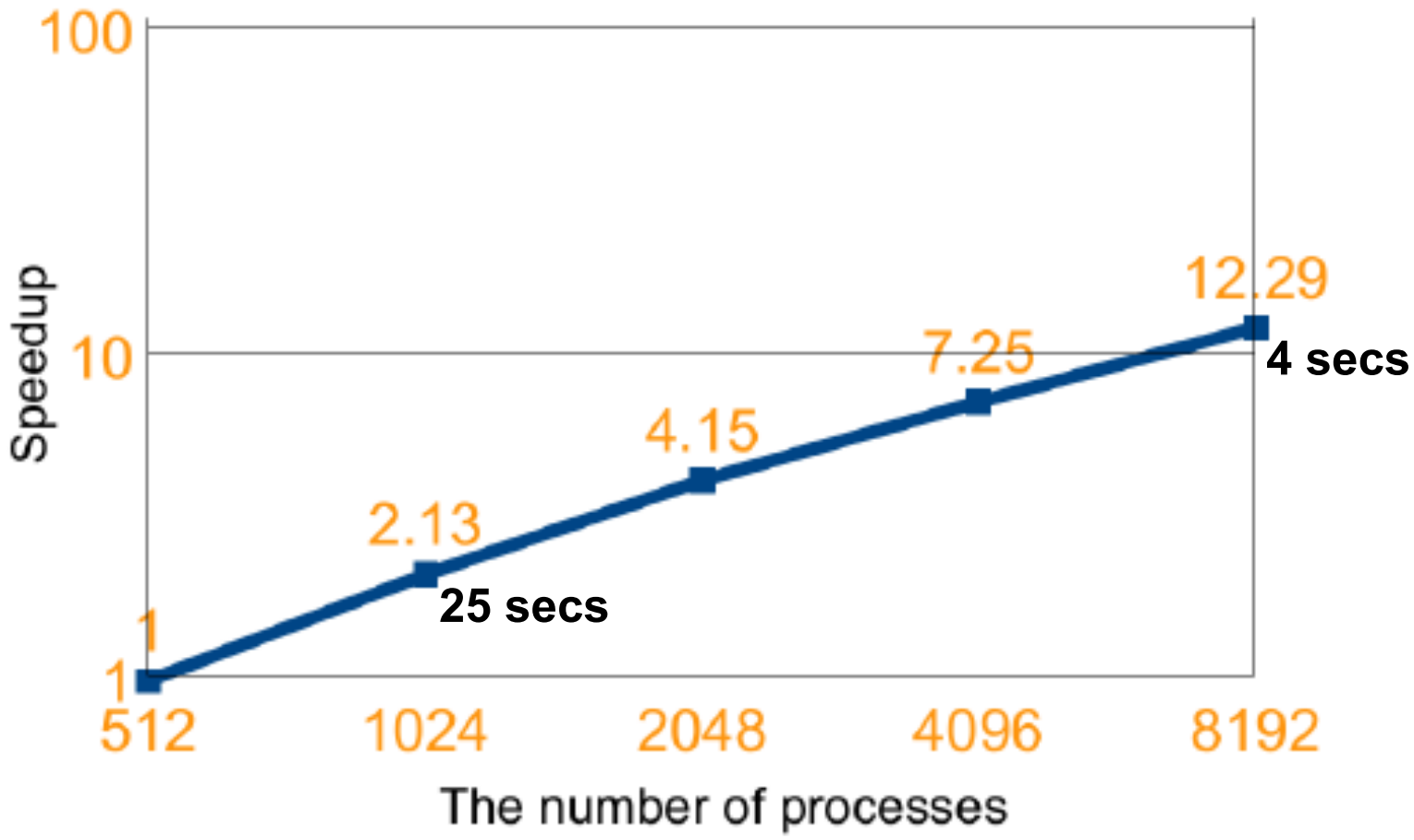
- Thus the diagonal of this function is nothing but it's trace…which counts how many eigenvalues there are less than the inflection points…then it is a matter of simple subtraction to get the spectrogram.

