NOA-AID:

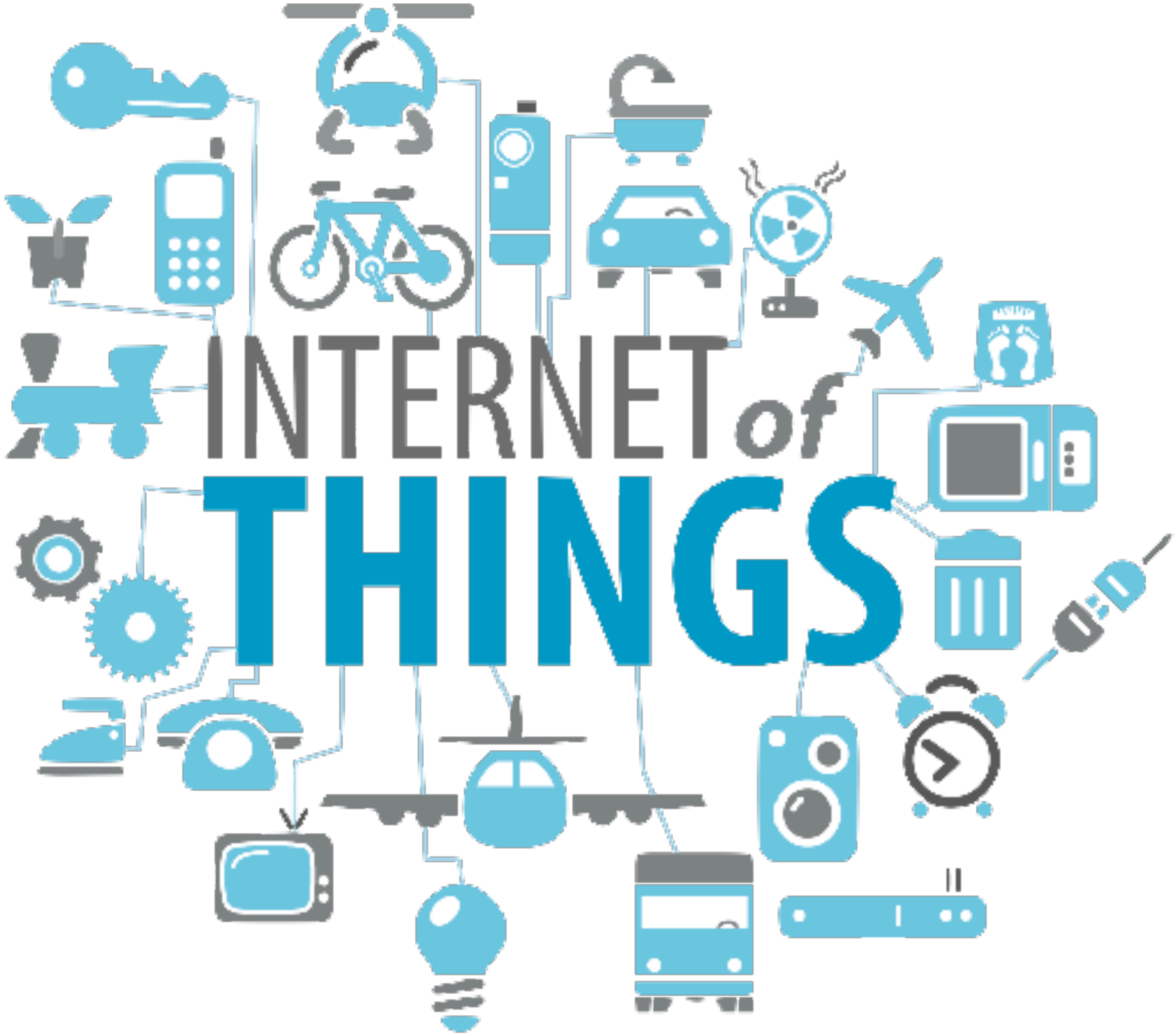
NETWORK OVERLAYS FOR ADAPTIVE INFORMATION AGGREGATION, INDEXING AND DISCOVERY AT THE EDGE

Patrizio Dazzi[a] *and Matteo Mordacchini*[b]

