Distributed Search Revisited:

Resolving the Conflict of Efficiency & Flexibility

Raouf Boutaba

(*with Reaz Ahmed*)

Cheriton School of Computer Science

University of Waterloo

http://bcr2.uwaterloo.ca/~rboutaba

University of Waterloo

IEEE LCN

October 16, 2008

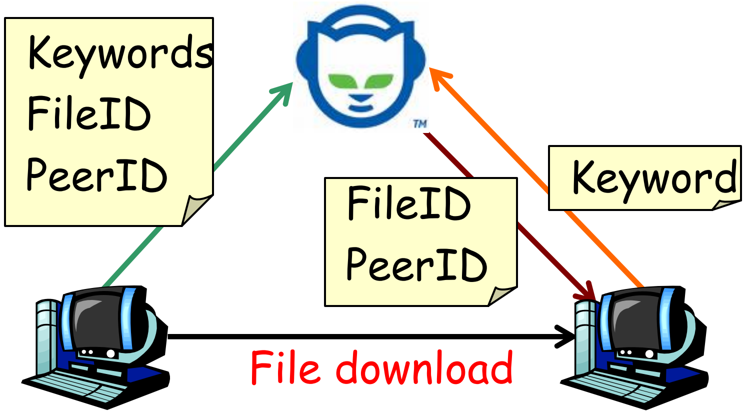
Outline

- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusion

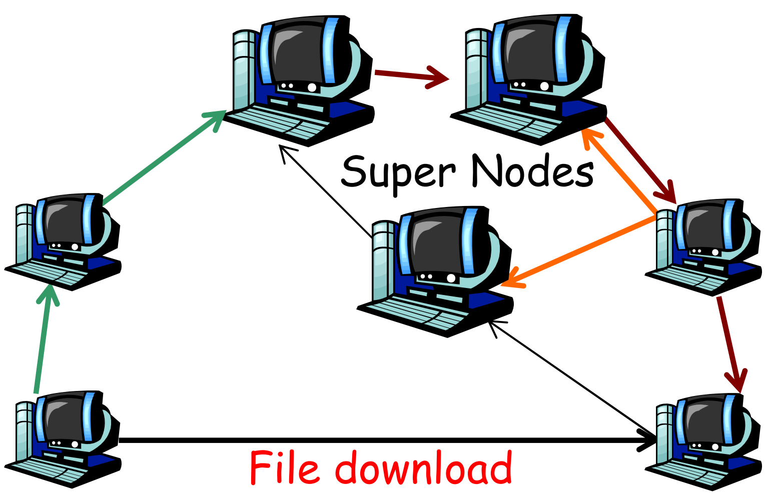
Large Scale Distributed Systems

- **Properties**
  - *Transient* populations of *autonomous* nodes
  - Content *dynamism*
  - *Heterogeneity* in nodes' capabilities
- **Representative domains**
  - P2P content sharing
  - Service discovery
  - Distributed XML databases
- **Search requirements**
  - Efficiency
  - Flexibility
  - Robustness
  - Completeness
  - Autonomy
  - Anonymity

Content-sharing P2P Systems



Centralized
(*Napster*)



Decentralized
(Kazaa, Gnutella2)

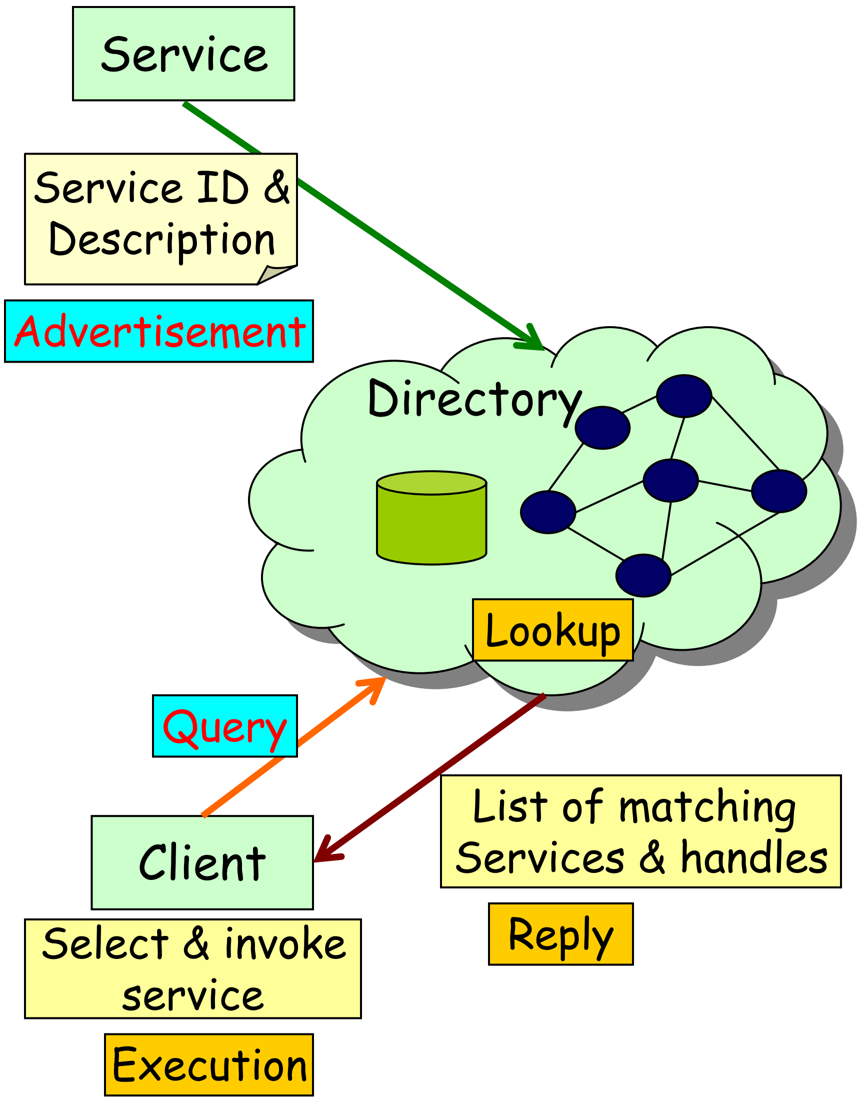
Advertisement

The Lord of the Rings - The Two Towers - 2002 (Extended Edition) DVDrip.AVI

Query

Lord of the Ring Two Tower

Service Discovery



Advertisement

Service-type = service:print

Scope-list = staff, grad

Location = DC3335

Color = true

Language = PS

Paper-size = legal, A4, B5

Query

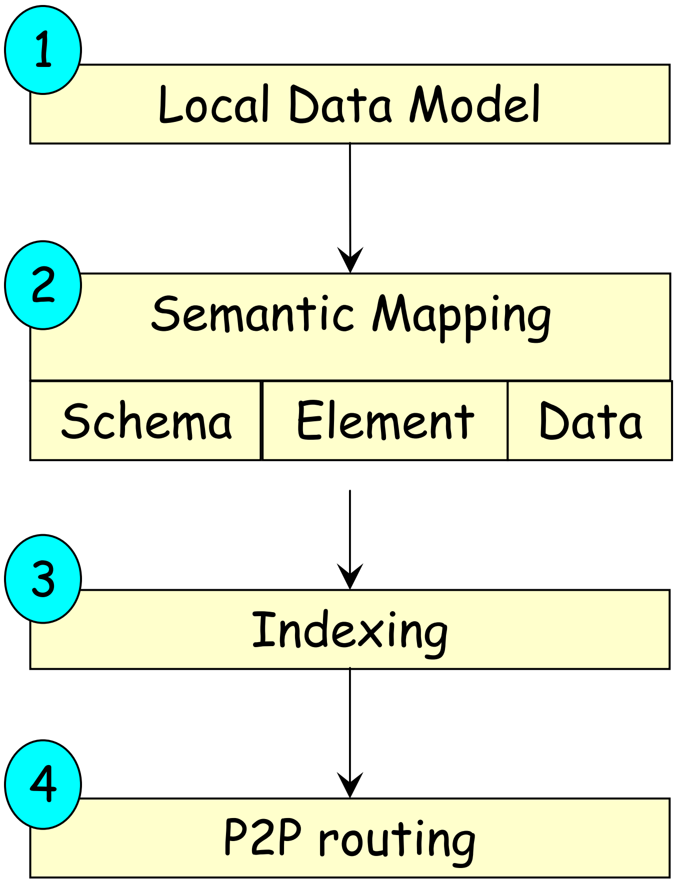
Service-type = service:print

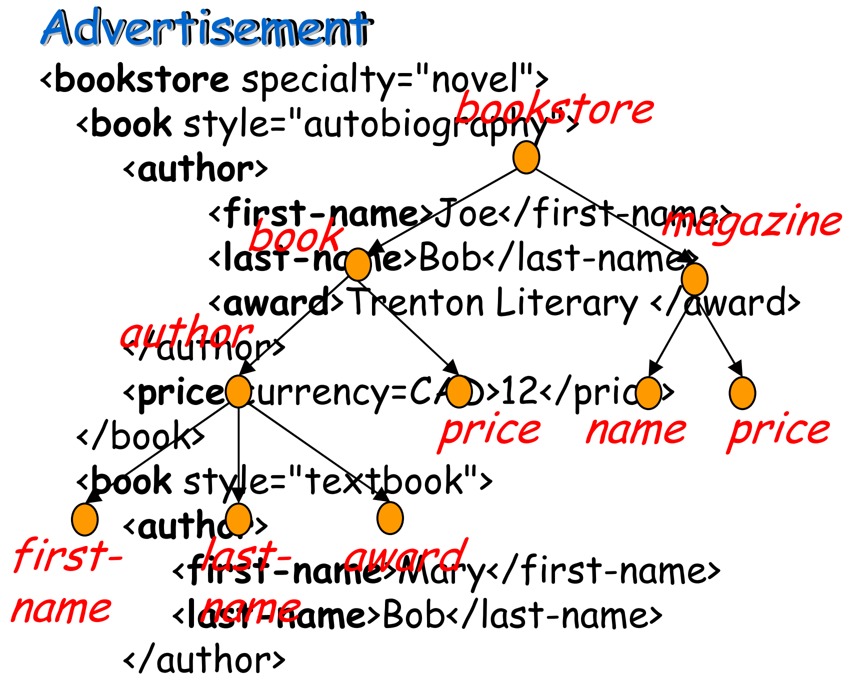
Scope-list = grad

Paper-size = A4

Resource and Service Discovery in Large Scale Distributed Systems. IEEE Communications Surveys &Tutorials, IEEE Press, Vol. 9 (4), pp. 2-30, Dec. 2007.

