Dealing with Skewed Data in Structured Overlays using Variable Hash Functions

Maeva Antoine, Fabrice Huet

University of Nice Sophia-Antipolis (France), CNRS, I3S, UMR 7271

The 15th International Conference on Parallel and Distributed Computing, Applications and Technologies

December 9-11, 2014 • Hong Kong

PDCAT 2014

Context

- Many applications integrate data at web scale to extract information & knowledge:

→ Big Data (Facebook, Twitter, Wikipedia, …)

Context

- Big Data is…
  - hard to manage on a single machine

    → P2P: large scale solution for Big Data management systems.
  - highly biased

    → requires a suitable load balancing solution

Context

- Big Data is…
  - hard to manage on a single machine

    → P2P: large scale solution for Big Data management systems.

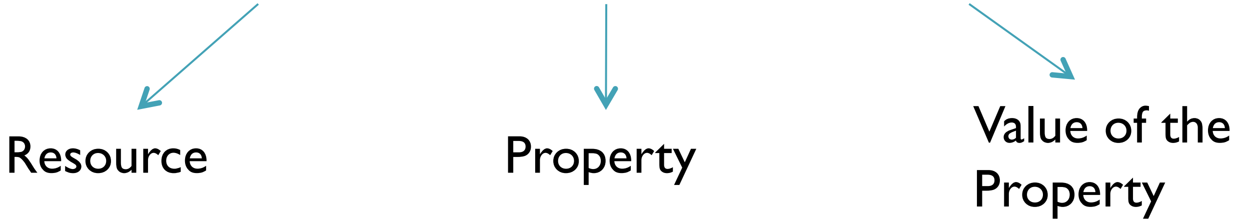
  - highly biased & **continuously produced**

    → requires a suitable & **adaptive** load balancing solution

The Semantic Web

- « Web of Data »
- Tools for describing knowledge and reasoning on web data.
- RDF triple format to represent data:

<Subject ; Predicate ; Object>



Resource

Property

Value of the Property

Exploiting Big Data: DBpedia

- RDF triple:

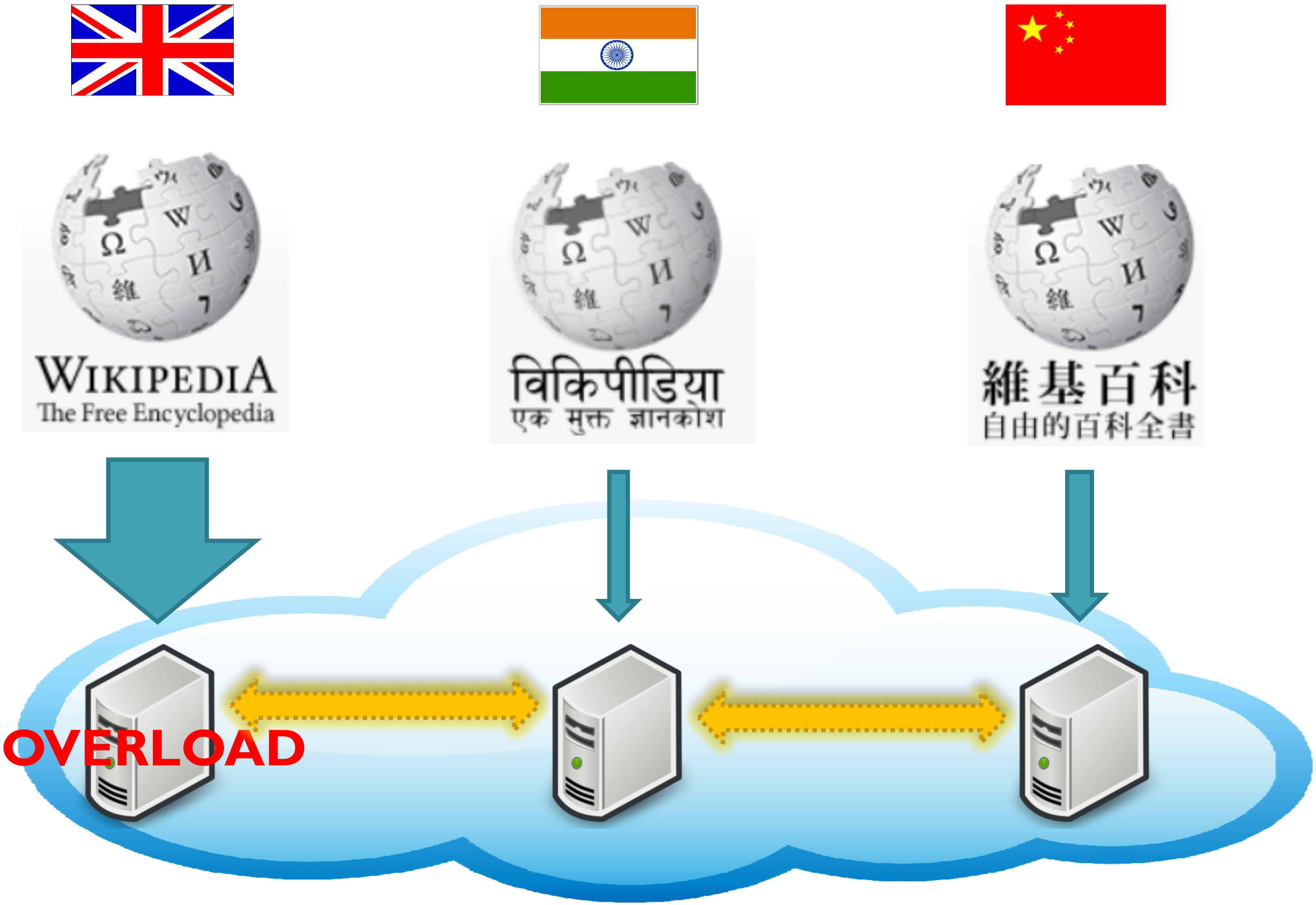
S • http://dbpedia.org/resource/Vienna

P • http://www.w3.org/2000/01/rdf-schema#abstract

O • "Vienna is the capital