[a]*Researcher @ ISTI-CNR, Italy, Pisa*
[b]*Researcher @ IIT-CNR, Italy, Pisa*

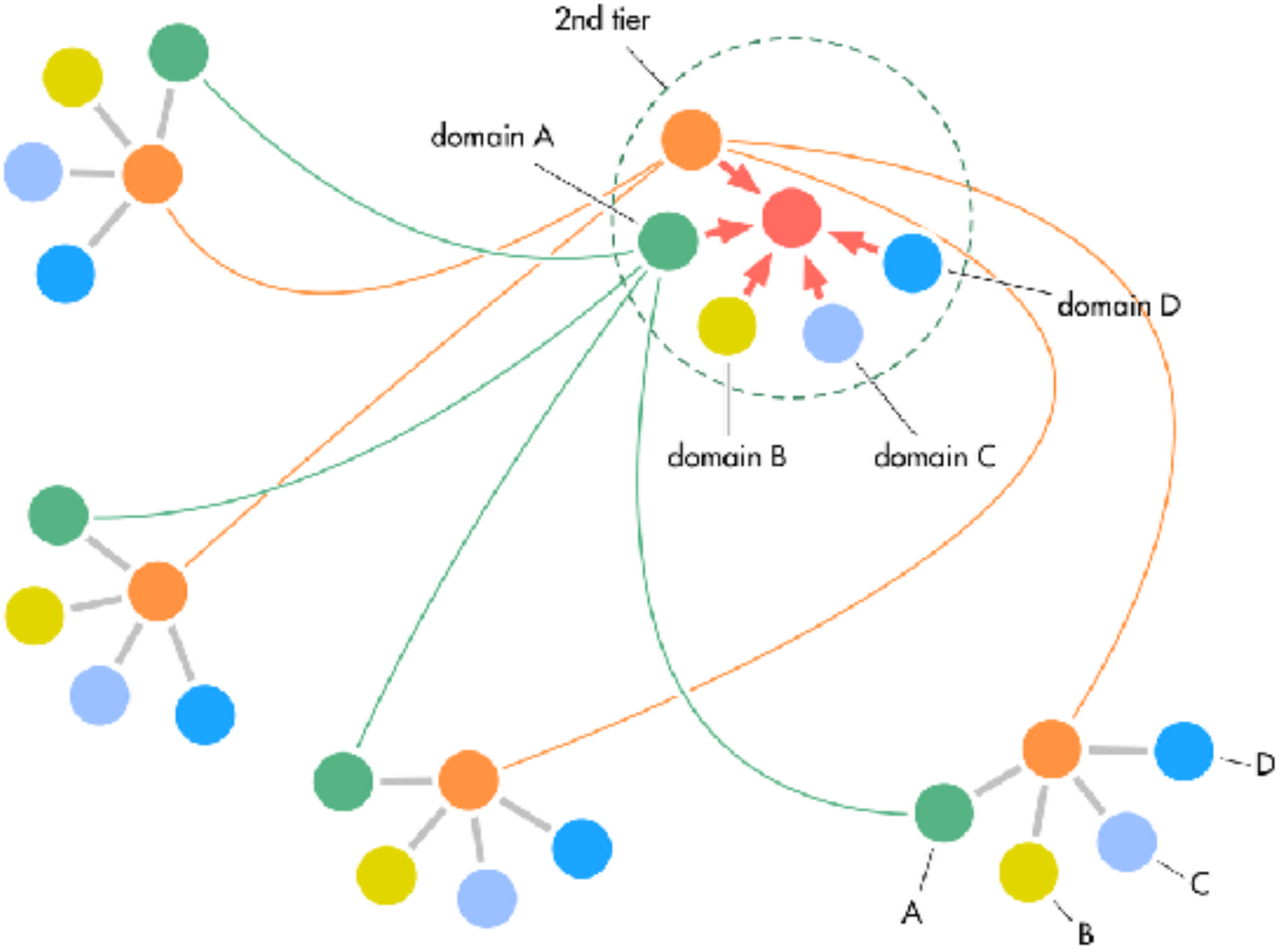
LET’S START

DATA, DATA, DATA AND… DATA IS ~~BIG~~ EVERYWHERE



- Pervasive computing environments
- Devices produce, transmit observe a huge amount of data in the form of **streams**
- Processing is of paramount importance
  - detect faults
  - issue alerts
  - trigger operations
- Information processing and management achieved as a **cooperative process**

COOPERATIVE PROCESSING: REQUIREMENTS



- **Efficient** and **effective** communication enabling information
  - gathering
  - exchange
  - **indexing**
  - **querying**

Expressiveness and Effectiveness as key aspects

- *Can ease the task and limit the overhead*
- *Give a support to highly dynamic environments*

HOW-TO EFFICIENTLY INDEX AND QUERY ?

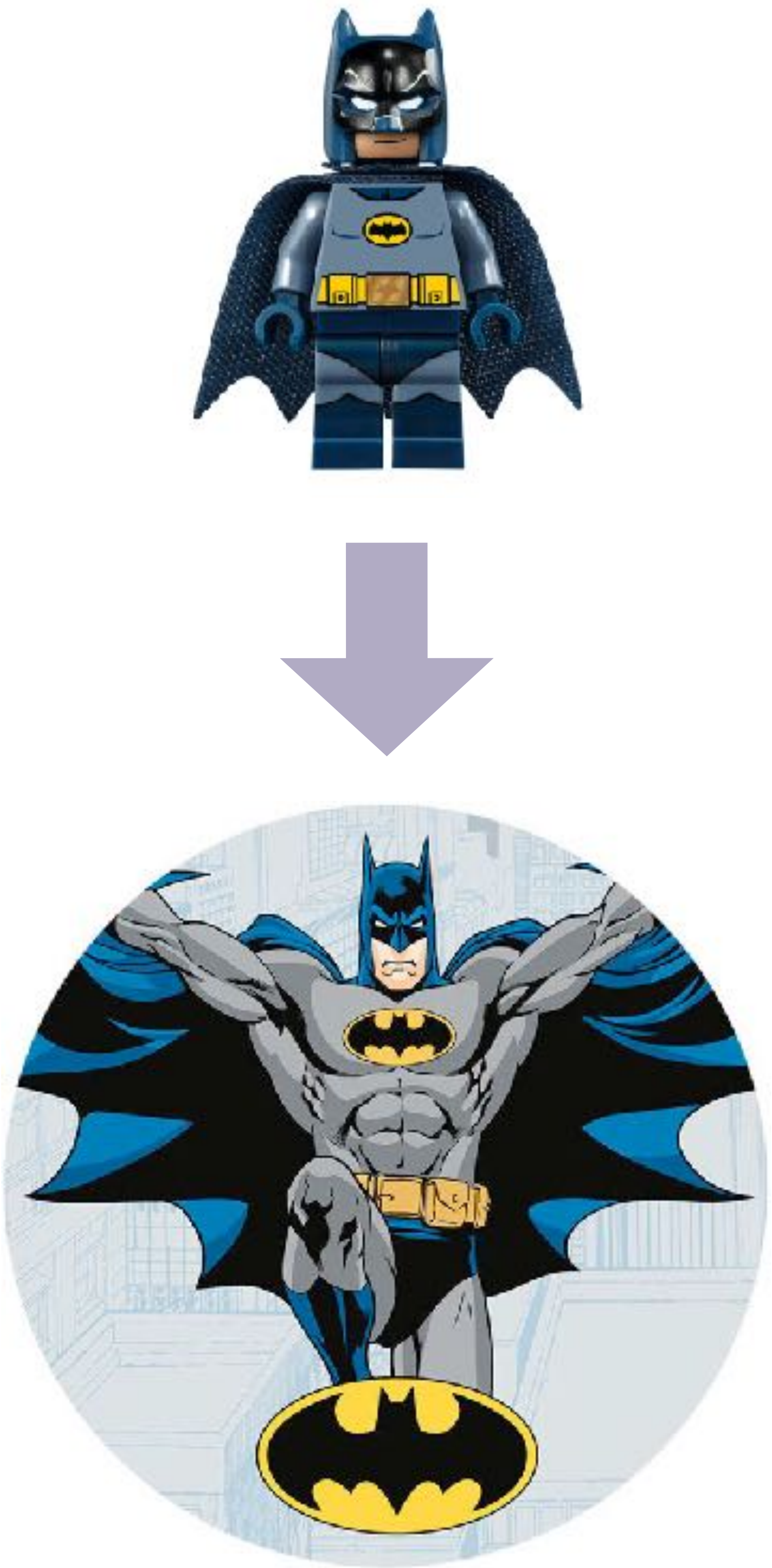
PROBLEM: ARE YOU SURE YOU KNOW WHAT YOU ARE LOOKING FOR ?