O(N) Method for Graph Spectrograms

Applying the filters in the diagonal stochastic estimator, we count how many eigenvalues are less than -0.5, 0 and 0.5. Thus it is trivial to get how many are in the intervals [-0.5, 0] and [0, 0.5]



Scalability of graph spectrogram calculator: Speedup



Road network of Europe. 50M nodes, 150M edges.

Conclusions

We focus on knowledge graph analytics and develop:

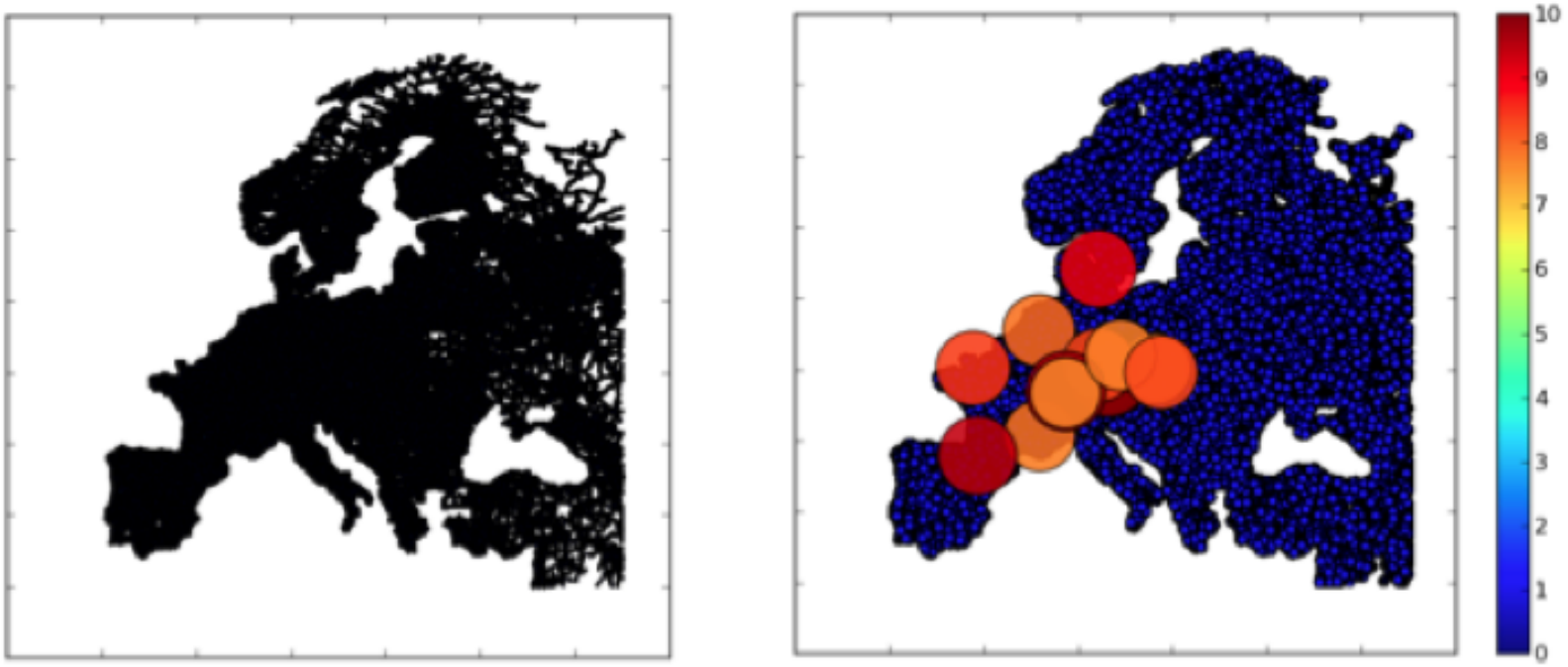
- O(N) cost methods that allow new powerful analytics
- Cloud deployable and scalable to HPC resources as well

  - Node importance: Sub-graph node centralities
  - Graph Comparisons: Spectrograms

We employ this work on key projects that drive impact and allow us to develop new functionality

This is part of our roadmap to systematically attack the complexity of advanced ML and Cognitive algorithms

Analyzing complex networks



Using single server resources:

Road network of Europe. 50M nodes, 150M edges.

- 14 minutes on 16 cores
- Real traffic monitoring at very large scale is thus now possible