and largest city of Austria, …”

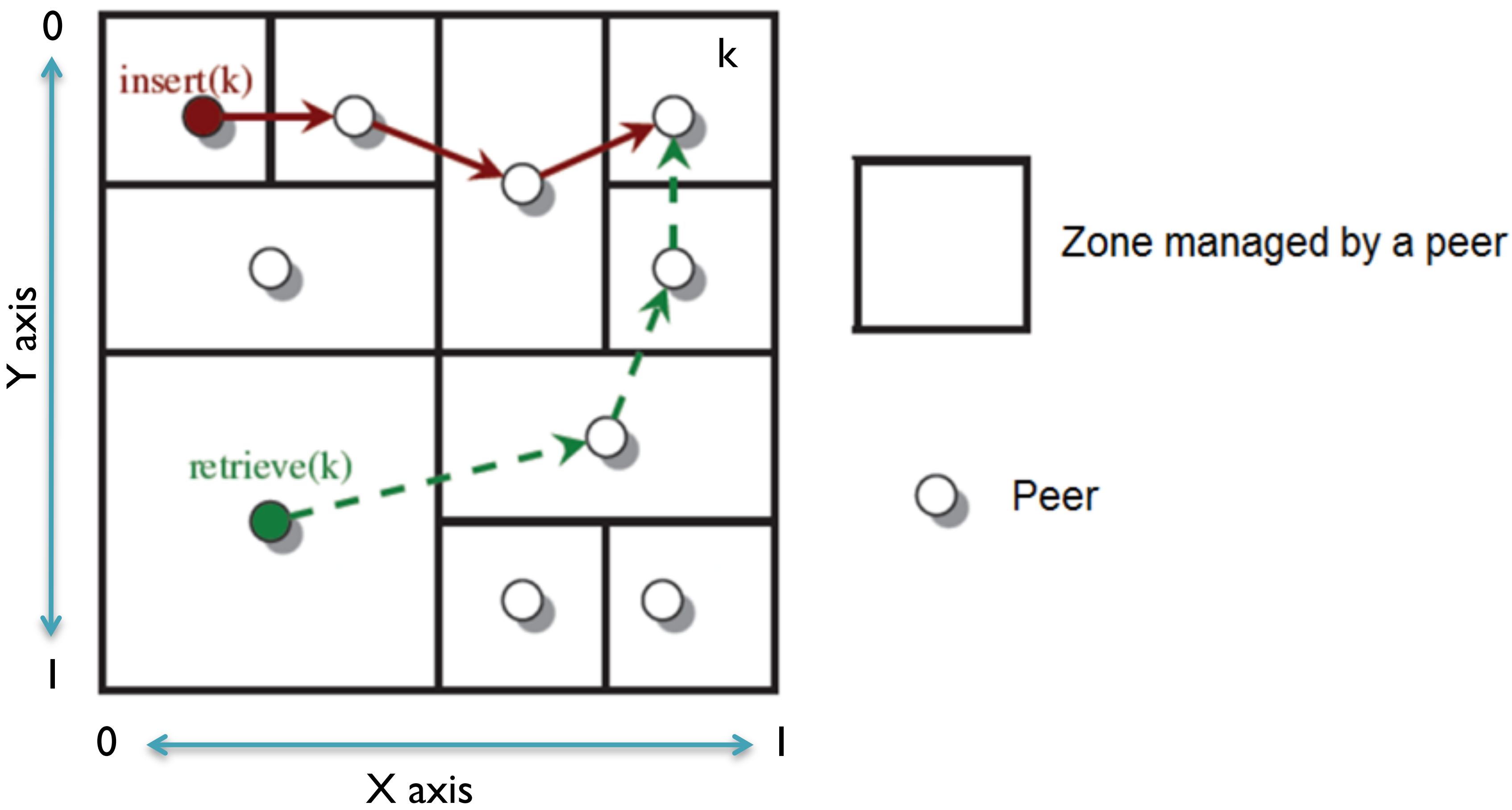
- Information extracted from Wikipedia.
- Datasets available in 125 languages