- Sometimes, in dynamic, heterogeneous environments managing **streams** it is not that trivial to have all the information for define a query
- Unfortunately, the heterogeneous nature of distributed devices and data dynamicity of the information makes the task of query formulation very complex
  - I.e. When an information is defined as the combination of many different attributes, it could be not that easy to identify the most relevant and discriminating features

I want a Dark Knight

with a black cape

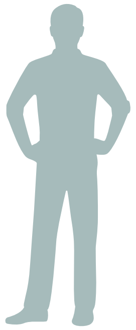
equipped with amazing technology



A GOOD ALTERNATIVE: THE SIMULACRUM

- Archetype of the information sought
- Identify the desired set of information relieving the requester from specifying complex queries
- Discovery system needs to be organised accordingly
    - Support approximated searches
    - Without causing a flood of data

SO... WHAT ?



NOA-AID

- A solution targeting **stream processing devices** easing the information indexing and retrieval
  - Devices observe the content of its managed **stream**
- A **Profile** represents data flowed through a **stream**
  - As a consequence the profile represent the device associated to that stream
  - Two ways to represent data observed - the profile
    - Term vectors
    - Adjacency Matrices }

More will follow…

  - A flexible query-by-example (the simulacrum) discovery mechanism

NOA-AID

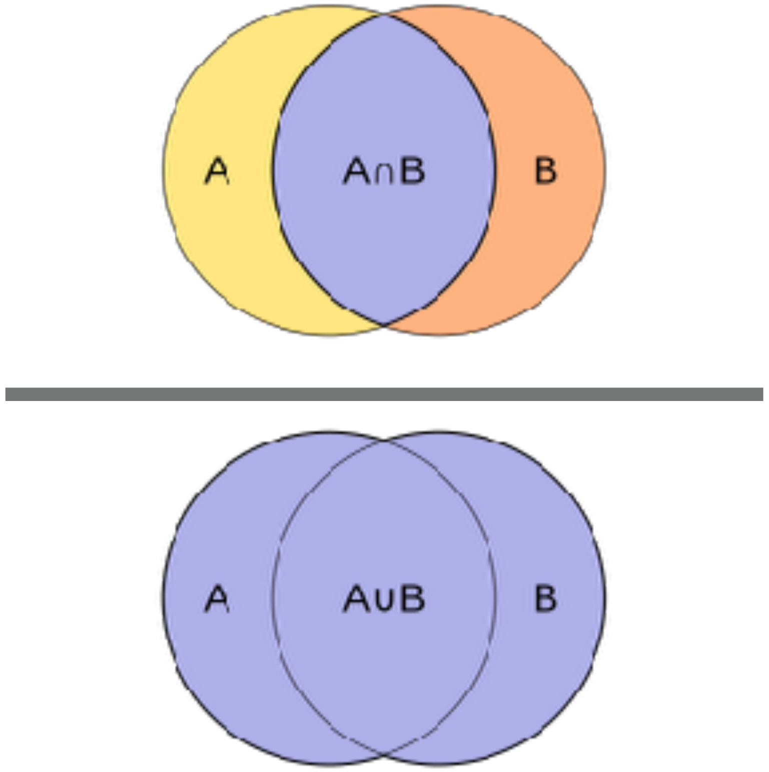
- Devices exchange their profiles in point-to-point handshakes
  - discover mutual similarities
  - gather in communities of similar devices
  - each community is represented by a leading device
- Each leading device will be the representer and entry point for that community
- NOA-AID provides means for finding these community entry point by providing a Simulacrum of the information sought

WE

HOW ~~THEY~~ DO IT ?

