Engineering Research Center for
Collaborative Adaptive Sensing of the Atmosphere

Ph.D. Dissertation – Fall 2012

Enhancing Collaborative Peer-to-Peer Systems Using Resource Aggregation & Caching:
A Multi-Attribute Resource & Query Aware Approach

Graduate Committee

- Prof. Anura P. Jayasumana (Advisor)
- Prof. V. Chandrasekar
- Prof. Daniel F. Massey
- Prof. Indrajit Ray

Dilum Bandara
dilumb@engr.colostate.edu

University of Massachusetts Amherst

University of Oklahoma

Colorado State University

University of Puerto Rico Mayaguez

CASA is primarily supported by the Engineering Research Centers Program of the National Science Foundation under NSF award number 0313747.

Contributions

Propose a peer-to-peer based approach to enable the collaboration of group of heterogeneous, dynamic, & distributed resources in a scalable & efficient manner

Developed resource discovery, caching, & distributed data fusion solutions that are more suitable for collaborative P2P systems by characterizing real-world resource, query, & user behavior

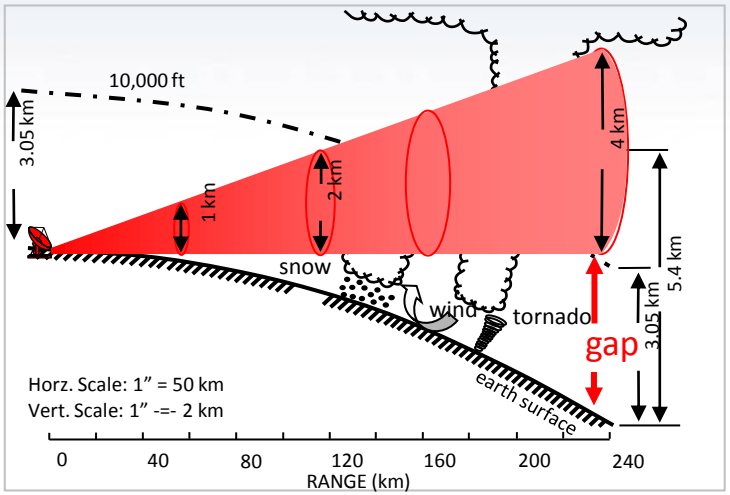
Outline

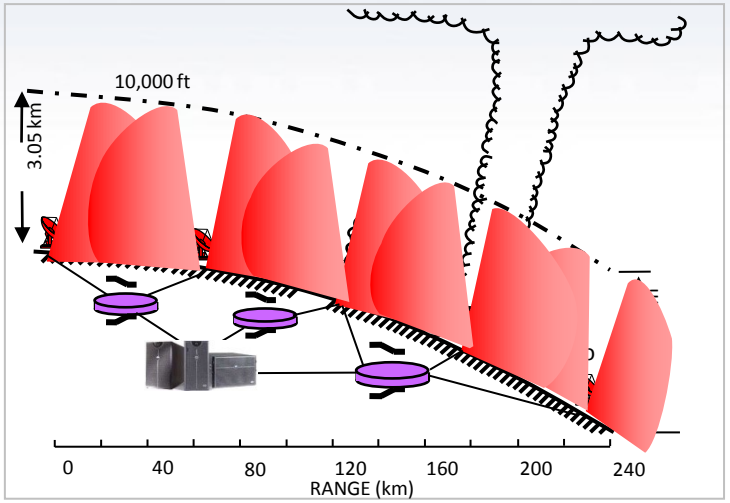
- Motivation
- Problem statement
- Multi-attribute resource & query characteristics
- Resource & query aware resource discovery
- Multi-attribute resource & range query generation
- Community-based caching
- NDN for DCAS
- Summary & future work

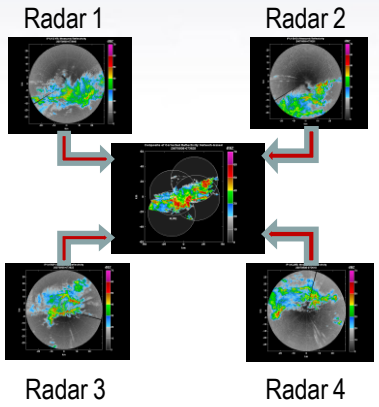
Collaborative Peer-to-Peer Systems

- Advances in Web 2.0, high-speed networks, cloud computing, & social networks
- P2P systems will play an even greater role in distributed resource collaboration
- Diverse peers bring in unique resources & capabilities to a virtual community to accomplish something big
- Scalable alternative to Distributed Collaborative Adaptive Sensing (DCAS), Internet of Things, cloud & opportunistic computing, etc.