DBpedia in a Distributed System



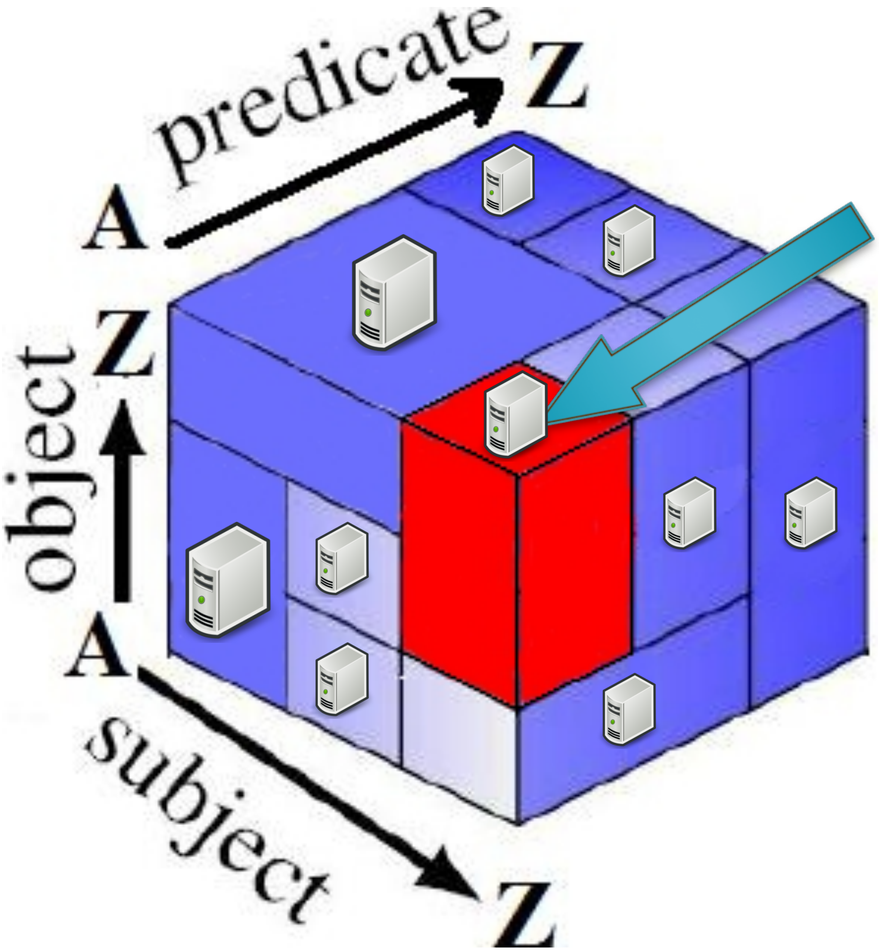
Content Addressable Network (CAN)

- Decentralized P2P infrastructure
- n dimensions are possible, example of a 2 dimensional CAN:



CAN storing RDF data

3 dimensional lexicographic CAN:



subject: Vienna

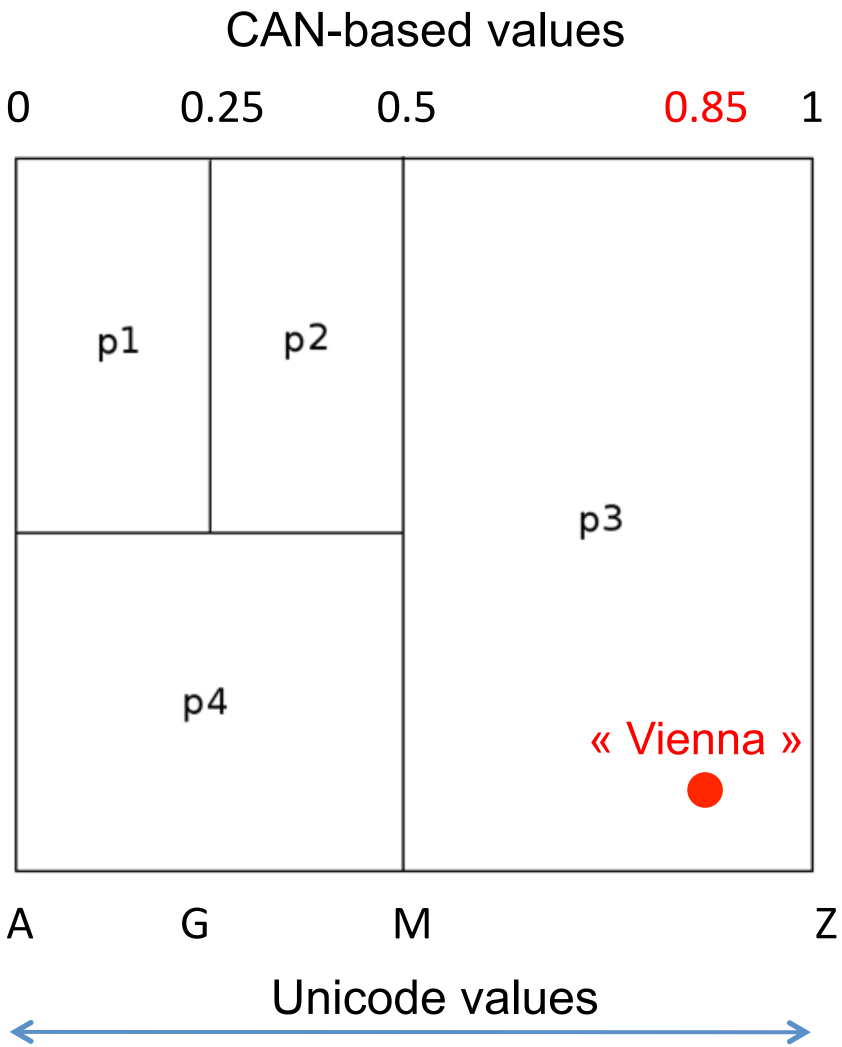
predicate: abstract

object: «Vienna is the capital and largest city of Austria... »

Related information stored by the same peer:

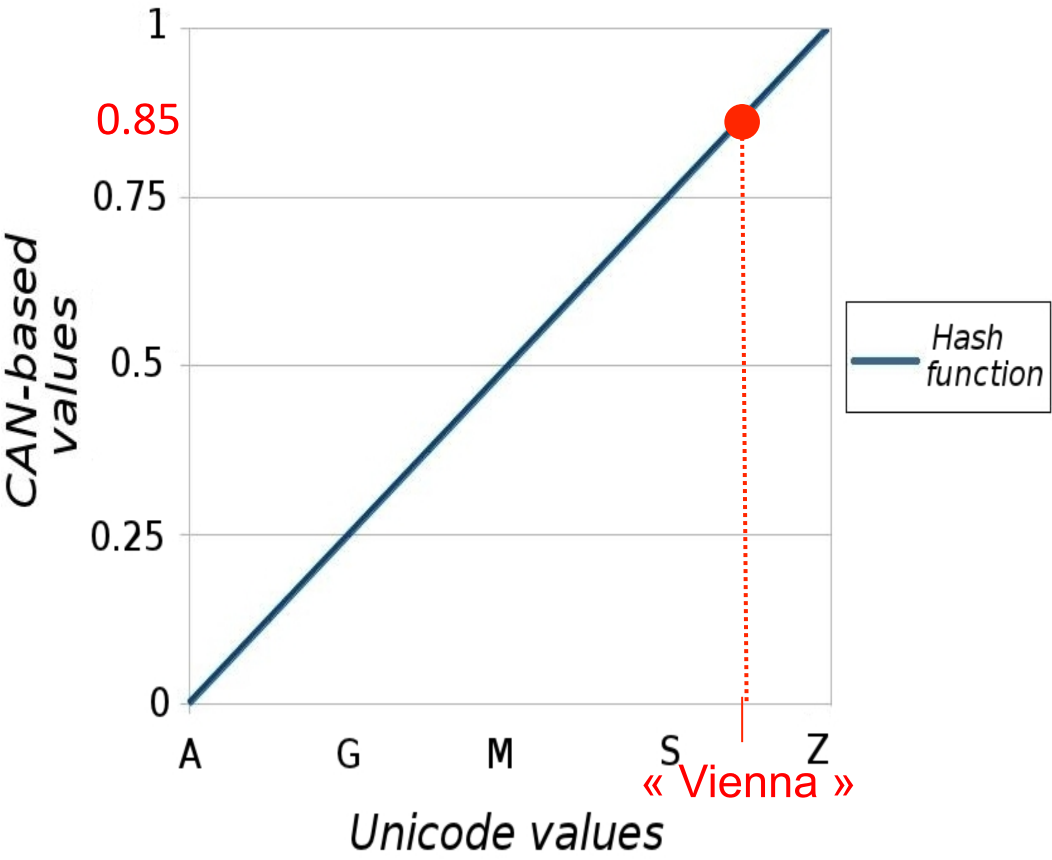
Vienna_International_Airport
Vienna_Festival
Vienna_State_Opera_Ballet
Vienna_Cricket_and_Football-Club
…