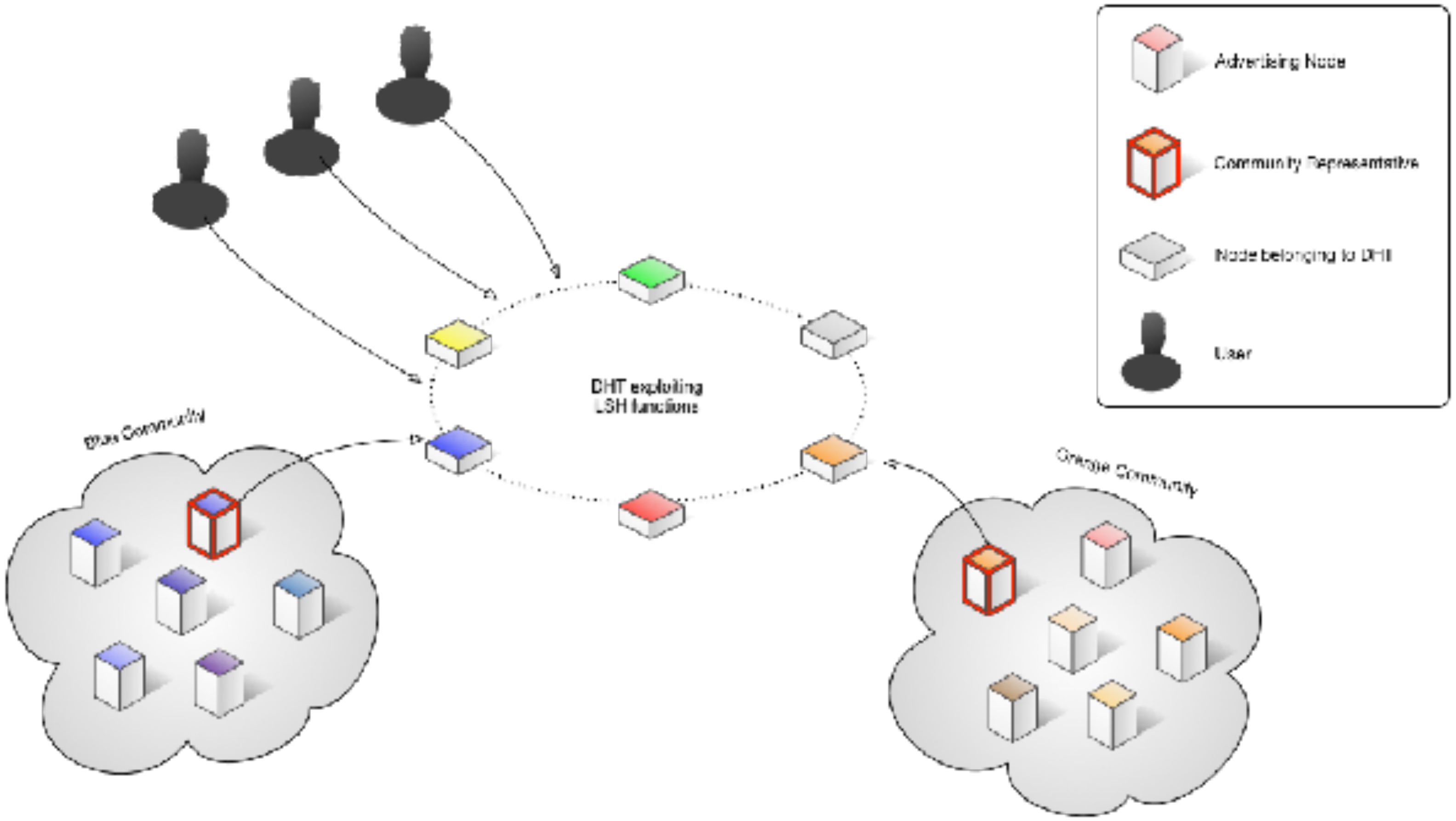
PROFILES: TERM VECTORS AND ADJACENCY MATRICES

- *Streams as collection of terms*
  - Term Vectors
    - Linear structure, weighted according to their relevance with respect to a profile
  - Adjacency Matrices
    - Weighted word of vector enriched with values estimating the correlation between attributes.
    - “Weighted” Jaccard Similarity



OVERALL ARCHITECTURE

- Two network layers
  - a **structured** overlay network made of a DHT behaving as a global index
  - an **unstructured** overlay network aimed at building communities made of similar content



UNSTRUCTURED LAYER: GROUP

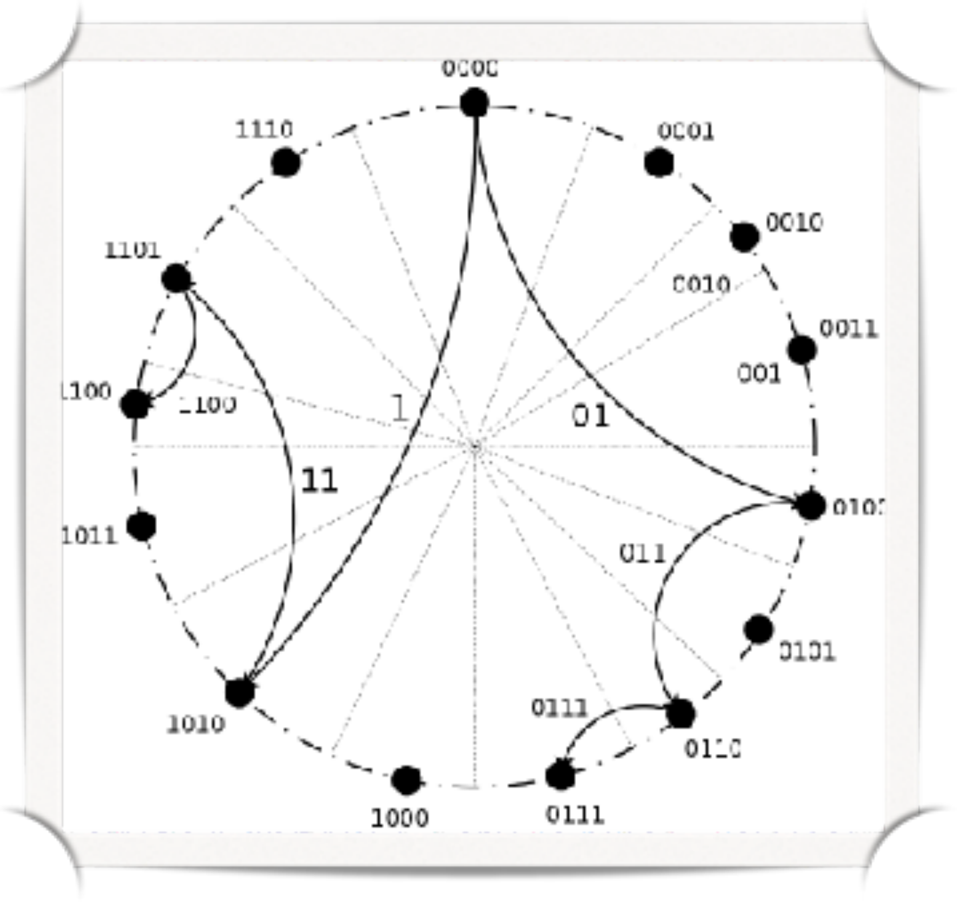
- Aimed at the detection and the creation of **self-emerging** communities made up of Advertising Nodes
- Based on a highly scalable epidemic protocol, **GROUP**
  - Group carries out communities of similar Advertising Nodes by achieving a logic partition of nodes belonging to a network
  - Each Pi includes a subset of nodes characterised by **similar profiles**. Each distinct partition Pi represents a different community.
  - To identify the communities GROUP exploits a distributed voting algorithm on the overlays built by other epidemic protocols.
  - This process is driven by the consensus that a certain AN gathers among the other ANs.