P2P Databases





XPath Query

//author[award]

/bookstore/book[author/last-name=Bob]

<

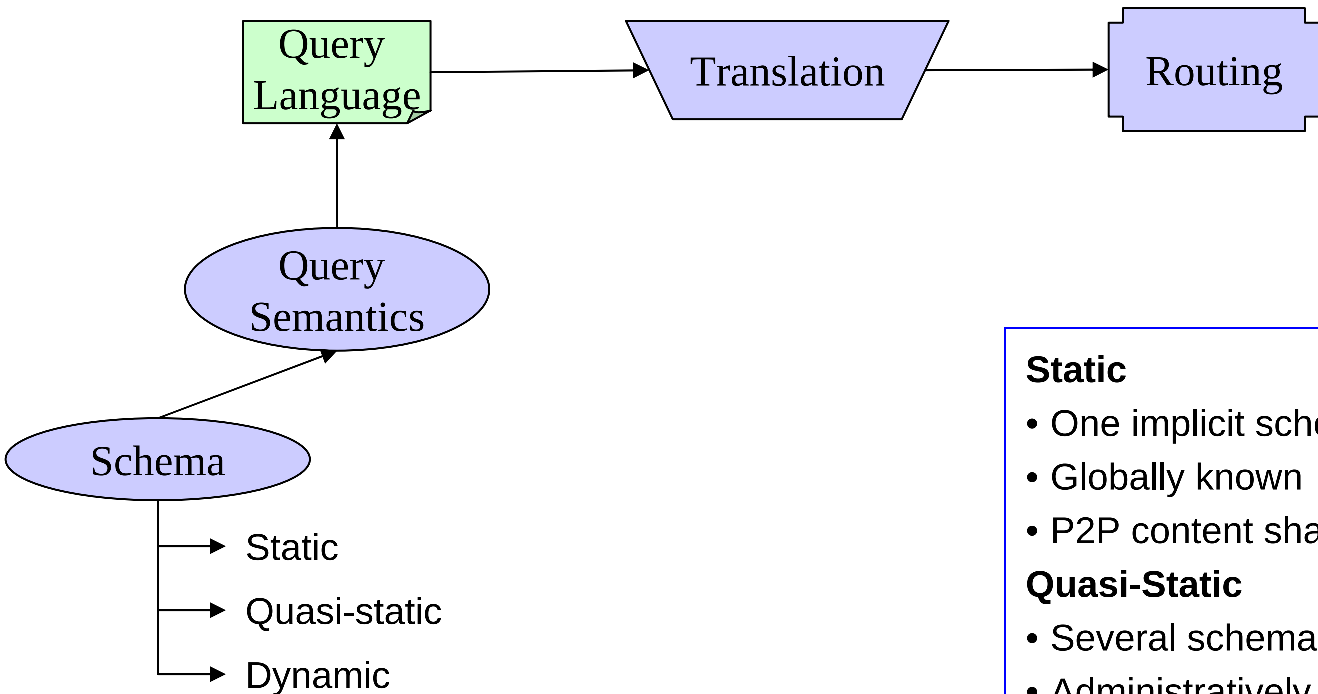
Bottom-line

- Query is based on partial information about the Advertisement.
- Query is a "subset" of an Advertisement it should match against

Outline

- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusion

Components of a Search Mechanism



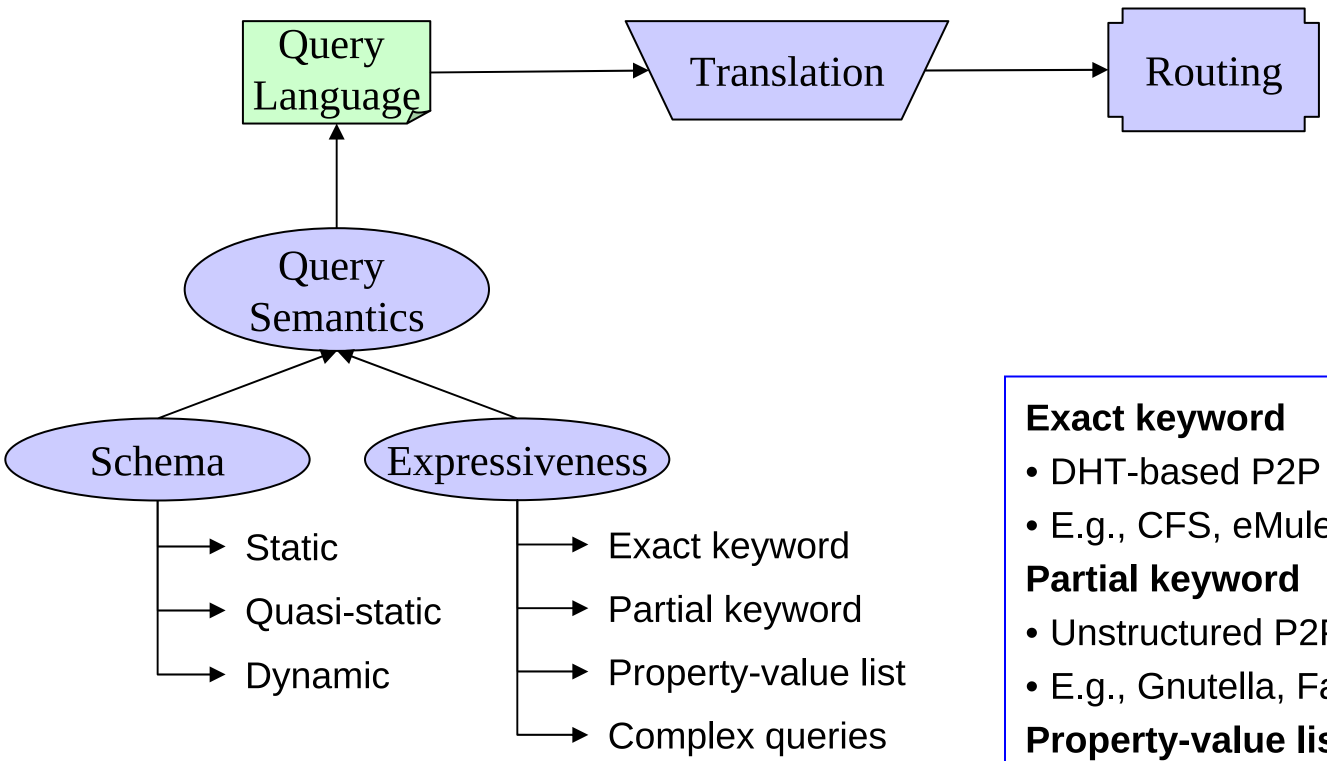
Static
- One implicit schema
- Globally known
- P2P content sharing: Gnutella, KaZaA etc.

Quasi-Static
- Several schemas, occasionally created
- Administratively scoped
- Service discovery: Jini, SLP, Salutation

Dynamic
- Heterogeneous schemas
- User scoped, semantic mapping needed
- PDBS: PeerDB, XP2P, RDFPeers etc.

A Survey of Distributed Search Techniques in Large Scale Distributed Systems. IEEE Surveys &Tutorials, IEEE Press, cdt. accepted 2007.