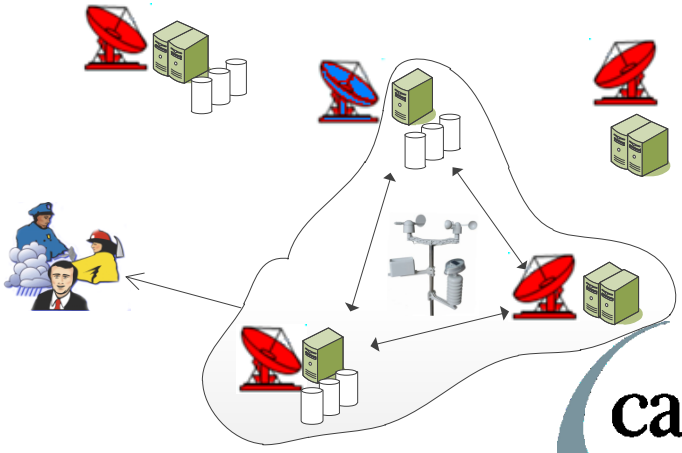
Collaborative Adaptive Sensing of the Atmosphere (CASA)



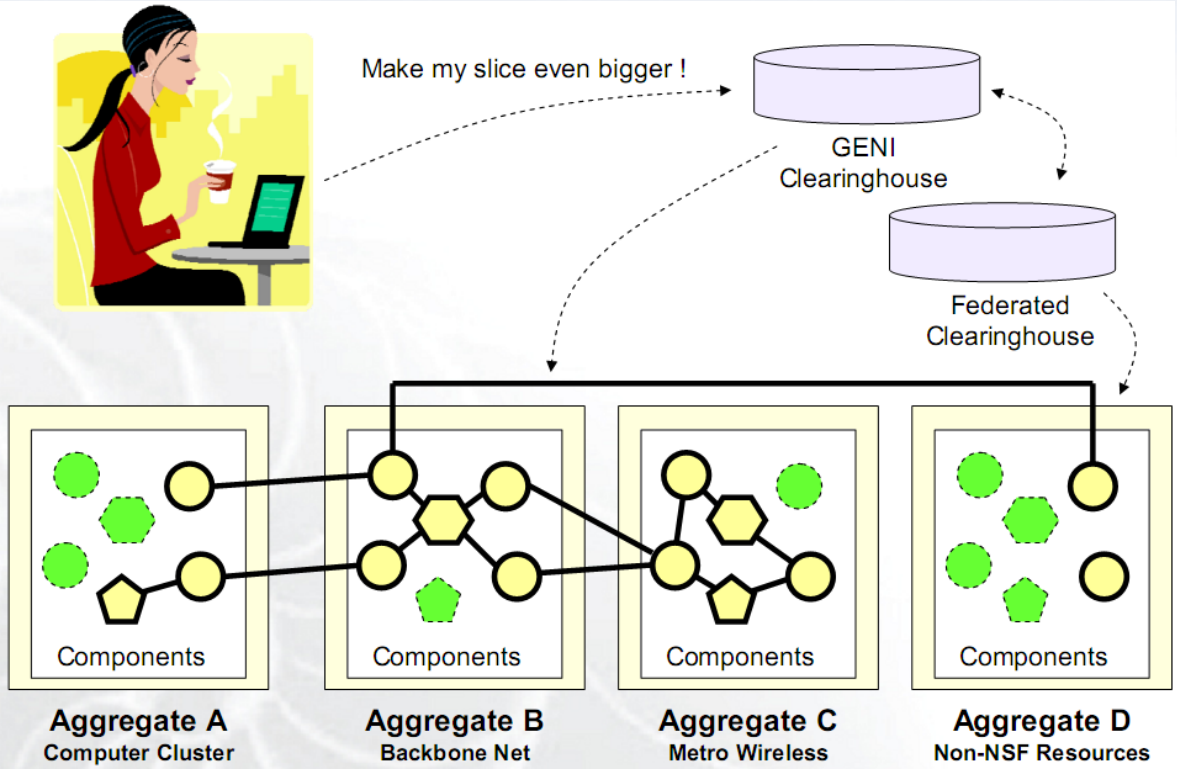




- Distributed Collaborative Adaptive Sensing (DCAS) system
- CASA aggregates groups of resources as & when needed
  - 10,000 radars to cover U.S.
  - High data rate, real-time, heterogeneous, multi-attribute, dynamic, & distributed
  - Dedicated & reliable resources



Global Environment for Network Innovations (GENI)

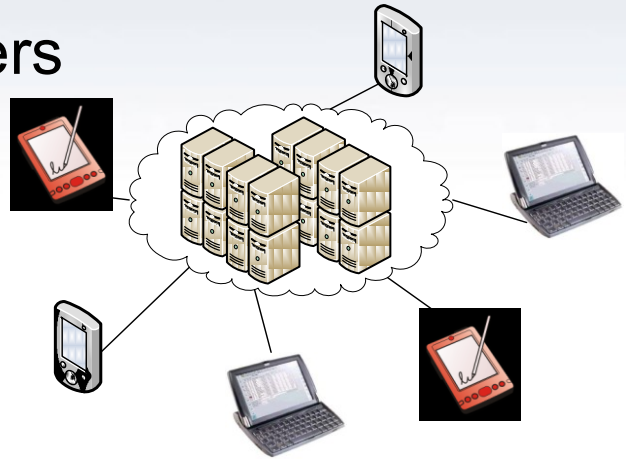


- Sensors
  - Cameras
  - Sensors mounted on busses
  - Micro weather stations
  - Radars