R Baraglia, P Dazzi, M Mordacchini, L Ricci
A peer-to-peer recommender system for self-emerging user communities based on gossip overlays *- Journal of Computer and System Sciences 79 (2), 291-308, 2013*

R Baraglia, P Dazzi, M Mordacchini, L Ricci, L Alessi
Group: A gossip based building community protocol
Smart Spaces and Next Generation Wired/Wireless Networking, 496-507, 2011

STRUCTURED LAYER: OVERVIEW

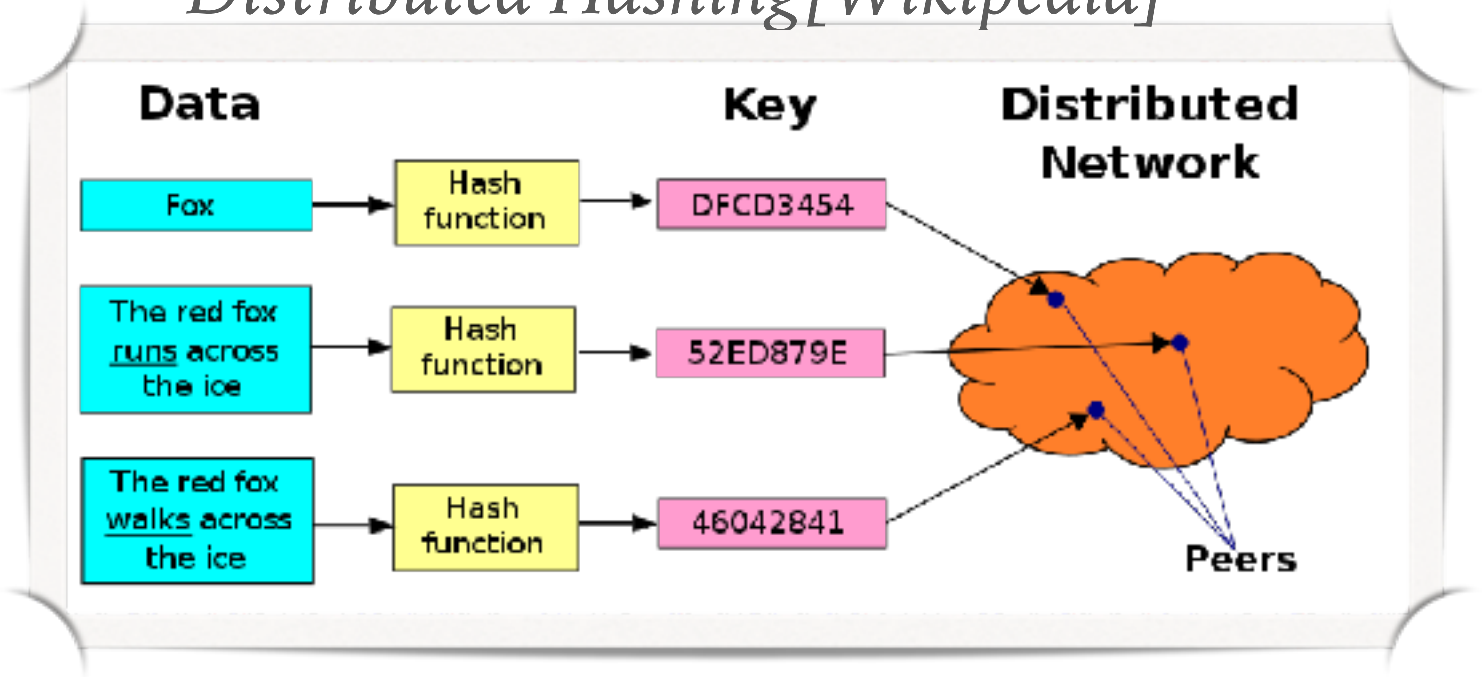
- **GROUP** enables the creation of communities of devices characterized by **similar data streams** but no support is provided for **indexing** communities **globally**.
  - To overcome this limitation, NOA-AID introduces a layer to this aim.
- A distributed, global index based on a "Special" Distributed Hash Table (DHT) able to perform approximate matches between a query and a community profile.



STRUCTURED LAYER: A “SPECIAL” DISTRIBUTED HASH TABLE

- A distributed hash table (DHT) provides a lookup service similar to a hash table
  - any participating node can efficiently retrieve the value associated with a given key.

