



# PERSISTENT NAMING FOR P2P WEB HOSTING

Presented By

**Md. Faizul Bari**

PhD Student

David R. Cheriton School of Computer Science  
University of Waterloo

# Web Hosting on P2P Networks

- Problems with client/server architecture:
  - Flash crowds
  - Poor scalability even with high end servers and geographically distributed CDNs
  - Human intervention (DNS redirection)
  - Administrative overhead
  - Hosting expenses

# Web Hosting on P2P Networks

- Advantages of P2P web hosting
  - No single point-of-failure
  - Self-CHOP (configuring, healing, organizing, protection)
  - In-network caching improves performance
  - Lower cost
  - Freedom of speech
  - Publisher anonymity

# Research Challenges for Web Hosting on P2P Networks

- Highly dynamic network structure
- Content dynamism
- Content placement
- No uptime guarantees
- No end-to-end trust framework
- Firewalls and NATs
- ...

only to name a few



Our objective:

Provide a persistent naming  
scheme for web hosting on  
P2P networks

# So What are the Research Challenges for Naming?

- Location and time independent naming
  - Internet: URLs are bound to particular hosts
  - P2P: Any peer with a valid copy can be a source
- Distributed name registration and resolution
  - DNS is not a suitable solution in the P2P context

# So What are the Research Challenges for Naming?

- Names must be attached to content
  - Independent of peer
- Flexible and human friendly names
- Persistent hyperlinks or bookmarks

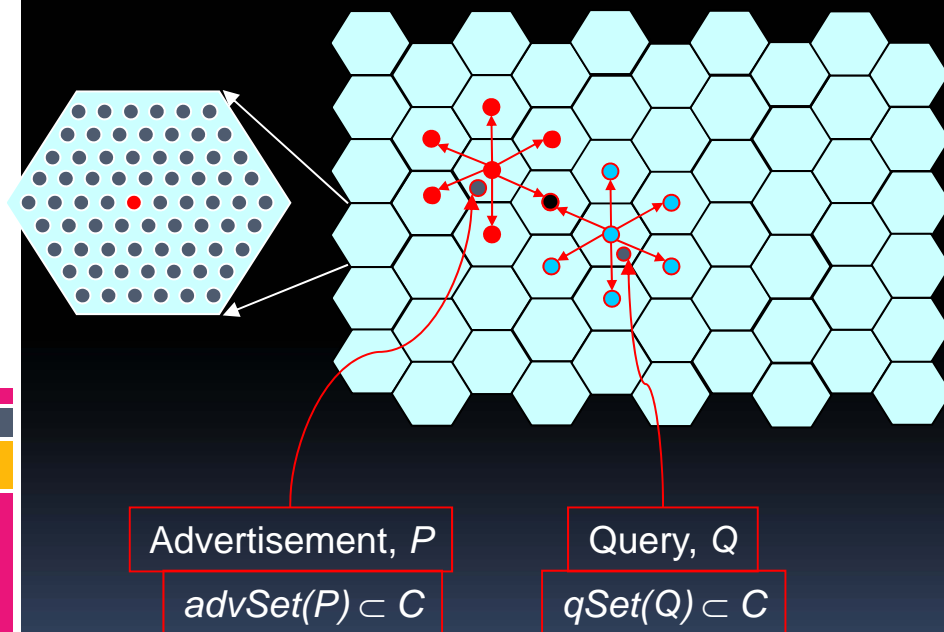
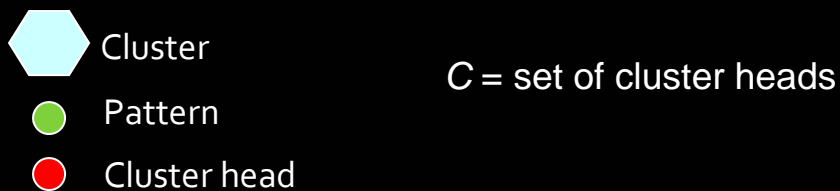


# Outline

- Plexus Routing
- pWeb Architecture
- Naming Scheme
- Experimental Results
- Conclusion