Components of a Search Mechanism



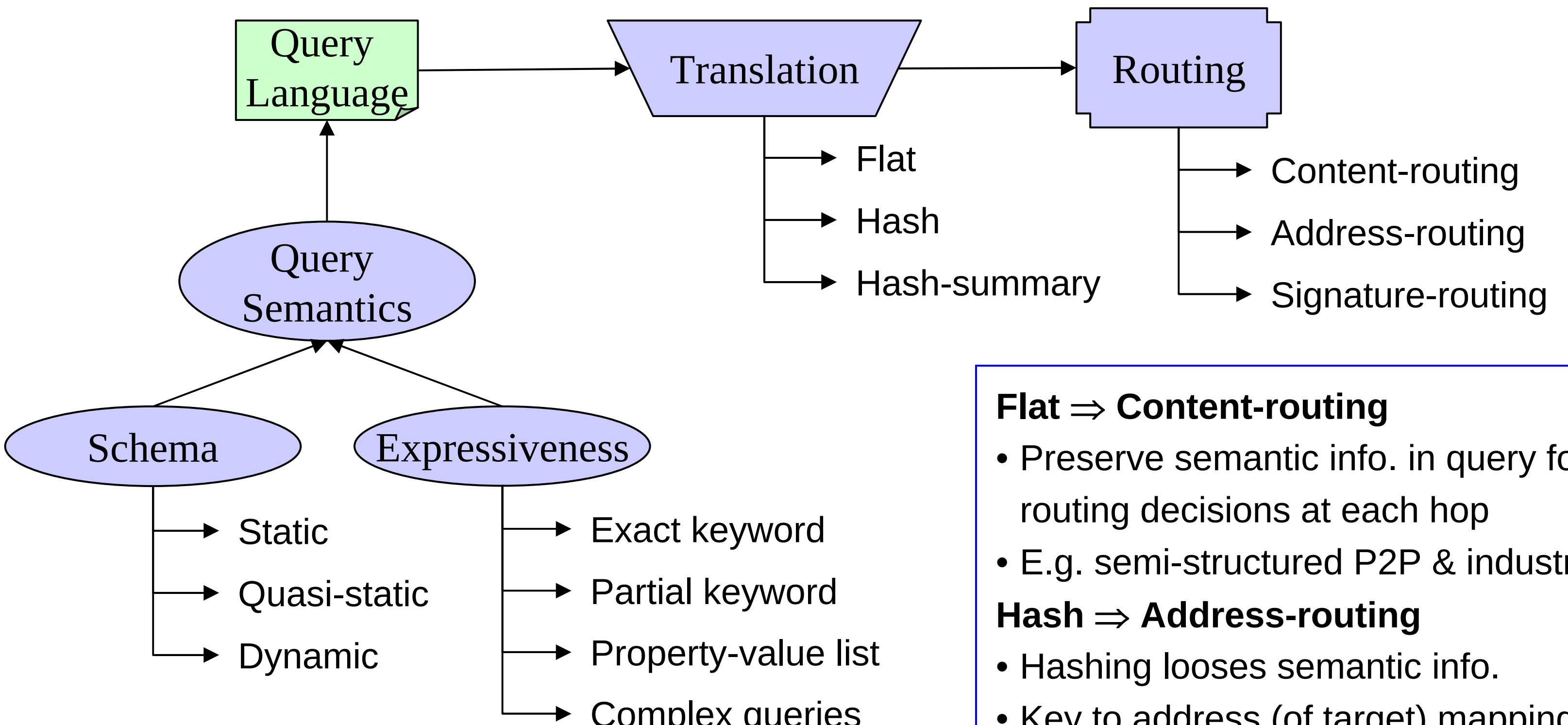
Exact keyword
- DHT-based P2P content sharing
- E.g., CFS, eMule

Partial keyword
- Unstructured P2P
- E.g., Gnutella, Fasttrack

Property-value list
- Most service discovery protocols (SDPs)
- E.g., Jini, Salutation, UPnP

Complex queries
- PDBSs and some SDPs
- Hierarchical, relational op. and ranges
- E.g., PeerDB, XP2P, SLP, Twine.

Components of a Search Mechanism



Flat ⇒ Content-routing

- Preserve semantic info. in query for use in routing decisions at each hop
- E.g. semi-structured P2P & industrial SDP

Hash ⇒ Address-routing

- Hashing looses semantic info.
- Key to address (of target) mapping
- E.g., DHT-techniques, SkipNet

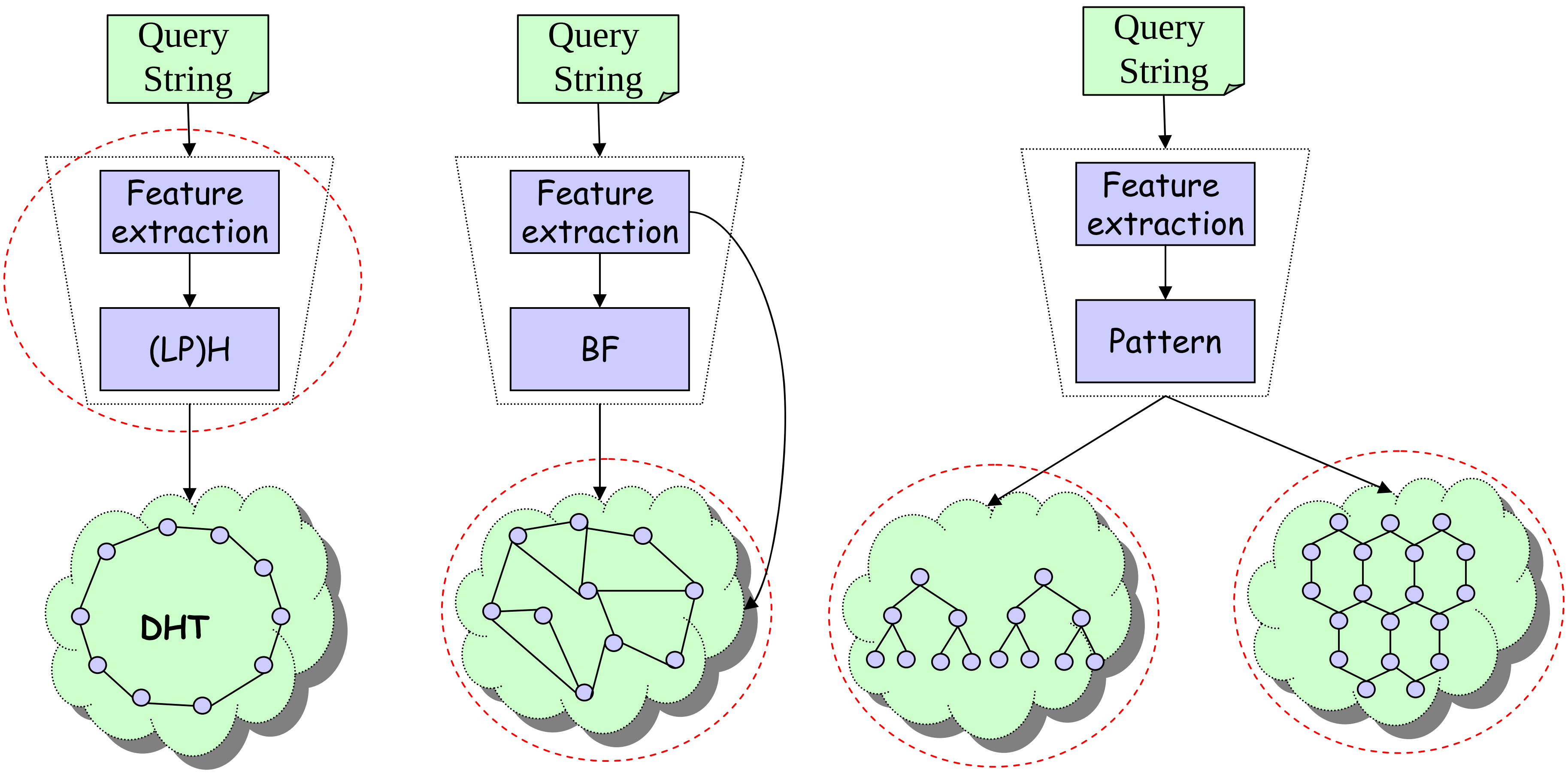
Hash-summary ⇒ Signature-routing

- Query semantic is preserved
- Bloom-filter based & lossy aggregation
- E.g., SSDS, NSS, **DPMS**, PLR

Examples

	GIA	pSearch	Squid	Twine & PWSD	NSS	SSDS
Query	Keyword	Full-text/ semantic	Prefix-match	Subtree / path: XML	Keyword	Subset of AV-list: XML
Trans-lation	Flat	LSI	Hilbert SFC	Stranding	Bloom-filter	Bloom-filter
Routing	R.Walk+ Cap. bias+ 1-hop idx.	CAN	Chord	Chord	Controlled flooding	Global Hierarchy
	PeerDB	XP2P	L. Galanis	RDFPeers	PLR	Humbolt
Query	SQL	XPath (absolute)	XPath (relative)	Partial RDF triple	Keyword	SPARQL/ RDF
Trans-lation	Synonym	Finger-print	XML elem. Hash	RDF elem. hash	Attenuat. Bloom filter	URI-hash + flat
Routing	TTL-flooding	Chord	Chord	Chord	Hint-based	DHT+ Ctrl. flooding

Research Trends



Address routing,
Multiple lookup/search?

Signature/Content routing,
Search overhead?
Rare item?