Distributed Hashing[Wikipedia]



- Responsibility for maintaining the mapping from keys to values is distributed among the nodes
  - i.e. a change in the set of participants causes a minimal amount of disruption, thus a DHT to scale to extremely large numbers of nodes
  - DHTs form an infrastructure that can be used to build more complex services, such as anycast, cooperative Web caching, distributed file systems and also peer-to-peer file sharing and content distribution systems.
- Notable distributed networks that use DHTs include BitTorrent's distributed tracker

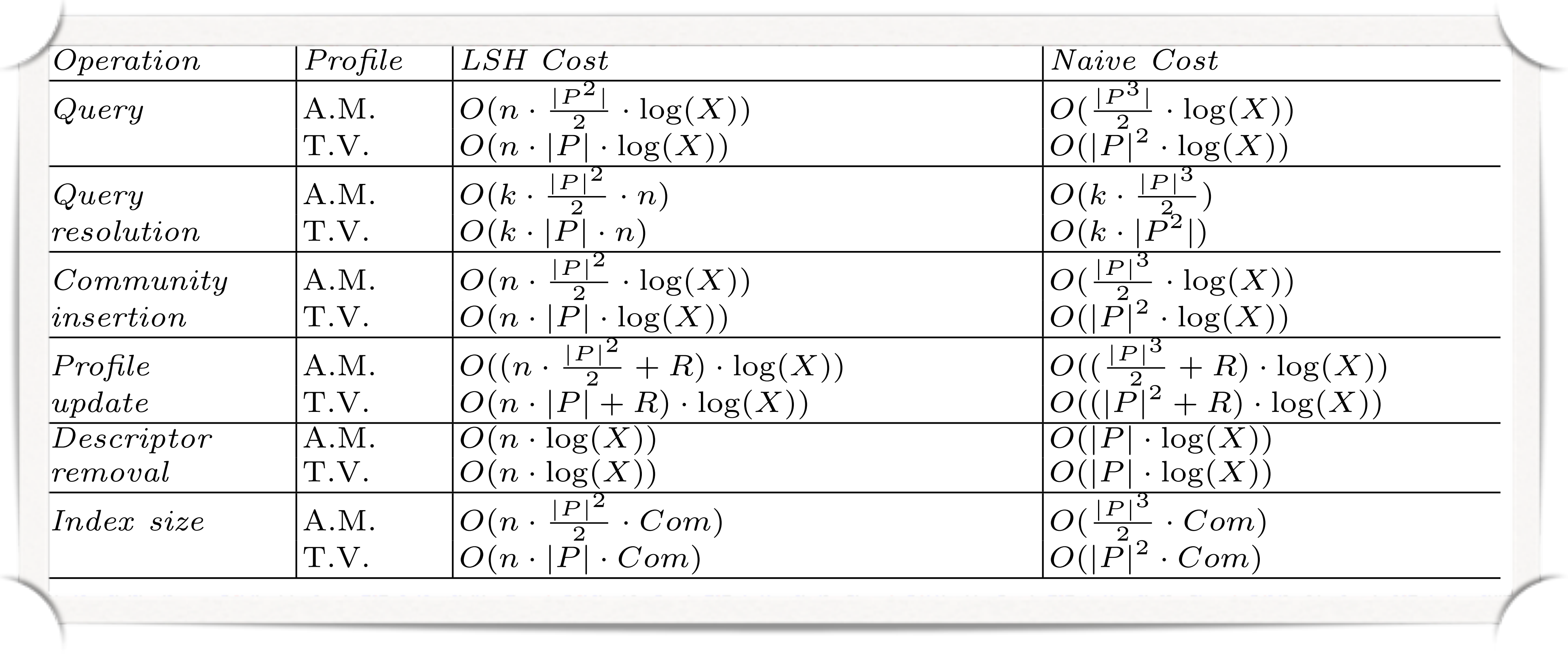
STRUCTURED LAYER: LSH AND SIMILARITY MEASUREMENTS

- However, **traditional** DHTs are very efficient to support the search for exact uni-dimensional data
- Unfortunately DHTs are not conceived for supporting approximate searches.
- Initially proposed by Zhu, The approximate search is obtained by exploiting a *Locality Sensitive Hash* (LSH) approach
- **Overall concept:** the hash function allows to map with high probability ***a*** and ***b*** in the same bucket if they are very close or in different buckets if they are quite different.

Zhu, Y., Hu, Y.
Efficient semantic search on DHT overlays
Journal of Parallel and Distributed Computing, 67(5) 604-616, 2007

STRUCTURED LAYER: COMPUTATIONAL COMPLEXITY

- Measured against the naive approach
  - The search for a profile to requires to send to a **traditional DHT** a request for each term composing the profile
  - Results would be then eventually combined searching for the profile that is maximising the overlap with the input query



Operation	*Profile*	*LSH Cost*	*Naive Cost*
Query	A.M. T.V.	$O(n \cdot \frac{\|P^2\|}{2} \cdot \log(X))$ $O(n \cdot \|P\| \cdot \log(X))$	$O(\frac{\|P^3\|}{2} \cdot \log(X))$ $O(\|P\|^2 \cdot \log(X))$
Query resolution	A.M. T.V.	$O(k \cdot \frac{\|P\|^2}{2} \cdot n)$ $O(k \cdot \|P\| \cdot n)$	$O(k \cdot \frac{\|P\|^3}{2})$ $O(k \cdot \|P^2\|)$
Community insertion	A.M. T.V.	$O(n \cdot \frac{\|P\|^2}{2} \cdot \log(X))$ $O(n \cdot \|P\| \cdot \log(X))$	$O(\frac{\|P\|^3}{2} \cdot \log(X))$ $O(\|P\|^2 \cdot \log(X))$
Profile update	A.M. T.V.	$O((n \cdot \frac{\|P\|^2}{2} + R) \cdot \log(X))$ $O(n \cdot \|P\| + R) \cdot \log(X))$	$O((\frac{\|P\|^3}{2} + R) \cdot \log(X))$ $O((\|P\|^2 + R) \cdot \log(X))$
Descriptor removal	A.M. T.V.	$O(n \cdot \log(X))$ $O(n \cdot \log(X))$	$O(\|P\| \cdot \log(X))$ $O(\|P\| \cdot \log(X))$
Index size	A.M. T.V.	$O(n \cdot \frac{\|P\|^2}{2} \cdot Com)$ $O(n \cdot \|P\| \cdot Com)$	$O(\frac{\|P\|^3}{2} \cdot Com)$ $O(\|P\|^2 \cdot Com)$