# Plexus: Index Clustering



Linear code,  $C \langle n, k, d \rangle$

Cluster head  $\leftrightarrow$  Codeword

Generator matrix based routing

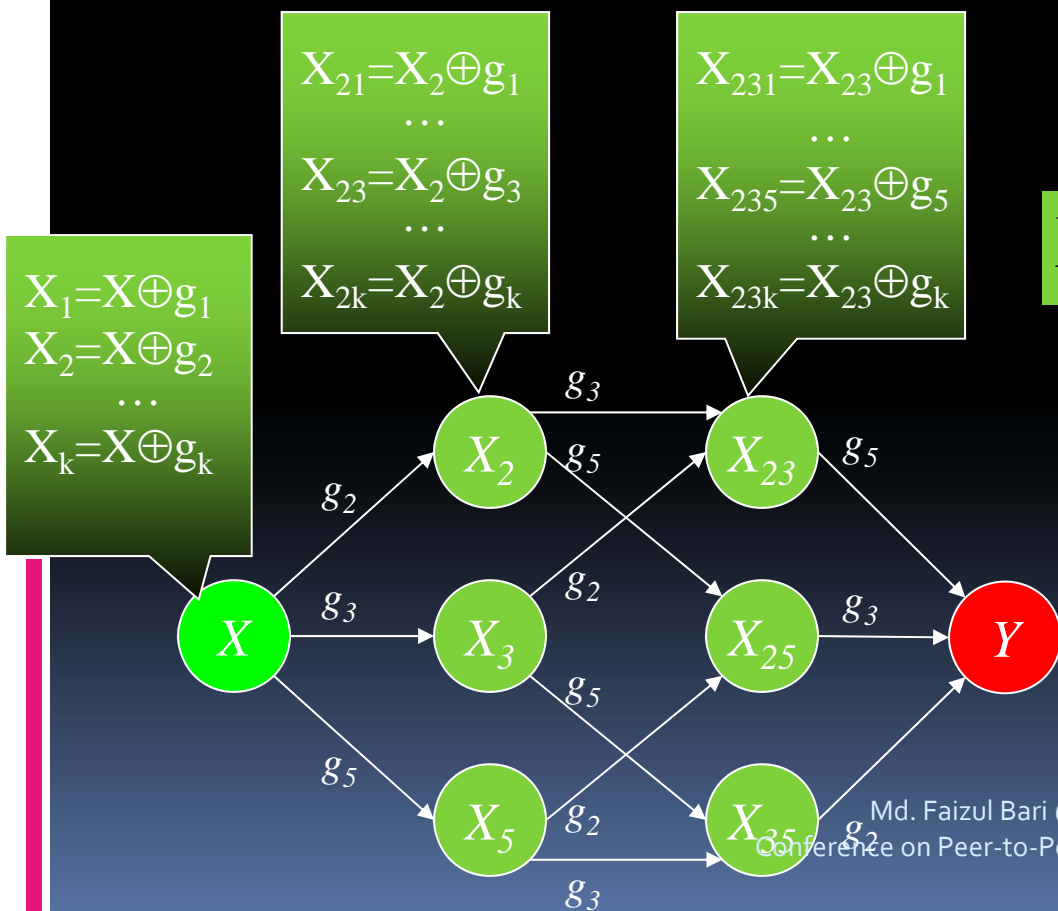
$$G = \begin{bmatrix} 47 \\ 23 \\ 15 \\ 0E \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$\langle 7, 4, 3 \rangle$  Hamming code