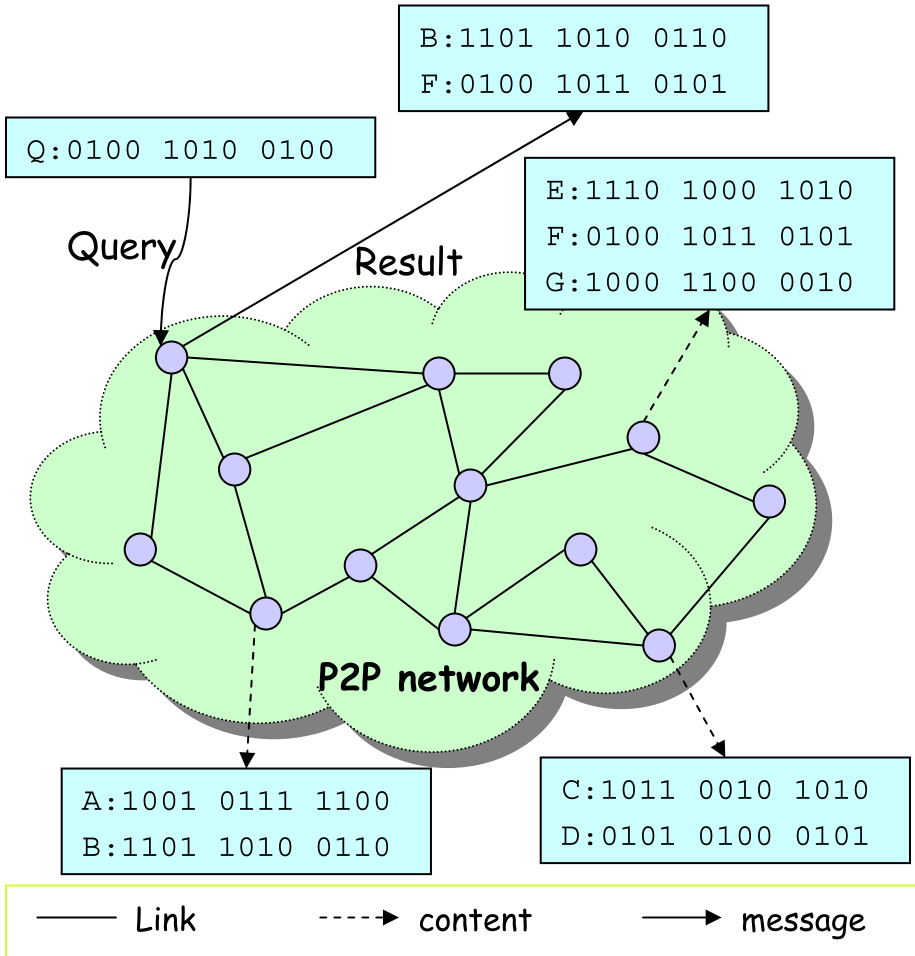
DPMS
Global hierarchies with
Replication & aggregation
(Signature routing)
Pattern as Index

Plexus
Index clustering &
ECC based
(Address routing)
Pattern as Address

Outline

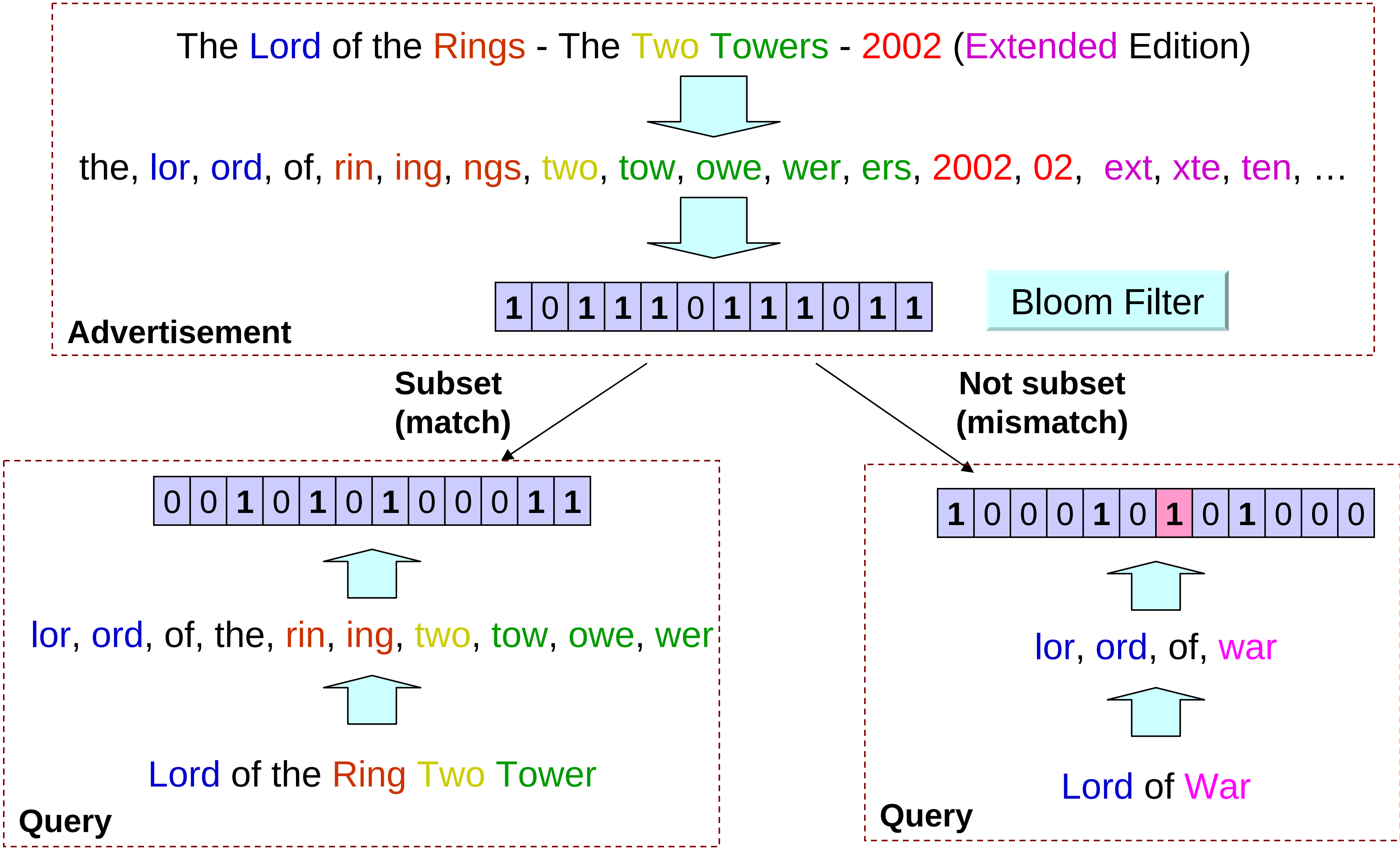
- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusion

Distributed Pattern Matching (DPM)

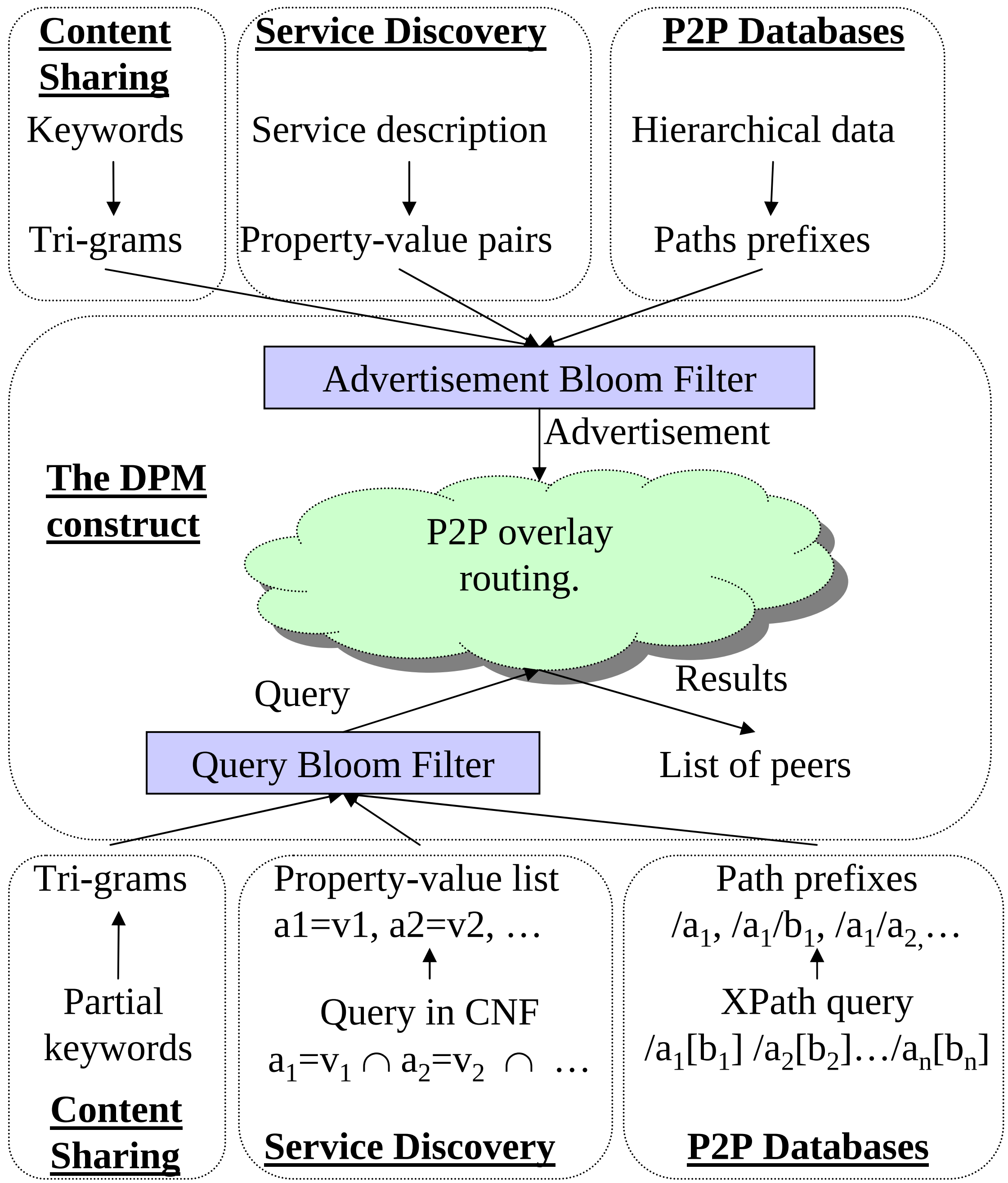


Distributed Pattern Matching for P2P Systems. In Proc. IEEE/IFIP Symposium on Network Operations and Management (NOMS), Vancouver (Canada), April 2006.