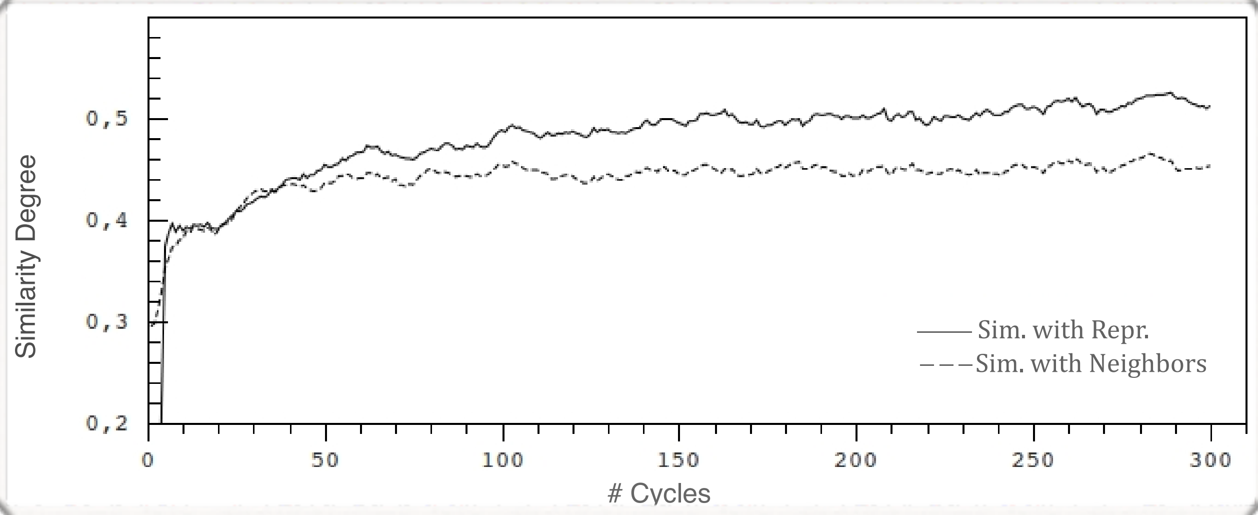
DID IT WORTH THE EFFORT ?

EVALUATION

- **Key question**
  - Is NOA-AID able to enable approximate queries over data coming from **data streams** in a distributed system based on IoT and Edge devices ?
- We measured the ability of our system to maintain high-quality community representatives when the indexed data changes, <u>i.e. significant changes in the items composing a stream</u>
- We analyse the ability of our system to efficiently resolve queries in a distributed fashion
- We validated our theoretical performance analysis comparing it against actual measured values

QUALITY OF COMMUNITY REPRESENTATIVES WHEN DATA CHANGES

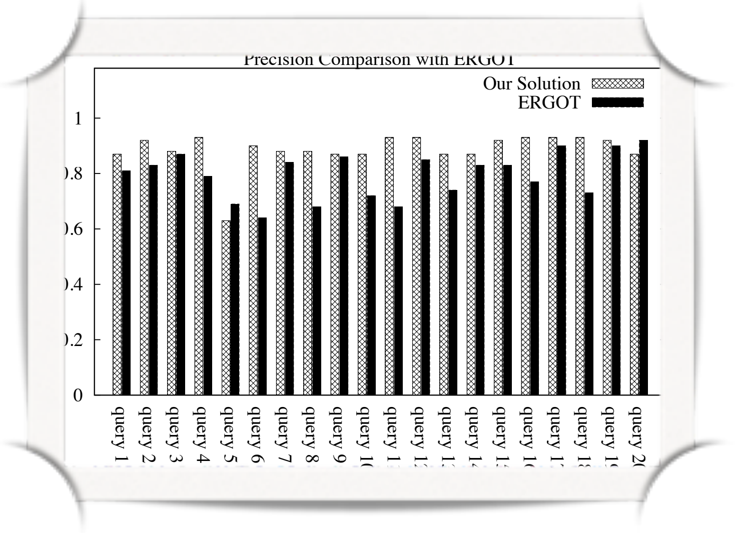
- We measured the average similarity of community members with the selected representatives, and with the other members of the same community.
- This experiment has been conducted by varying the actual composition of the information extracted by the stream processing devices starting from the simulation cycle #50.
  - Every cycle we changed the 5% of the information content of a randomly selected set representing the 2% of the nodes.
  - As can be observed, the similarity of nodes with their representative is essentially not affected by the changes. Thus the system is able to adaptively react to changes. By selecting different representatives



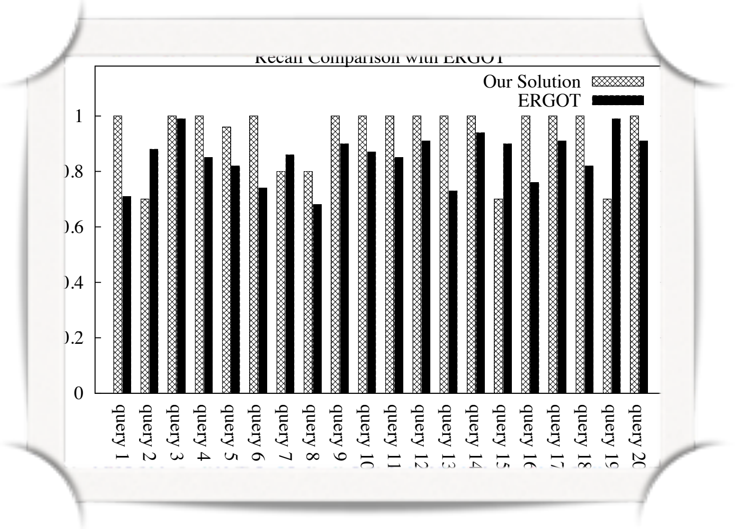
ABILITY TO EFFICIENTLY RESOLVE TEXTUAL QUERIES IN DISTRIBUTED

- Comparison against ERGOT, a DHT-based Semantic Overlay Network structured on two layers.
- The dataset is the one used in ERGOT evaluation
  - The dataset has been built by exploiting the WordNet ontology and the WordNet domain.
  - Used for generating textual streams according to a Zipf distribution.
  - The evaluation has been focused on the ability of retrieving relevant profiles given an input query.
    - Using the set of 20 queries presented in the original paper of ERGOT
      - Recall and Precision.