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

# Plexus: Routing

- Observation:  $C$  is closed under  $\oplus$  operation



Example: Route from  $X$  to  $Y$  where,

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

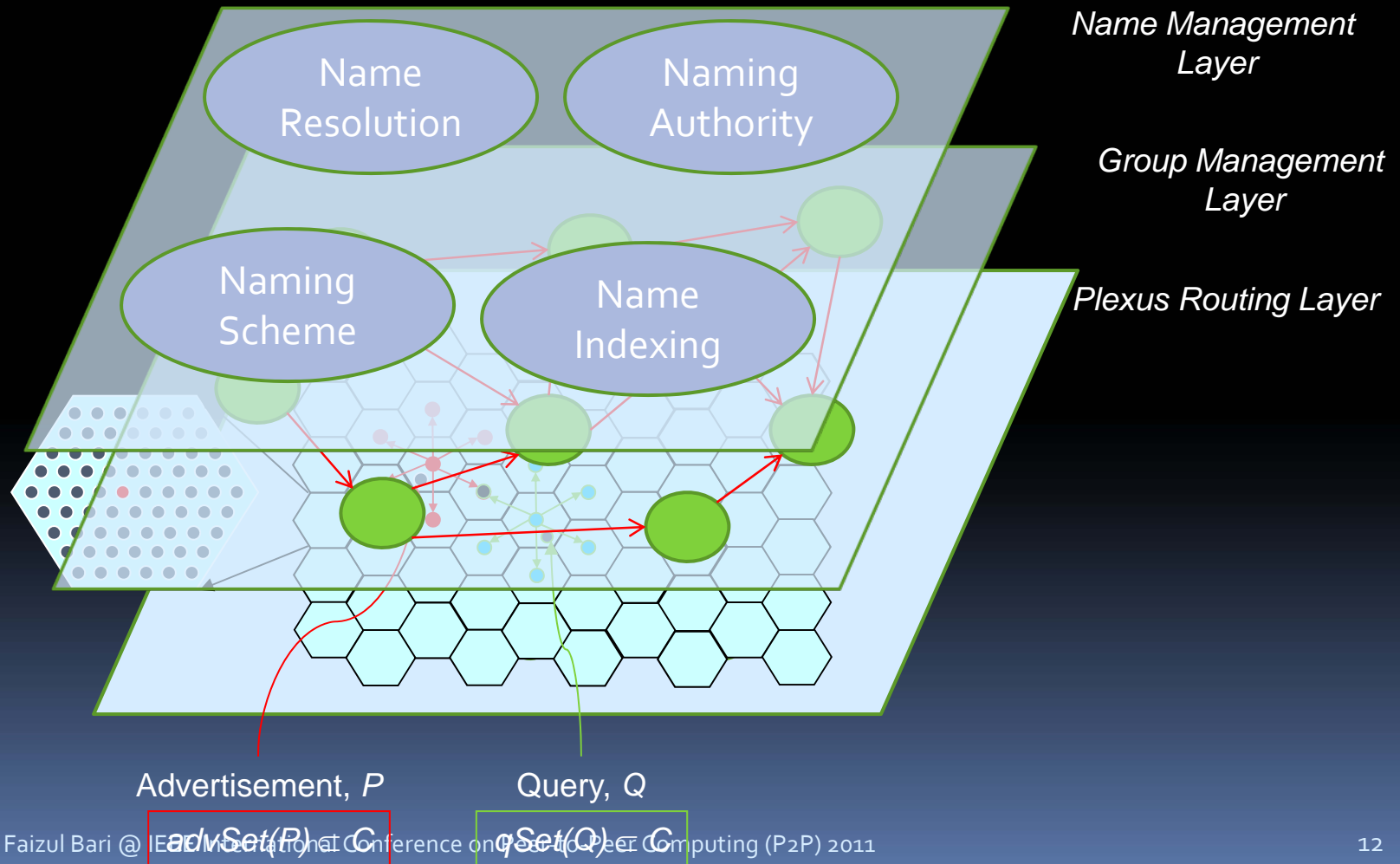
$$X_{k+1} = X \oplus g_1 \oplus g_2 \oplus \dots \oplus g_k$$



# Outline

- Plexus Routing
- pWeb Architecture
- Naming Scheme
- Experimental Results
- Conclusion