Default Hash Function



Subject value of a triple

Hash(Vienna) = 0.85



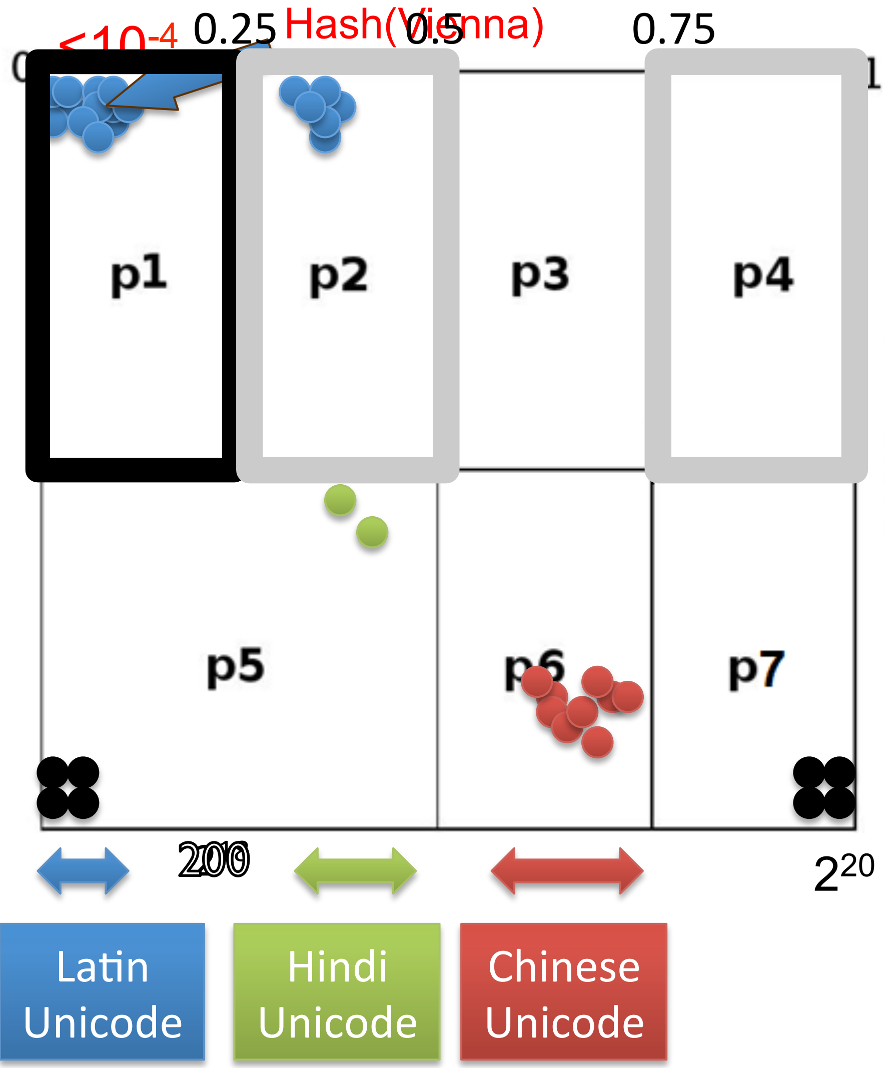
Hash function for the horizontal dimension