Precision

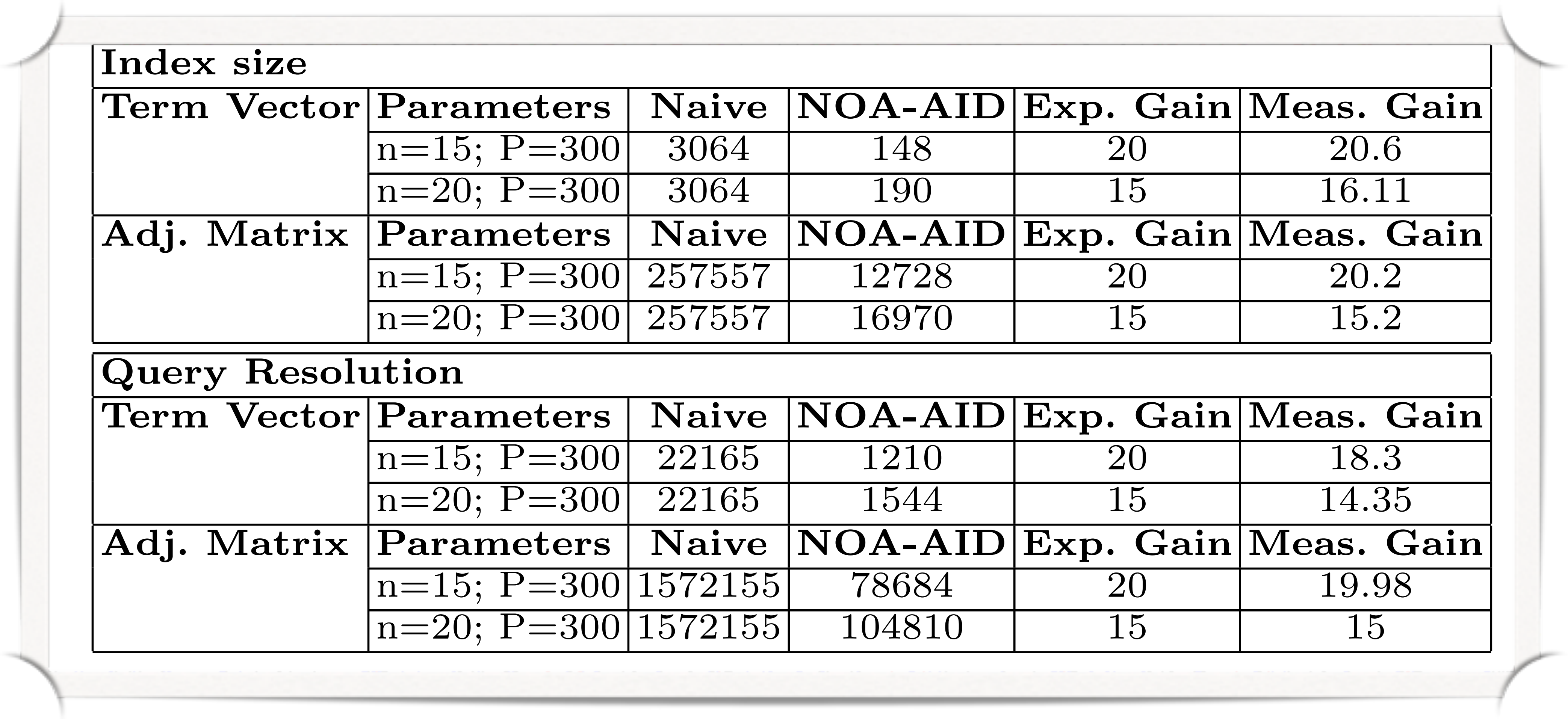


Recall



AMOUNT OF DATA NEEDED COMPARED AGAINST THE NAIVE SOLUTION

- We measured the claims we did about the computational complexity offered by our solution
- It resulted to be good estimations of the actual, measured, values



Index size					
Term Vector	**Parameters**	**Naive**	**NOA-AID**	**Exp. Gain**	**Meas. Gain**
	n=15; P=300	3064	148	20	20.6
	n=20; P=300	3064	190	15	16.11
Adj. Matrix	**Parameters**	**Naive**	**NOA-AID**	**Exp. Gain**	**Meas. Gain**
	n=15; P=300	257557	12728	20	20.2
	n=20; P=300	257557	16970	15	15.2

Query Resolution					
Term Vector	**Parameters**	**Naive**	**NOA-AID**	**Exp. Gain**	**Meas. Gain**
	n=15; P=300	22165	1210	20	18.3
	n=20; P=300	22165	1544	15	14.35
Adj. Matrix	**Parameters**	**Naive**	**NOA-AID**	**Exp. Gain**	**Meas. Gain**
	n=15; P=300	1572155	78684	20	19.98
	n=20; P=300	1572155	104810	15	15

CONCLUDING REMARKS

CONCLUSION

- An overlay network architecture
  - providing a flexible query-by-example indexing and discovery mechanism
  - targeting stream processing devices belonging to a highly dynamic and distributed environments.
- Two overlays
  - lower level unstructured, epidemic-based, network able to autonomously adapt and self-organize
    - aimed at grouping stream processing devices into communities
  - higher network layer indexes such communities and provides a query-by-example solution easing their discovery.
- We provided both a theoretical as well as a experimental evaluation of the approach showing its effectiveness and efficiency.

ACKNOWLEDGEMENTS

- This work has been partly supported by H2020 EU-Korea BASMATI project
  - http://www.basmati.cloud

QUESTIONS ?