# pWeb Architecture



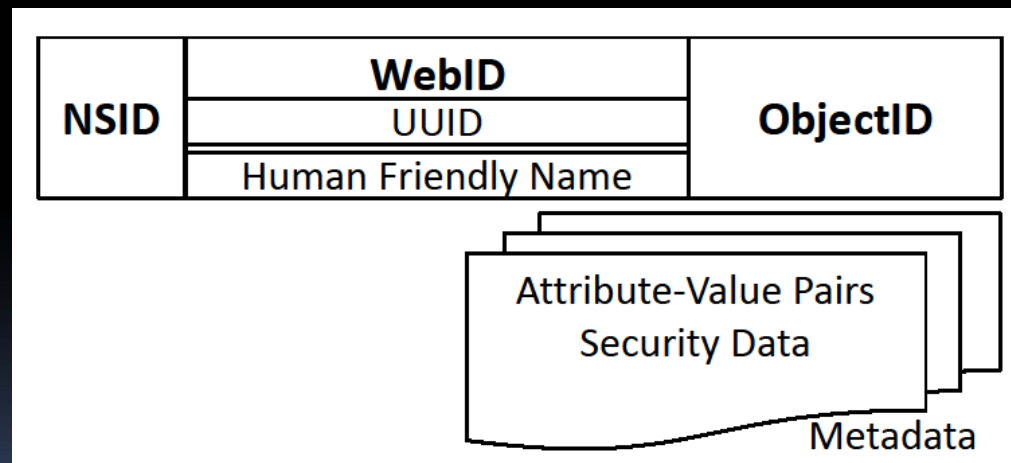


# Outline

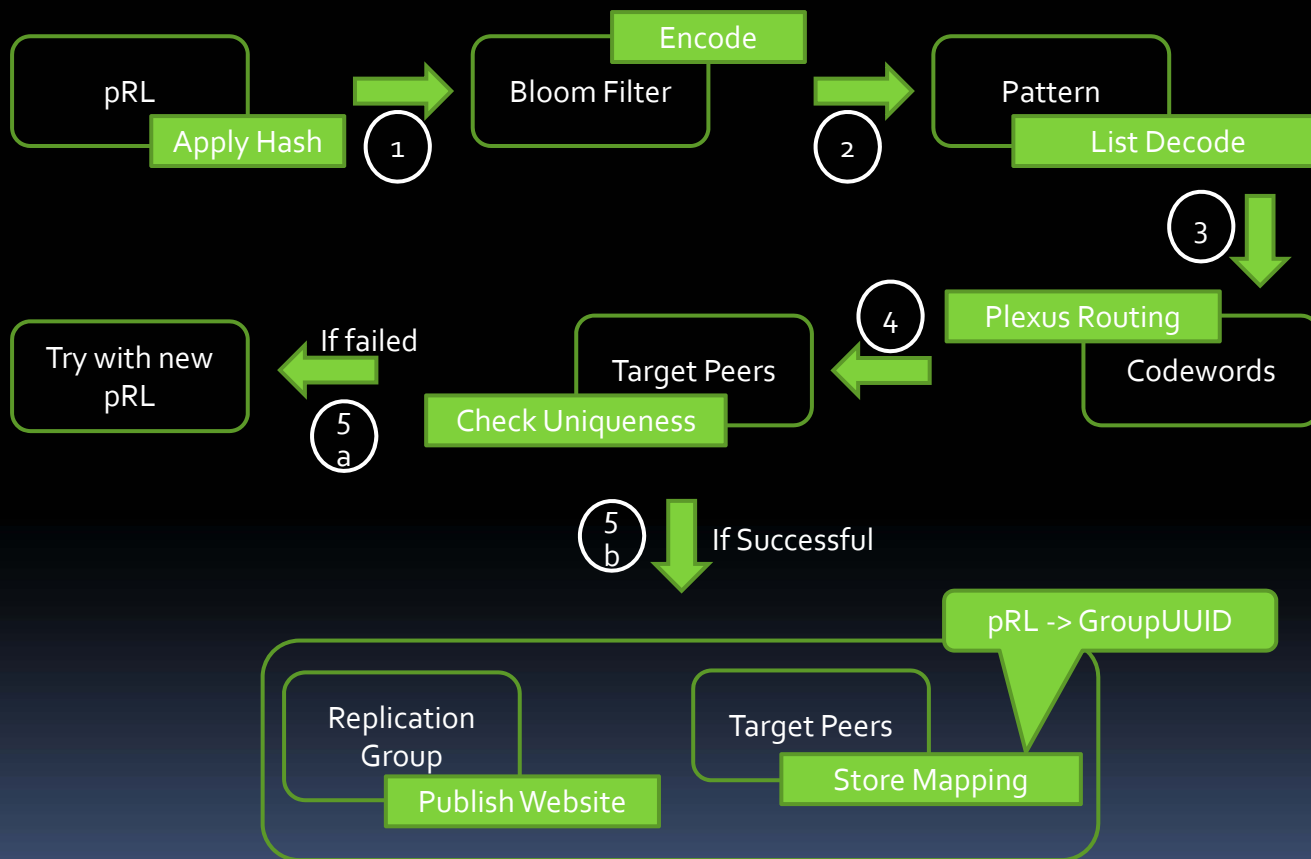
- Plexus Routing
- pWeb Architecture
- Naming Scheme
- Experimental Results
- Conclusion