Mapping Keywords to Patterns



The Big Picture



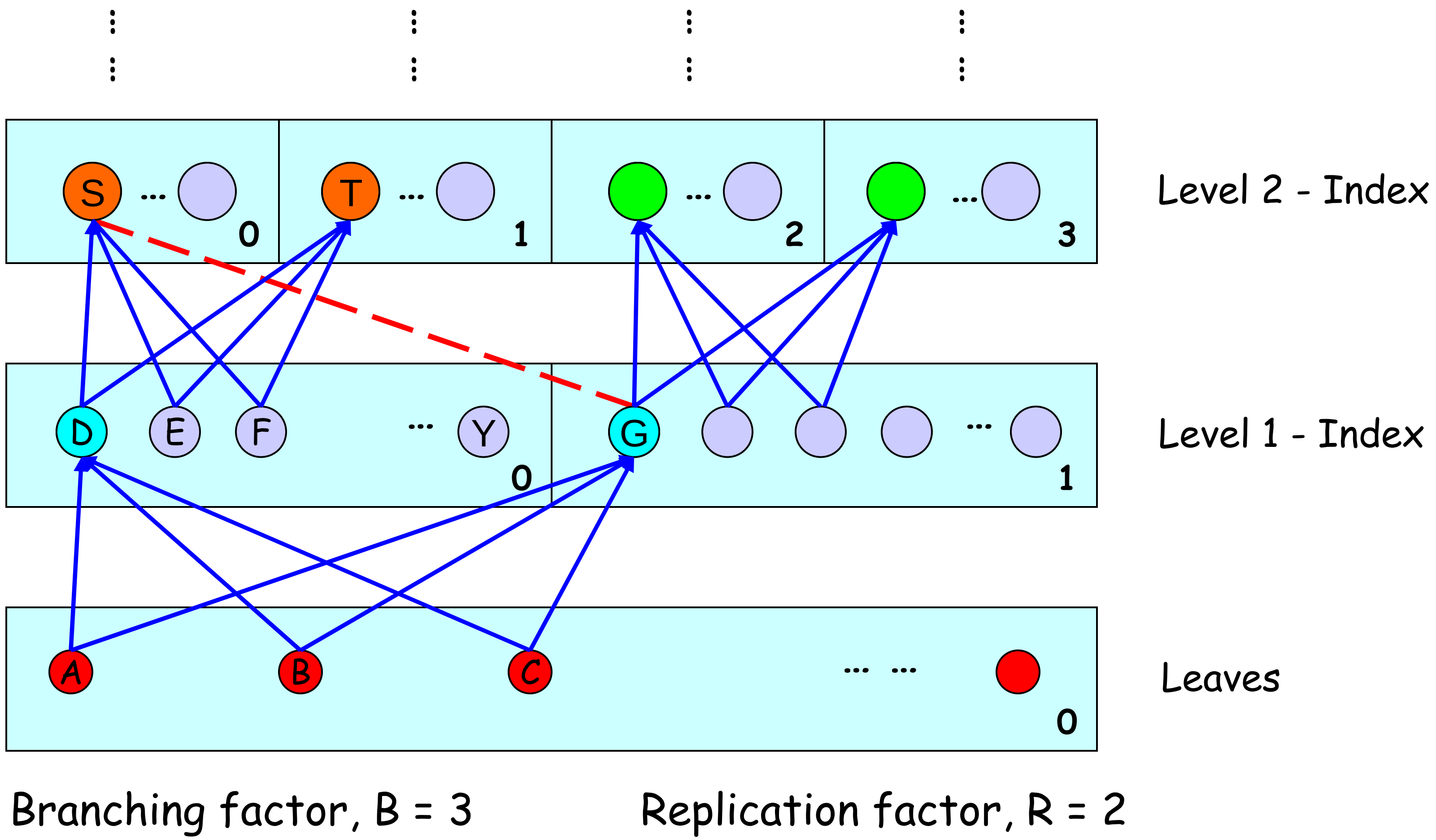
Solving the DPM Problem

- Challenge: PM requires linear time algorithm
- Solutions:
  - **DPMS:**
    - ***Signature routing***
    - ***Hierarchical indexing*** with index aggregation
    - Goal: Find few matches in a few hops
  - **Plexus:**
    - ***Address Routing***
    - ***Index clustering*** with Error Correcting Codes
    - Goal: Find all in reasonable number of hops

Outline

- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusion

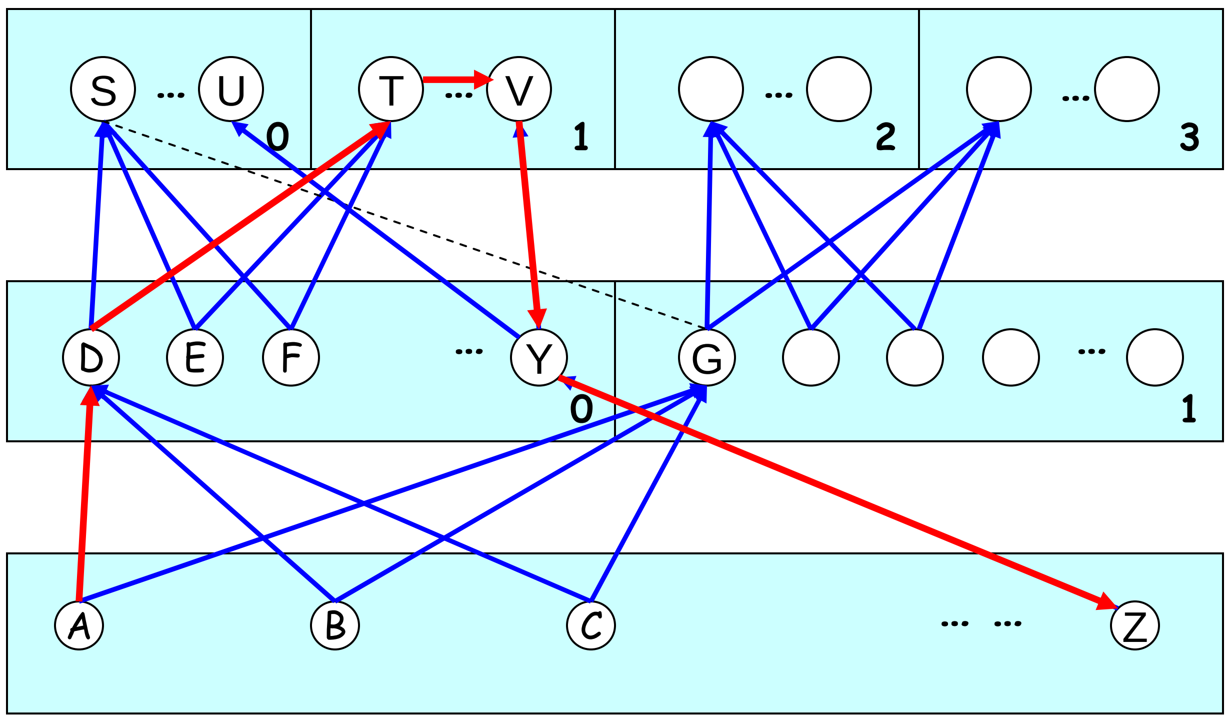
Pattern Distribution



Distributed Pattern Matching: A Key to Flexible and Efficient P2P Search. IEEE Journal on Selected Areas in Communications (JSAC), Vol. 25 (1), pp. 73-83, 2007.

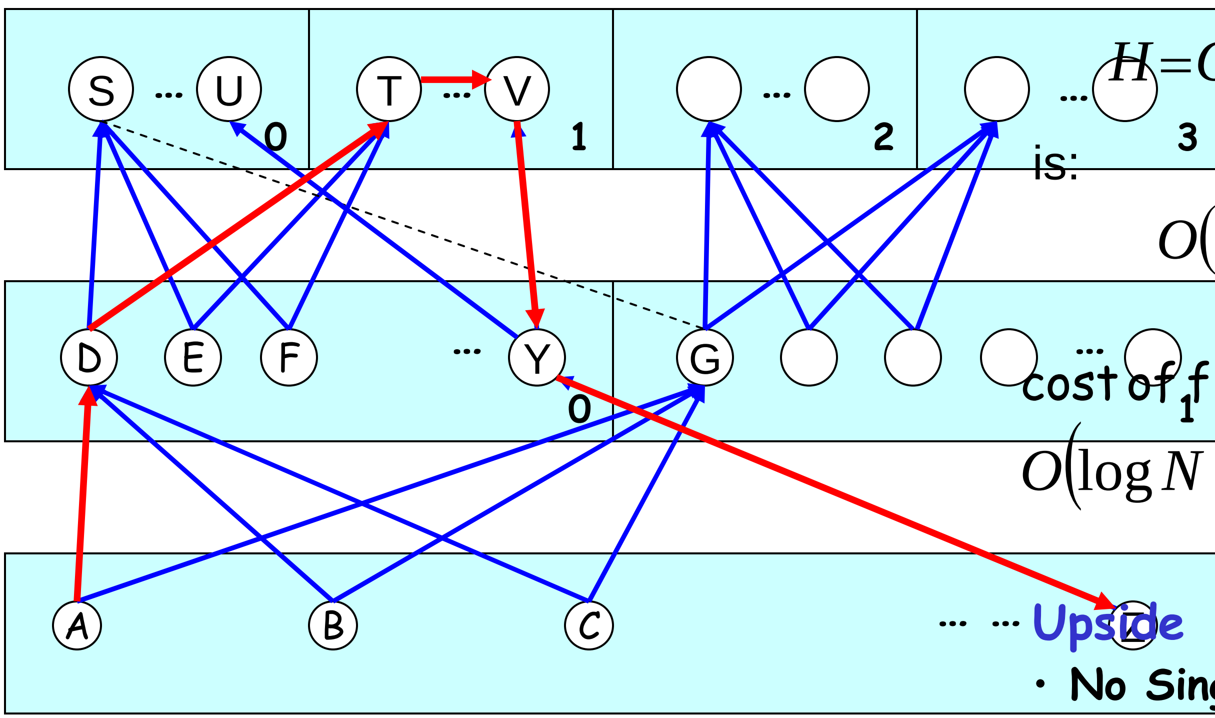
Query Routing

Peer **A** is looking for a pattern, say advertised by **Z**



Query Routing

Peer **A** is looking for a pattern, say advertised by **Z**



Efficiency?