- Processing & storage
  - Amazon EC2
  - Amazon S3
- Networks
  - Internet2
  - Emulab
  - BEN dark fibers

- Collaborative & exploratory platform for innovation
- Aggregating groups of resources across multiple administrative domains
  - Dedicated & reliable resources

Community (P2P) Cloud Computing

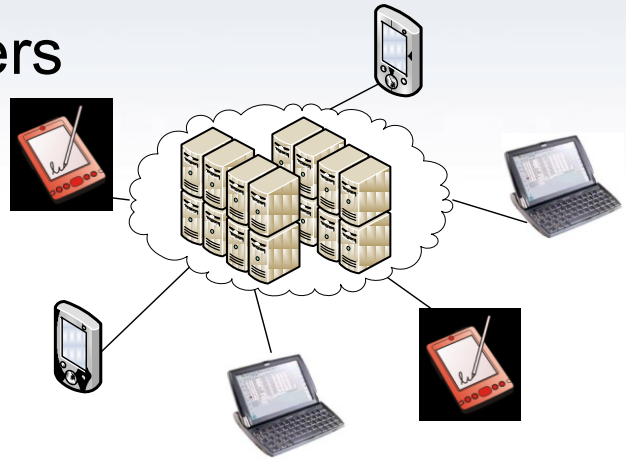
- Resource aggregation within datacenters
  - Data intensive cloud computing
  - Encryption, business logic, & scientific algorithms
  - Storage, GPUs, FPGAs
  - Virtual networks in/out & within cloud
    - Sensors can't be inside a datacenter