# Naming Scheme -> Name Structure

- Names are called pRL
  - pWeb Resource Locator

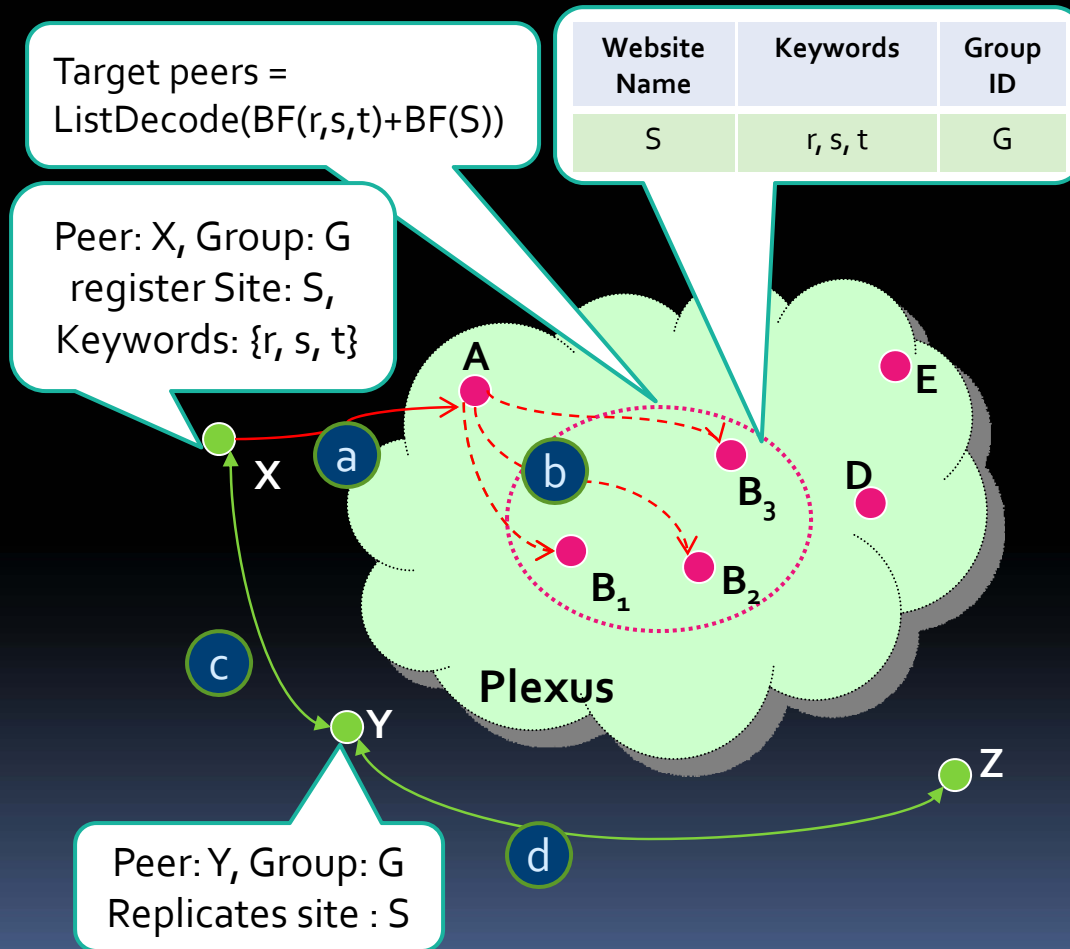


# Naming Scheme -> Name Registration



- Apply Hash
- Encode
- List Decode
- Route
- Check Uniqueness
- Publish

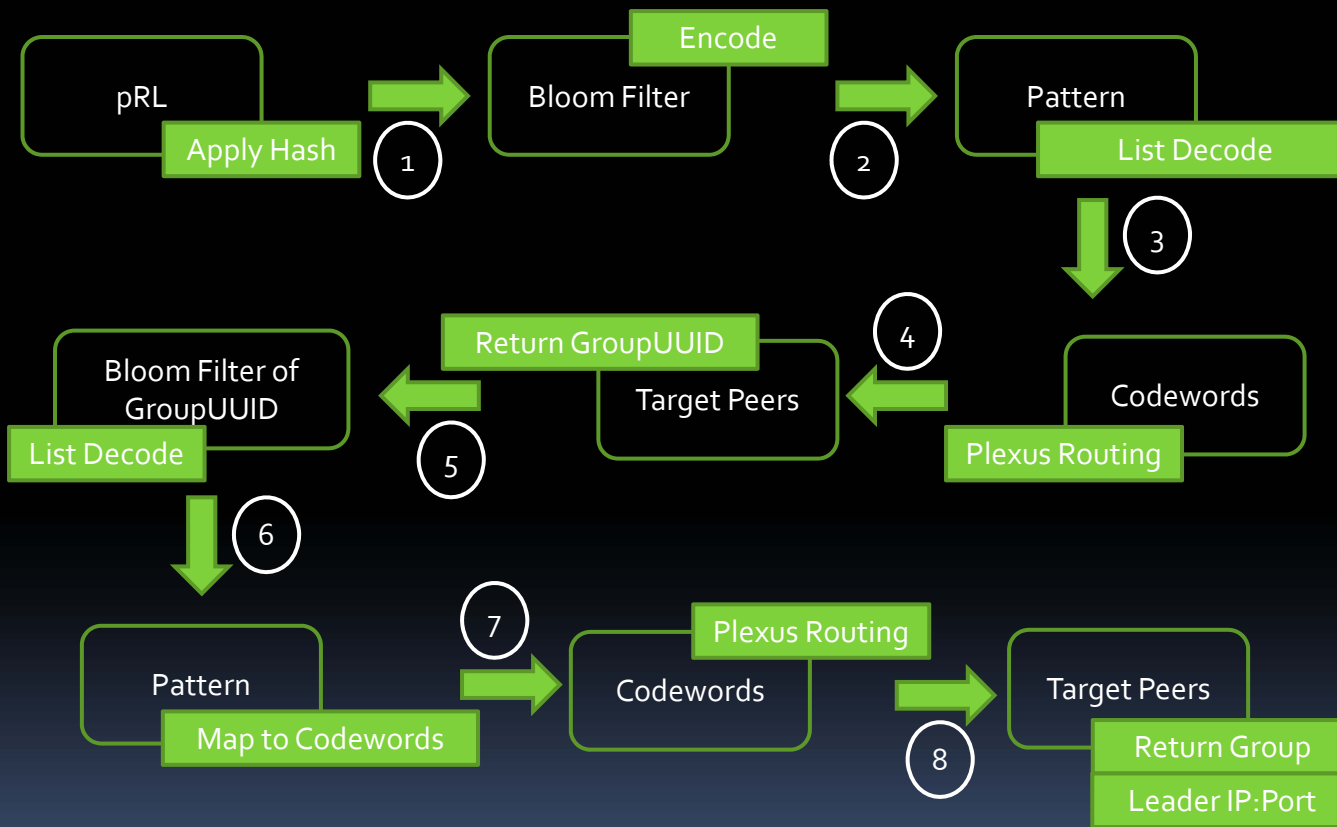
# Naming Scheme -> Name Registration



- Apply Hash