Number of peers in a group at height, say H, where,

$$H = O\left(\log \frac{N}{\log N}\right)$$

is:

$$O(\log N)$$

cost of finding κ matches

$$O\left(\log N + \varepsilon\kappa \log \frac{N}{\log N}\right)$$

Upside

- No Single point of failure
- No Performance bottleneck

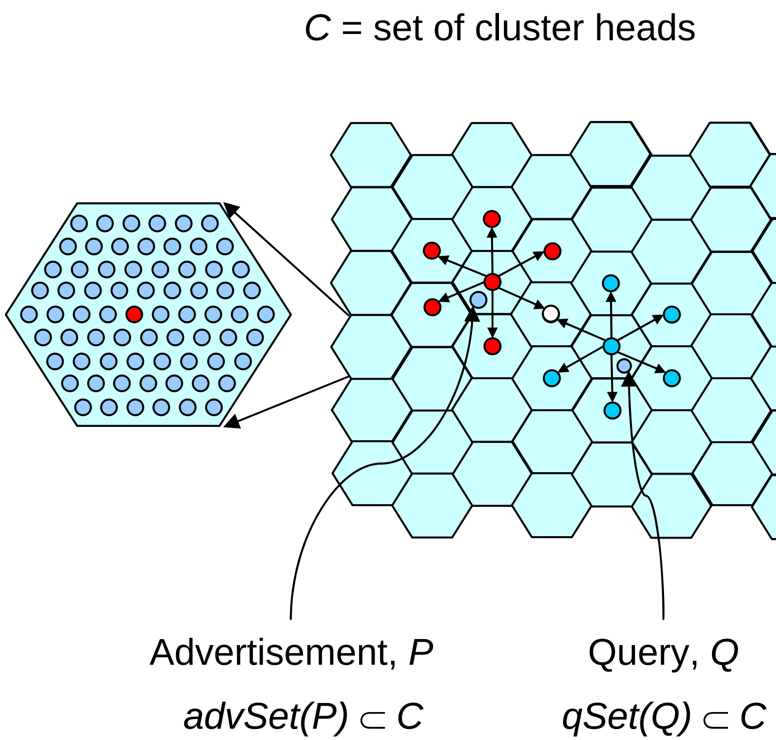
Downside

- Indexing overhead

Outline

- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusion

Plexus: Index Clustering



Error Correcting Codes

- *Linear Covering code*
- Cluster head ⇔ *Codeword*
- *Generator matrix* based routing

$$Q \subseteq P \Rightarrow qSet(Q) \cap advSet(P) \neq \phi$$

Plexus: A Scalable Peer-to-peer Protocol Enabling Efficient Subset Search. IEEE Transactions on Networking (TON), IEEE Press, To appear February 2009.

Linear Binary Code

- $\mathcal{C}$ = <n, k, d> linear binary code
  - **n**: number of bits in a codeword
  - **k**: dimension → 2^k codewords in code
  - **d**: minimum distance between any pair of codewords
  - e.g., $\mathcal{G}_{24}$=<24, 12, 8>
- Generator Matrix ***G***,

$$G = \begin{bmatrix} g_1 \\ g_2 \\ \cdots \\ \cdots \\ g_k \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & \cdots & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & \cdots & g_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ g_{k1} & g_{k2} & \cdots & \cdots & g_{kn} \end{bmatrix}$$

- 2^k codewords can be formed by applying XOR to any combination of these k rows.

Plexus: Routing Table

- In a *complete network* each peer is responsible for a codeword
- Peer with codeword X maintains **k** links as follows:
  - Link $\mathbf{X_i = X \oplus g_i}$ $1 \leq i \leq k$
- Optionally X can link to:
  - $\mathbf{X_{k+1} = X \oplus g_1 \oplus g_2 \oplus \ldots \oplus g_k}$
  - Replicate to X_{k+1}

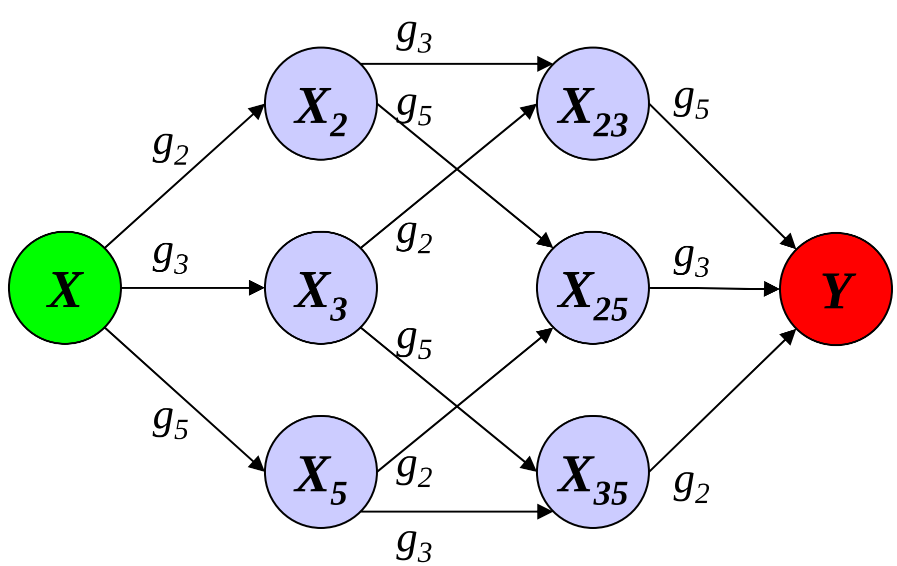
Plexus: Routing

- Observation: $\mathcal{C}$ is closed under $\oplus$ operation

$$X, Y \in C \Rightarrow Y = X \oplus g_{i_1} \oplus g_{i_2} \oplus \cdots \oplus g_{i_t}$$

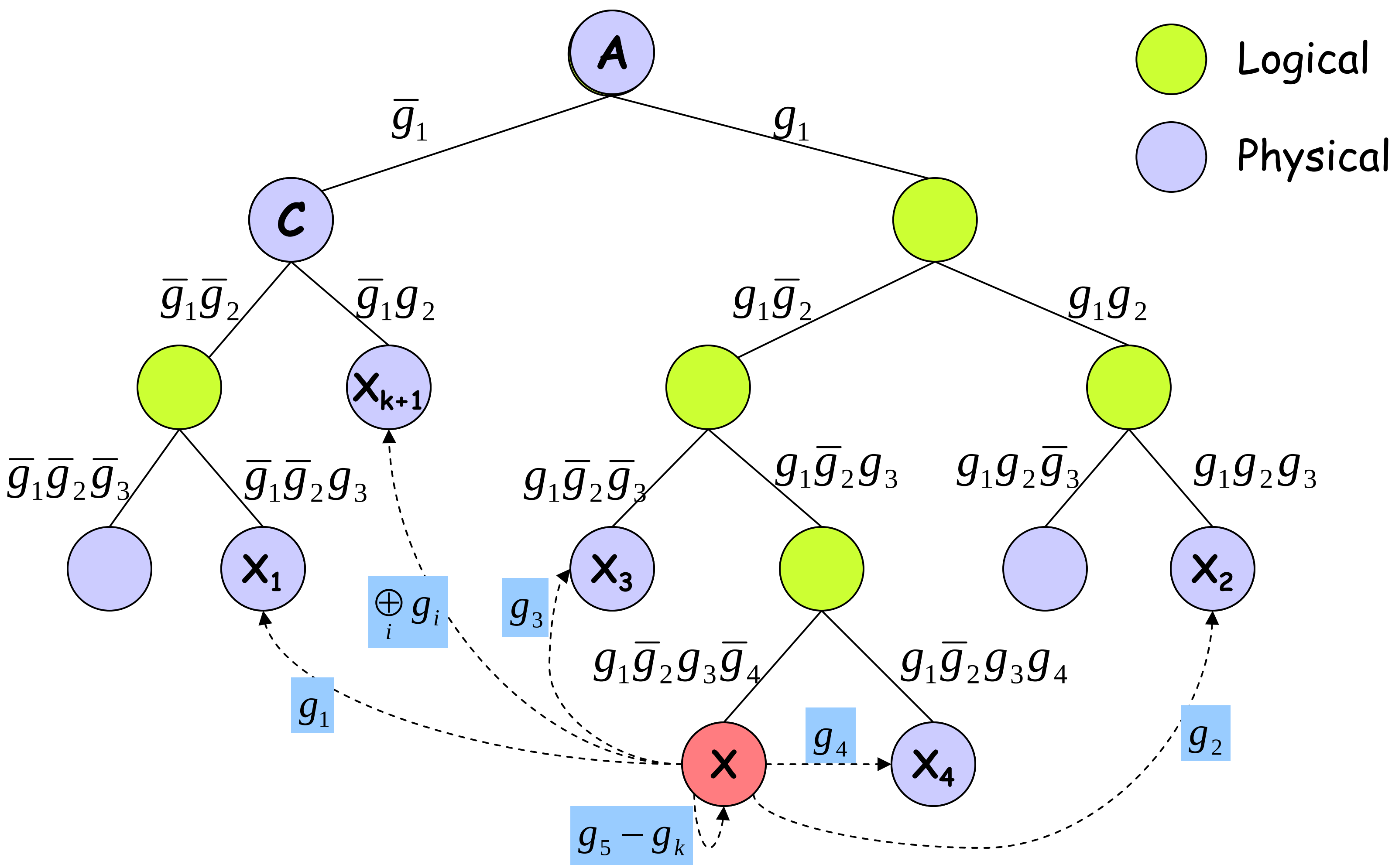
- Example: Route from X to Y where,

$$Y = \underbrace{\underbrace{\underbrace{X \oplus g_2}_{X_2} \oplus g_3}_{X_{23}} \oplus g_5}_{X_{235} = Y}$$