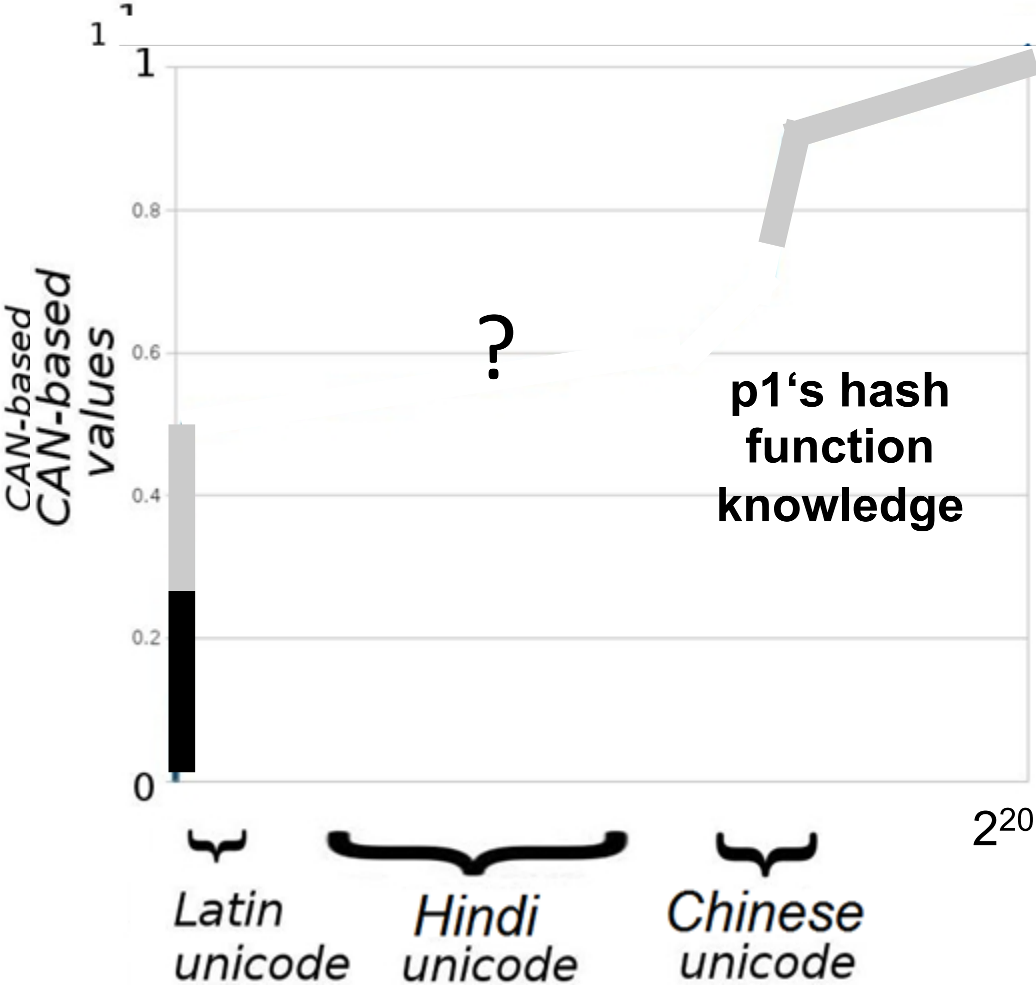
Skewed Data: Balanced Distribution

Skewed Data: Flexible Distribution

Why? Uniform hash function for a wide interval: $[0;2^{20}[$

CAN-based values

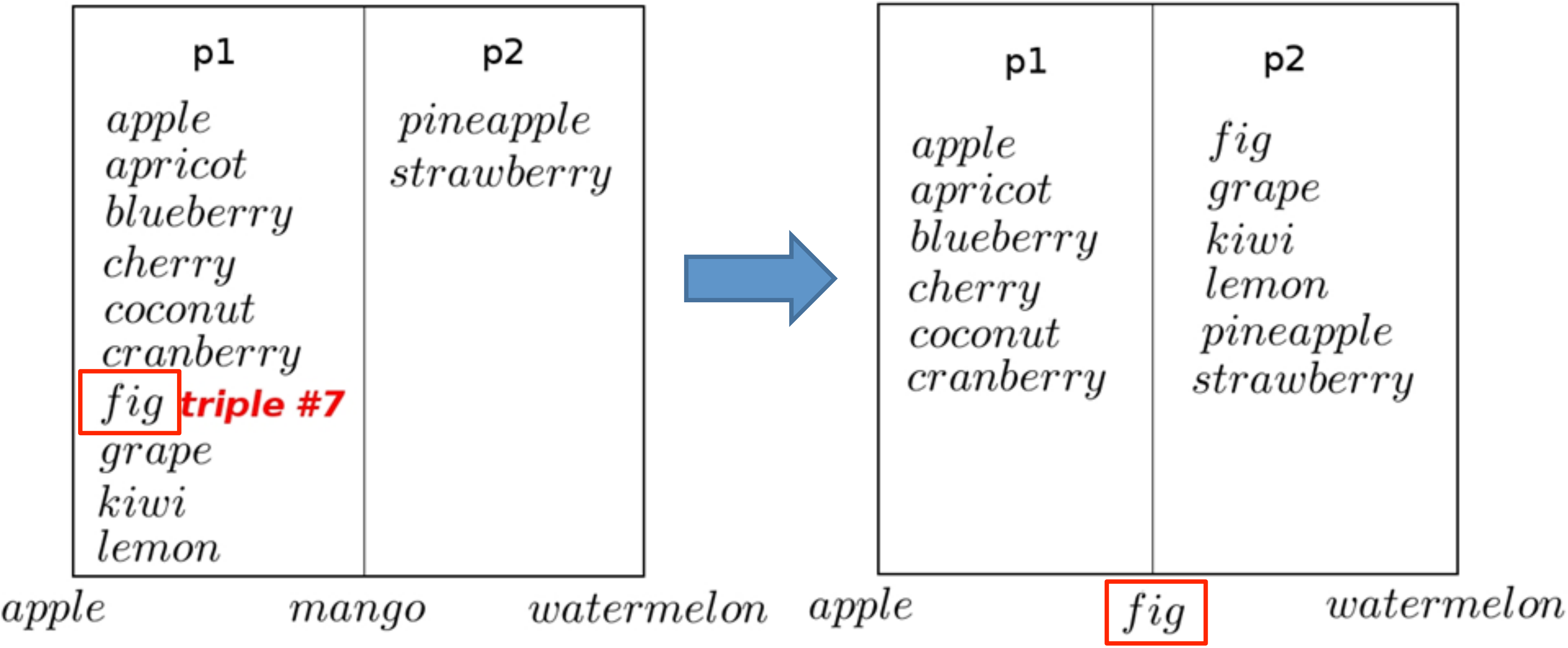




Computing a New Hash Function

How to determine the new value of a bound?