- Encode
- List Decode
- Route
- Check Uniqueness
- Publish
- Replicate

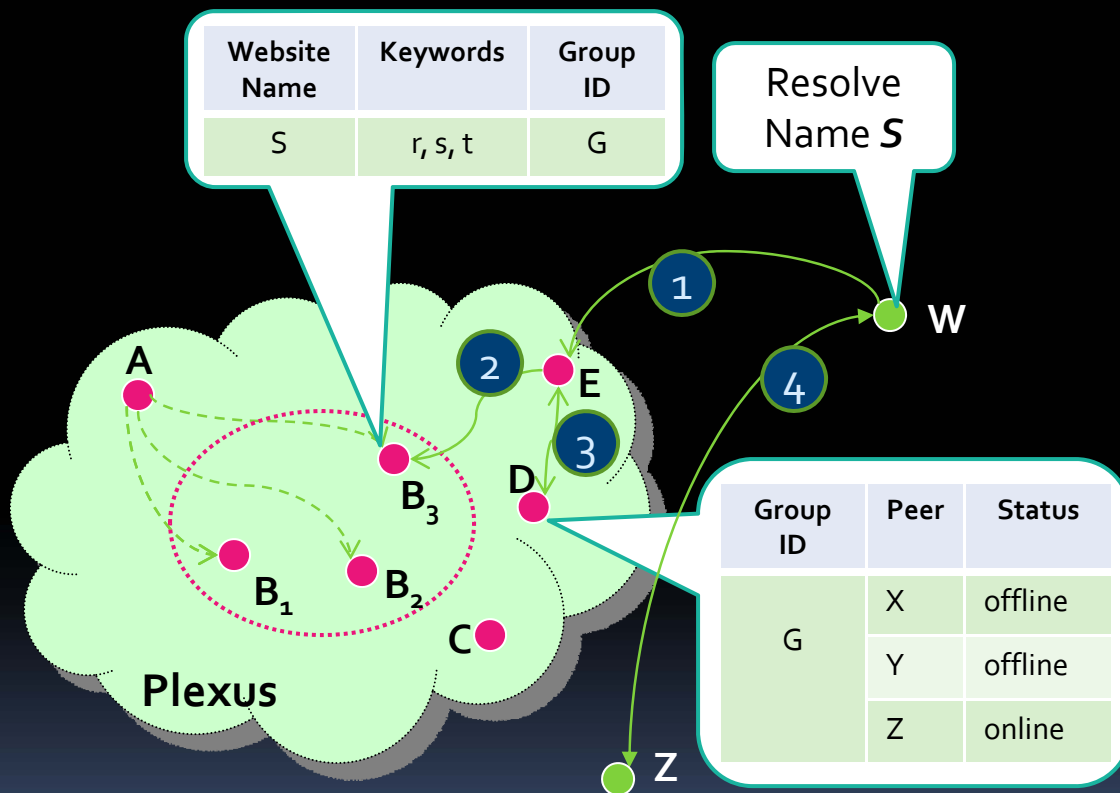


# Naming Scheme -> Name Resolution



- Apply Hash
- Encode
- List Decode
- Route
- Return GroupUUID
- List Decode
- Map to Codewords
- Route
- Group Leader's IP:port