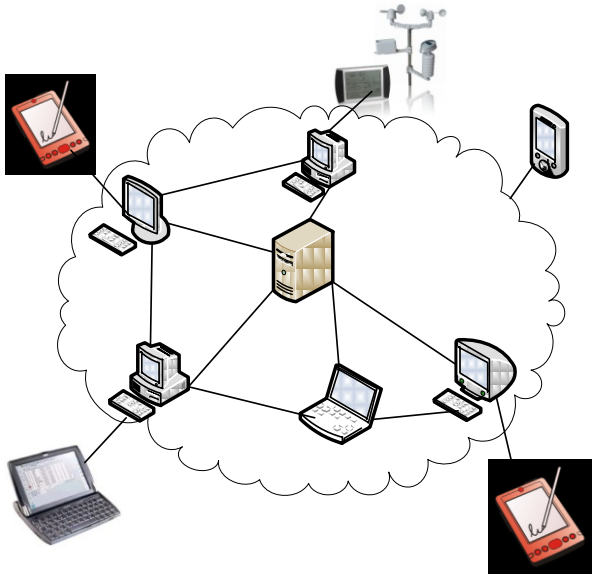
- Community as a datacenter
  - Resourceful peers & home servers
  - Aggregation of bandwidth at edge
  - Users govern themselves & hold data
  - Monetary & non-monetary benefits
  - Voluntary & unreliable resources

Community (P2P) Cloud Computing

- Resource aggregation within datacenters
  - Data intensive cloud computing
  - Encryption, business logic, & scientific algorithms
  - Storage, GPUs, FPGAs
  - Virtual networks in/out & within cloud
    - Sensors can't be inside a datacenter



- Community as a datacenter
  - Resourceful peers & home servers
  - Aggregation of bandwidth at edge
  - Users govern themselves & hold data
  - Monetary & non-monetary benefits
  - Voluntary & unreliable resources



Problem Statement

- Motivation
  - CASA, GENI, & cloud computing need to aggregate heterogeneous, multi-attribute, & dynamic groups of resources that are distributed
  - Very little is known about their characteristics
  - Existing solutions rely on many simplifying assumptions
    - Few attributes, i.i.d. attributes, attribute values ~uniform/Zipf's, large domains, very specific queries, ignore dynamic attributes
- Goal
  - Develop better resource discovery & distributed data fusion solutions & necessary tools, while characterizing real-world resources, queries, & user behavior
  - Empower peers to engage in greater tasks beyond capabilities of individual peers
    - Enhanced performance, efficiency, scalability, & resource utilization

Outcomes

1. Detailed analysis of real-world resource, query, & user characteristics, & their impact on P2P-based resource discovery – CCNC ‘12 [6], AICCSA ‘11 [7], [4], [12]
2. Resource & query aware multi-attribute resource discovery solution – LCN ‘12 [3]
3. Tool to generate large synthetic traces of multi-attribute resources & range queries – GLOBECOM ’11 [8], [13]

4. Community-based caching solution for large P2P systems – TPDC [1], ICC ‘11 [9]

5. Demonstrated applicability of Named Data Networking (NDN) for Distributed Collaborative Adaptive Sensing (DCAS) systems such as CASA – [10]

Resources & Queries

- Multi-attribute resources
  - Computing, storage, network, sensors, etc.
  - Static – CPU speed, no of CPU cores, Doppler radar, sensor range
  - Dynamic – Free CPU, memory, bandwidth, sensing frequency

$$r = (a_1 = v_1, a_2 = v_2, \ldots, a_i = v_i)$$

$$r = \begin{pmatrix} CPUSpeed = 2.4\ GHz, NumCores = 2, CPUArchi = \text{"}\times 86\text{"}, CPUFree = 53\% \\ MemoryFree = 1071MB, OS = \text{"}Linux_2.6.31\text{"}, \text{Application} = \text{"}NS\ 3.01\text{"} \end{pmatrix}$$