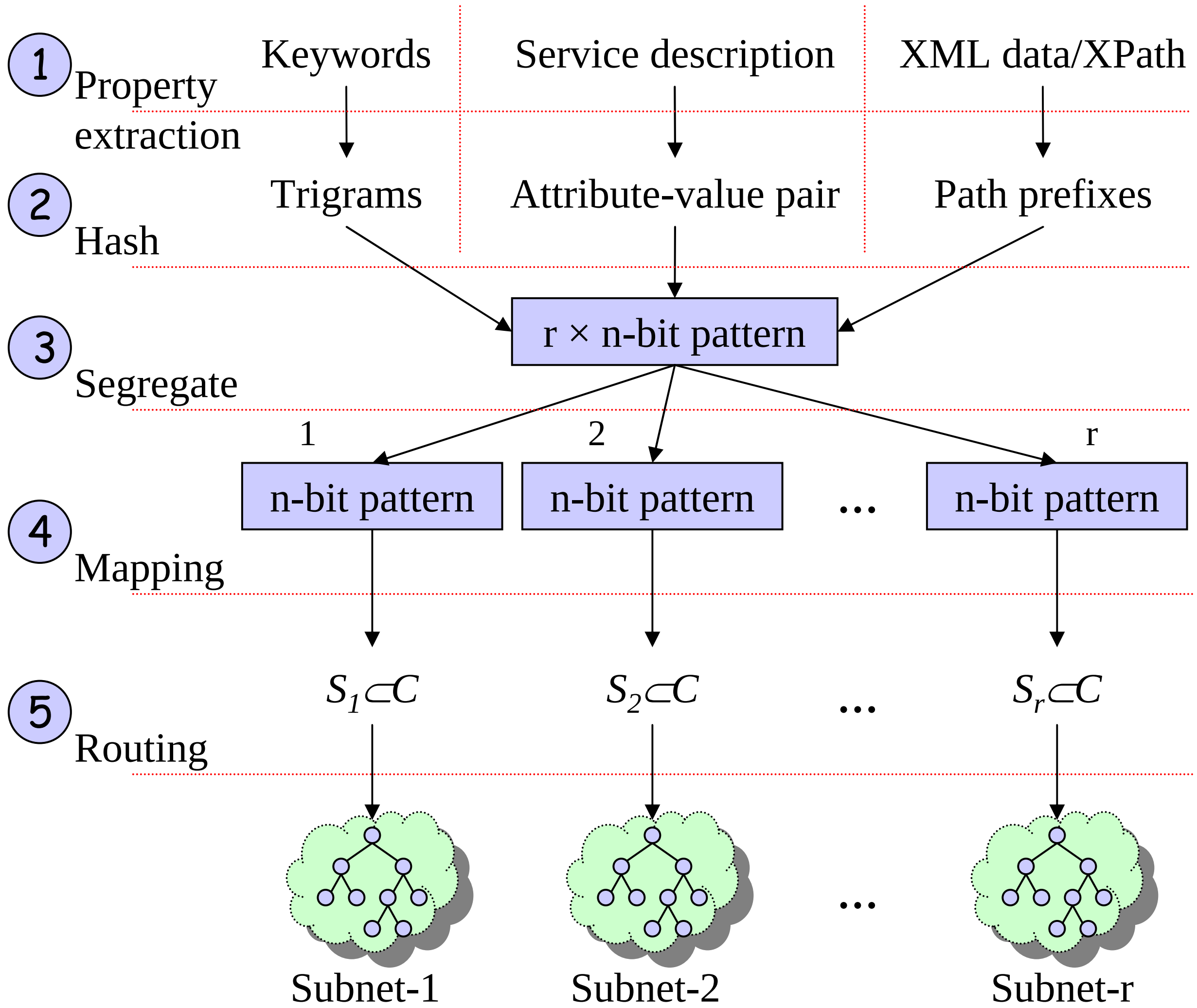


Plexus: Codeword Assignment

- Mapping codewords to peers in networks with less than 2^k peers.



Plexus: Multiple subnets

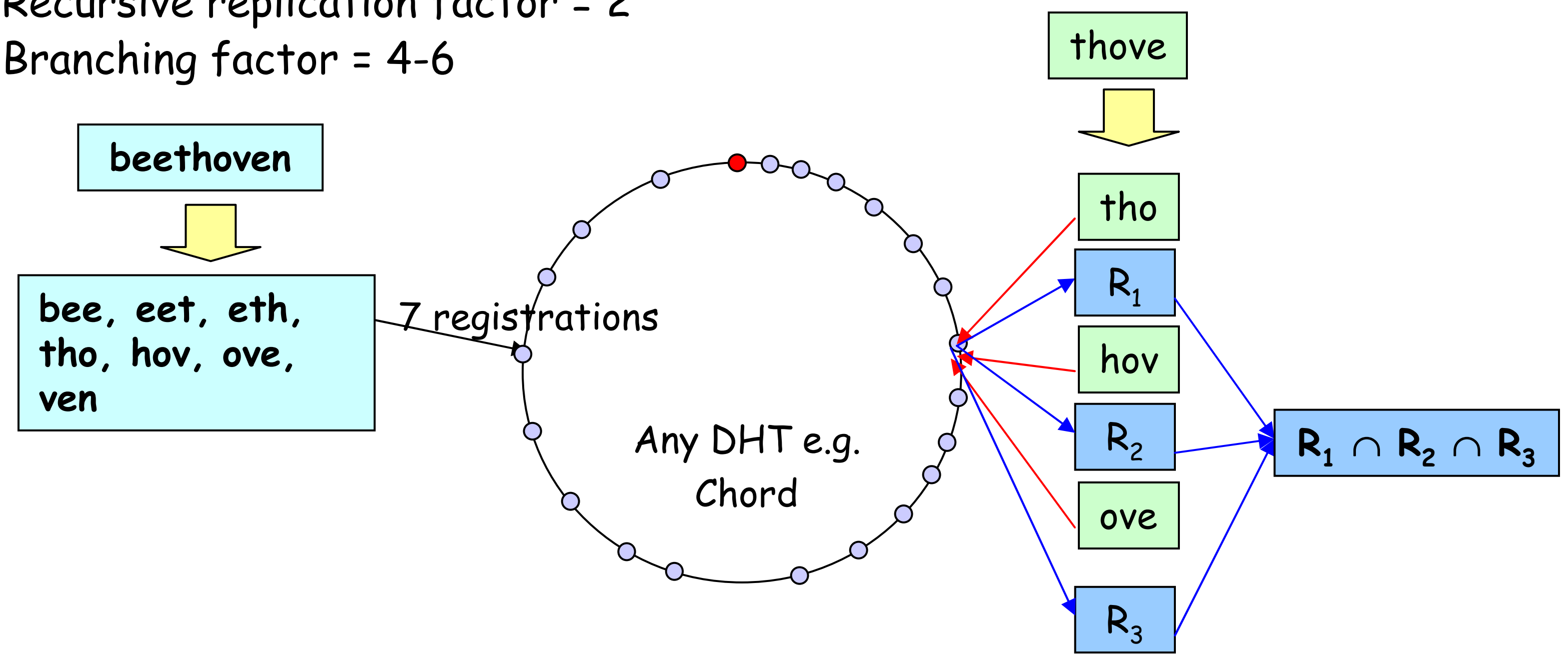


Outline

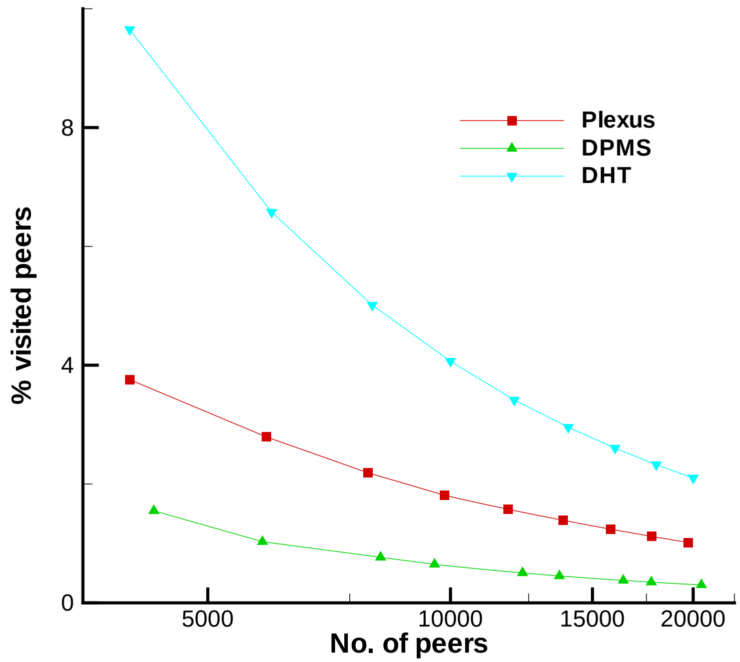
- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusion

Experimental Setup

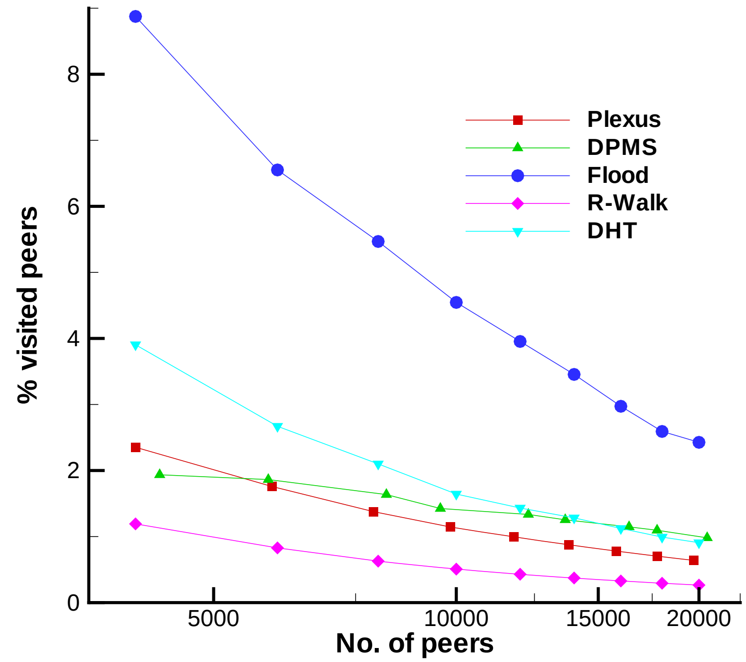
- Search systems
  - Flooding
    - Uniform replication with avg. 120
    - TTL = 4
  - Random walk
    - Uniform replication with avg. 120
    - Walker = 15
  - DPMS
    - Recursive replication factor = 2
    - Branching factor = 4-6
  - Plexus
    - No. of subnets = 7
  - DHT/Chord
    - Replica per key = 4



Routing Efficiency



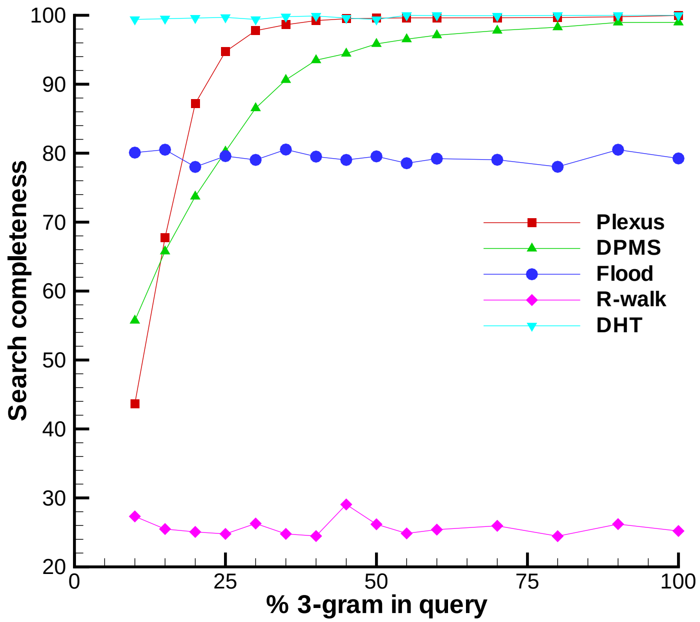
Advertisement traffic



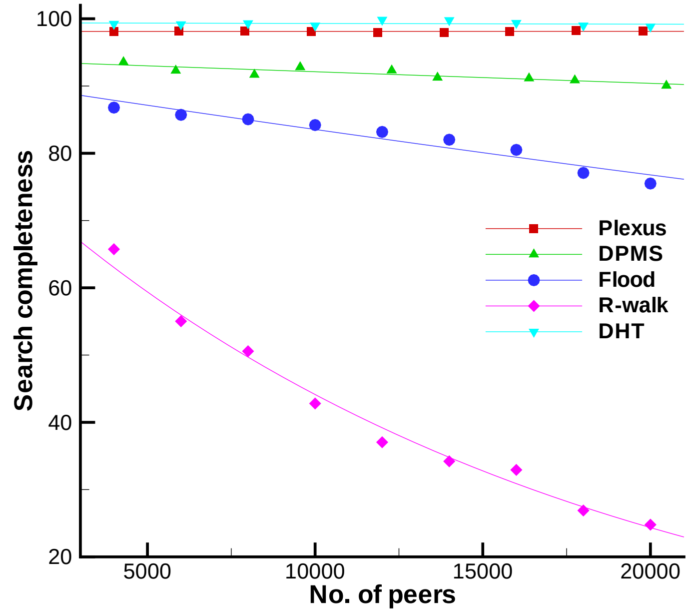
Search traffic

At % n-gram in query = 35%

Search Completeness

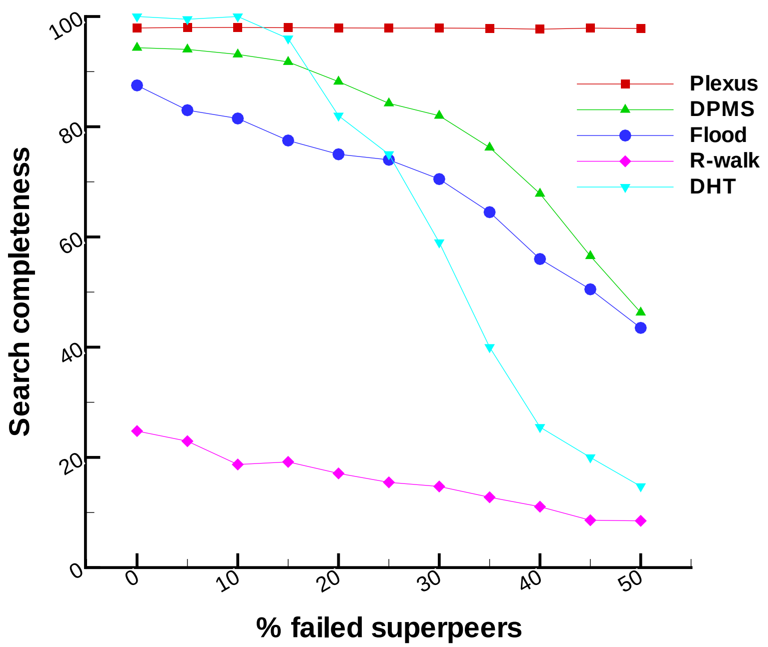


Search Completeness
Network size ≈ 20K

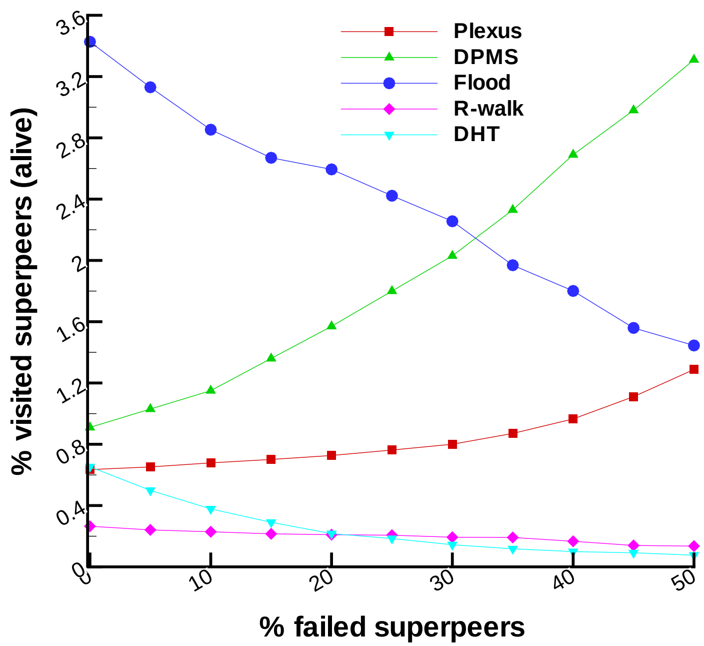


Search completeness
At % n-gram in query = 35%

Fault Resilience



Search completeness
With failed peers



Query traffic
With failed peers

Outline

- Advertisement and querying in LSDS
- Existing search mechanisms
- The DPM framework
- DPMS
- Plexus
- Experimental evaluation
- Conclusions

Conclusions

- We have formulated DPM, a new problem, which can be used to model search in a number of LSDS applications
  - We have shown how P2P search, Service discovery systems and P2P databases can be mapped to DPM.
- We have provided two solutions, DPMS and Plexus, which solve the DPM problem
  - Plexus surpasses all known search techniques in both structured and unstructured LSDSs.
  - We have demonstrated that it is possible to reconcile flexibility and efficiency.
- We believe that DPM has great potential in many existing and emerging applications
  - Examples include molecular databases, fingerprinting, phonetic search, sound alike search, etc.