# Naming Scheme -> Name Resolution



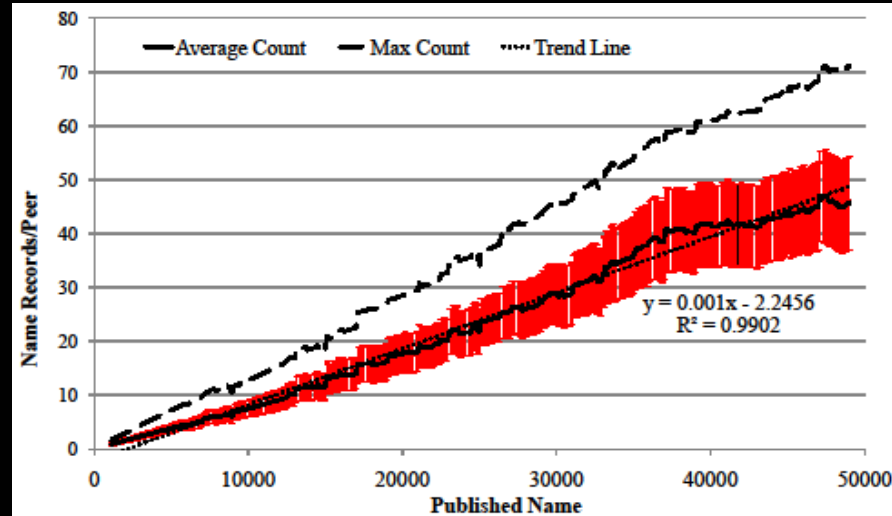
- Apply Hash
- Encode
- List Decode
- Route
- Return GroupUUID
- List Decode
- Map to Codewords
- Route
- Group Leader's IP:port



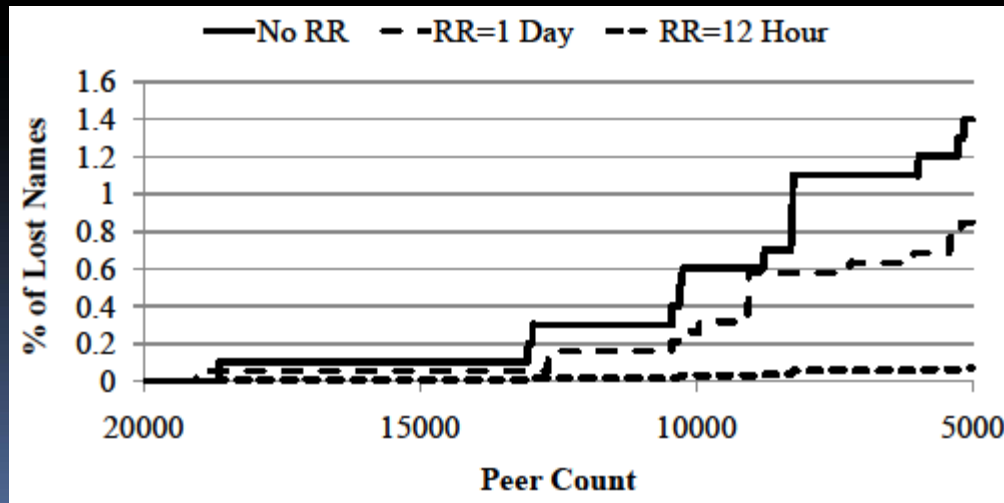
# Outline

- Plexus Routing
- pWeb Architecture
- Naming Scheme
- Experimental Results
- Conclusion