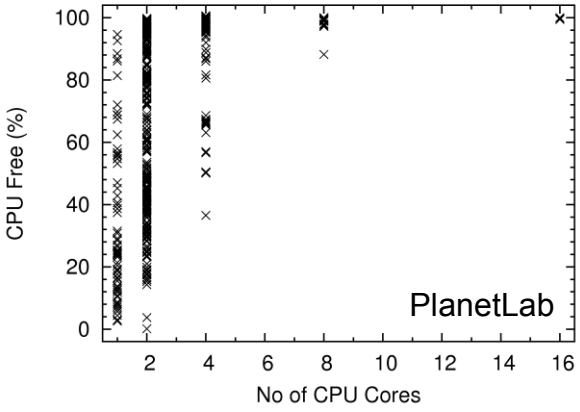
- Multi-attribute range queries

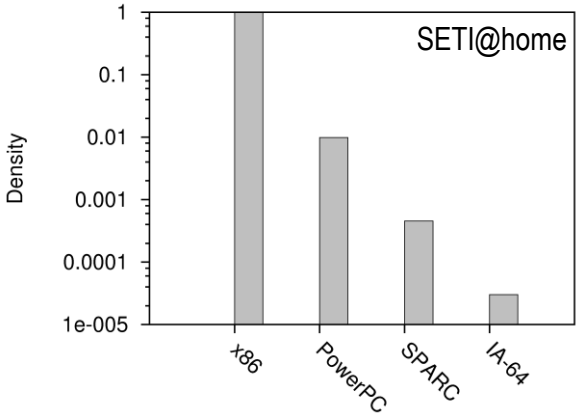
$$q = (n, a_1 \in [l_1, u_1], a_2 \in [l_2, u_2], \ldots, a_i \in [l_i, u_i])$$

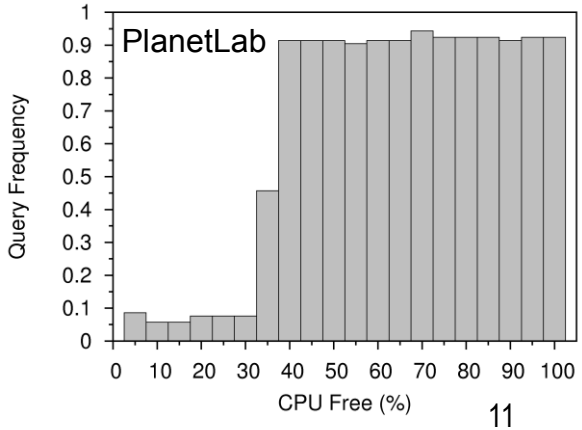
$$q = \begin{pmatrix} n = 5, CPUSpeed \in [2.0GHz, MAX], \\ MemoryFree \in [256MB, 512MB], OS = \text{"}Linux_2.6.32\text{"} \end{pmatrix}$$

Multi-Attribute Resource & Query Characteristics [7, 12]

- Datasets – PlanetLab, SETI@home, GCO grid, & CSU
- Real-world resource & query characteristics diverge substantially from conventional assumptions
  - Few attributes → Many attributes
  - i.i.d. → Complex correlation patterns
  - Uniform/Zipf's → Different marginal distributions, highly skewed
  - Large domains → Small domains for some attributes
  - Ignore dynamic attributes → Most popular, change rapidly
  - Very specific queries → Less specific queries

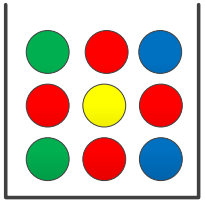




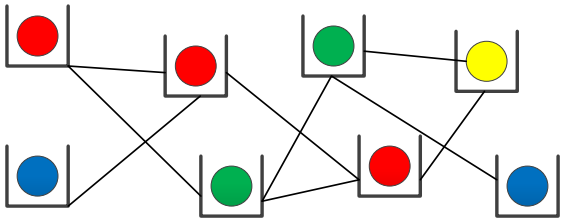


How These Characteristics Will Affect Resource Discovery?