Threshold = 6

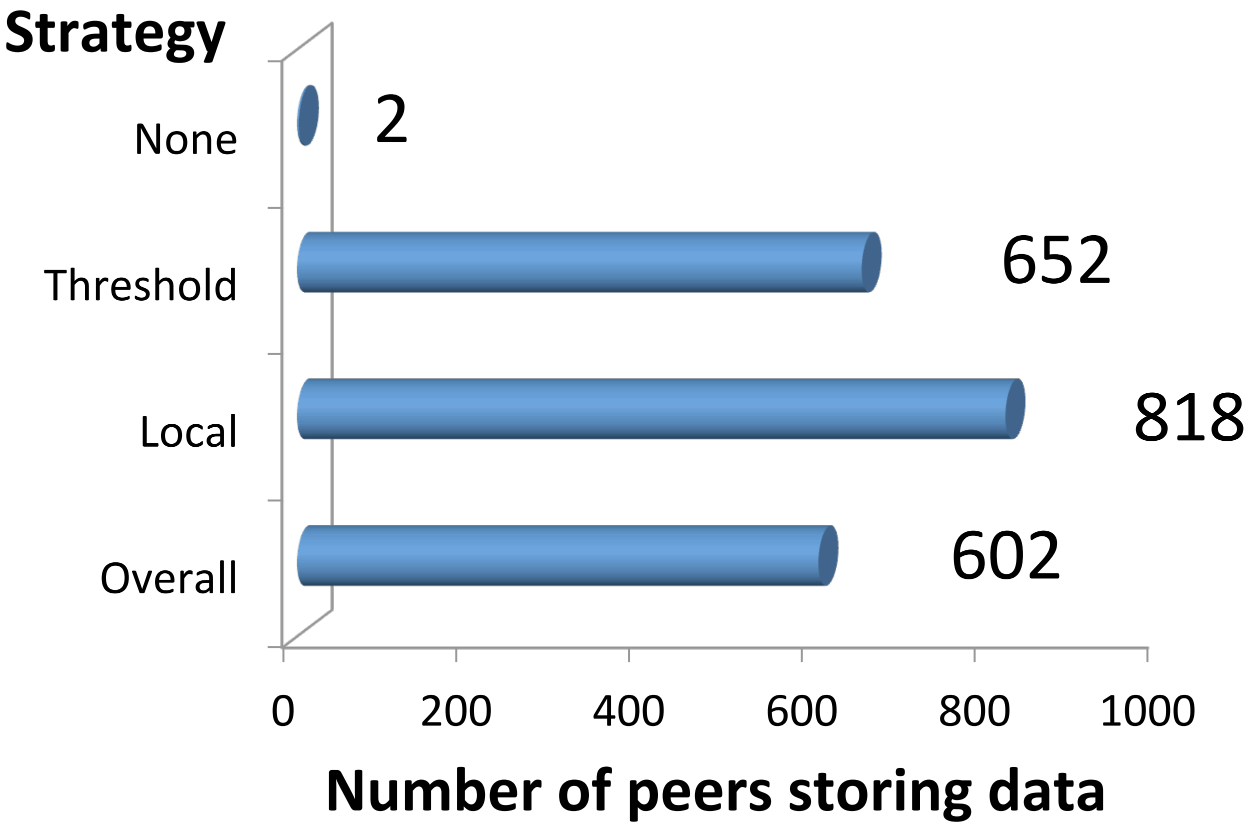


Load Balancing Strategies

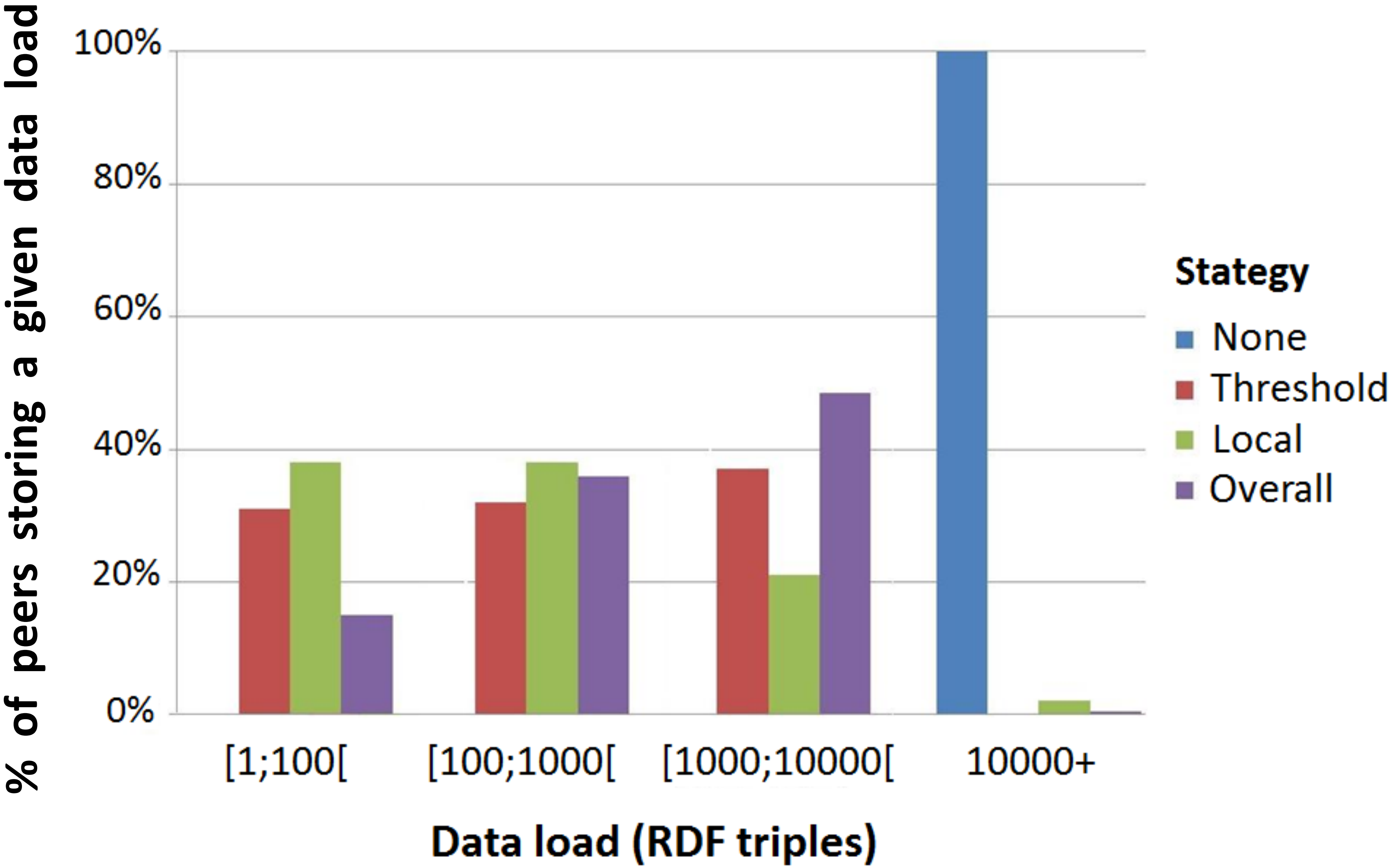
- Threshold: no load information exchanged
  - → New bound value = first Unicode value above threshold
- Local: load of neighbors to determine new value
- Overall: using average/estimate of network load

Experiments

Inserted **1 million** highly biased triples (English & Japanese DBpedia) in a network made of **1000 peers**.



Data Distribution among Peers



Conclusion

- Dynamic adaptation of hash functions to data skewness.
- It is not necessary for all peers to use the same hash function.

→ Improved data distribution without *a priori* knowledge.

→ Same principles are applicable on other DHT overlays.

The End

- Thank you!

- Questions?