# Experimental Results

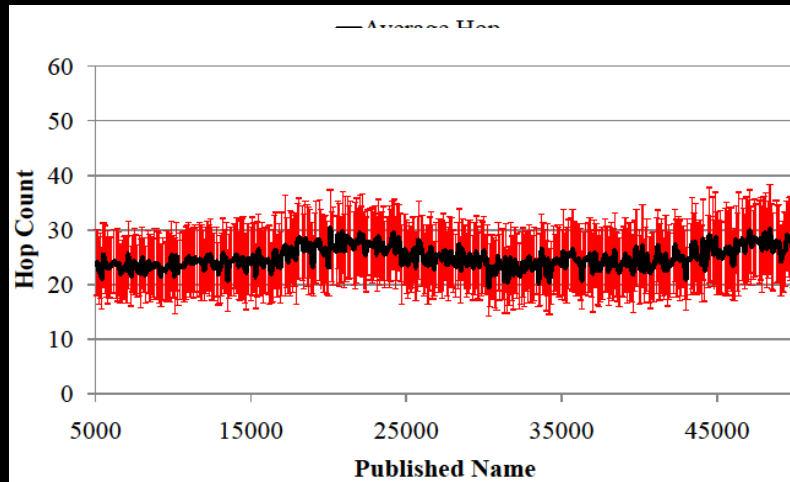


(a) Name record count

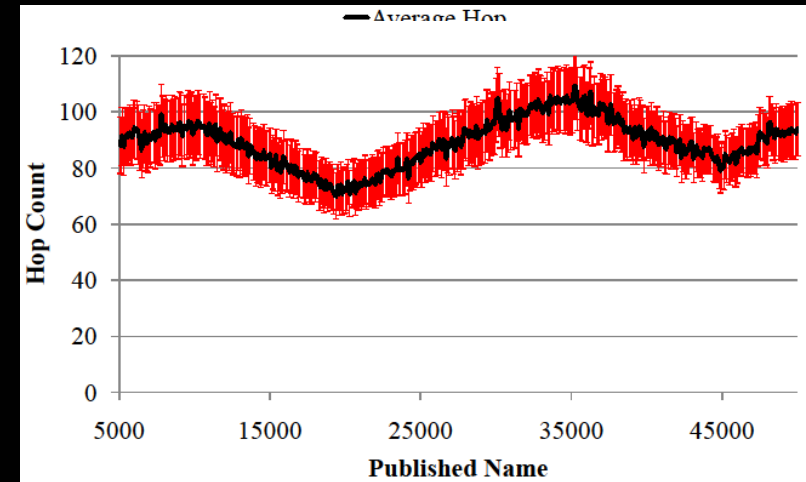


(b) Percentage of lost names

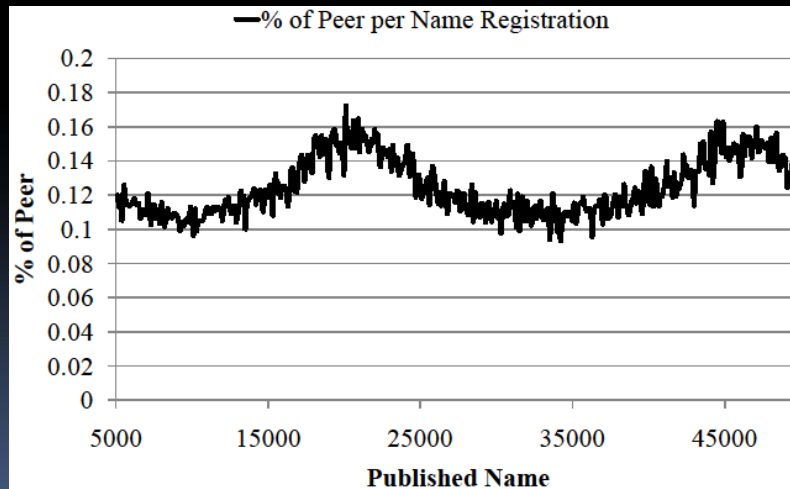
# Experimental Results



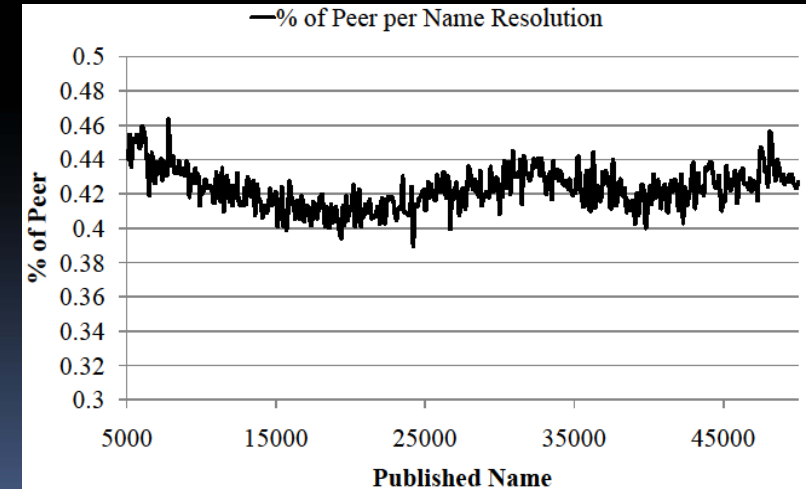
(a) Worst case name registration hop count



(b) Worst case name resolution hop count



(c) % of peer accessed/registration



(d) % of peer accessed/resolution



# Outline

- Plexus Routing
- pWeb Architecture
- Naming Scheme
- Experimental Results
- Conclusion

# Related Work

- Information Centric Networks
  - NetInf
  - DONA
  - CCN
- P2P Networks
  - BitTorrent: Hash of file chunk

# Summary

- We have proposed a naming scheme
  - That is
    - Distributed
    - Persistent
    - Scalable and
    - Fault-tolerant
- It provides a flat namespace with support for both Human friendly and secure distribute names