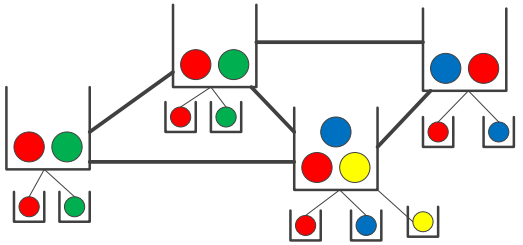
- Evaluate fundamental design choices for resource discovery
- Used node & query traces from PlanetLab



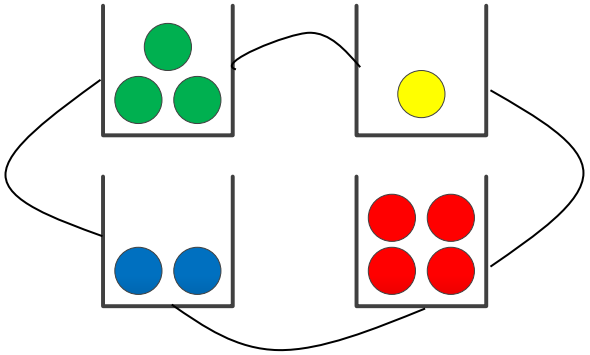
Centralized
$O(1)$



Unstructured P2P
$O(hops_{max})$



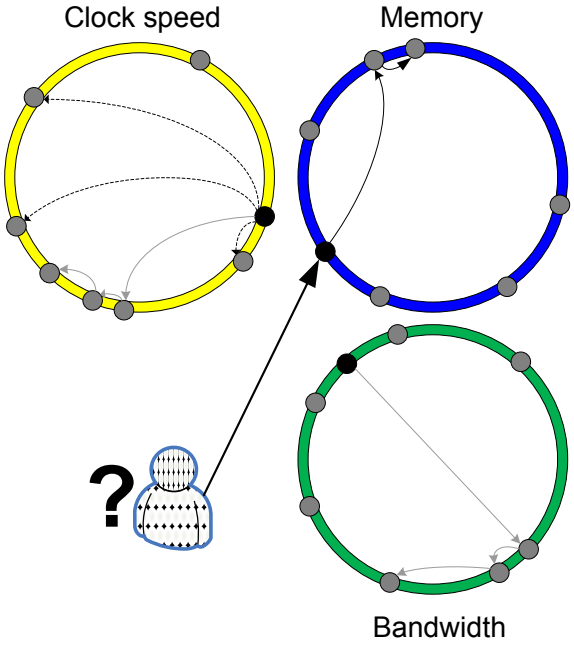
Superpeer
$O(hops_{max})$

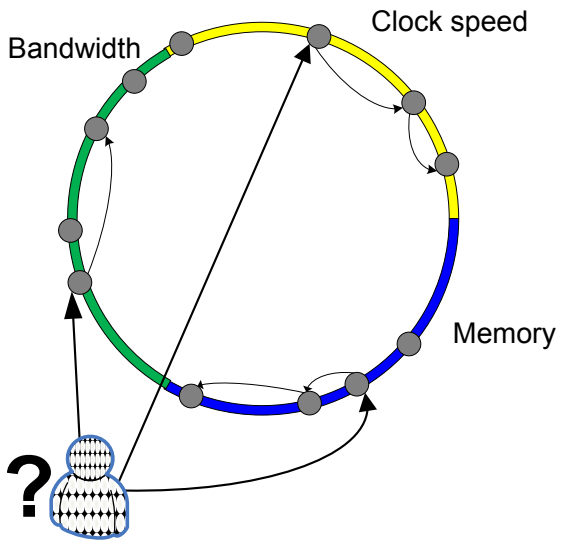


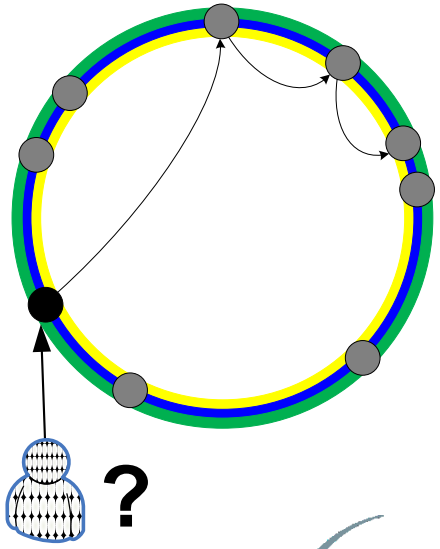
Structured P2P – Distributed Hash Table (DHT)
$O(\log N)$

How These Characteristics Will Affect Resource Discovery?

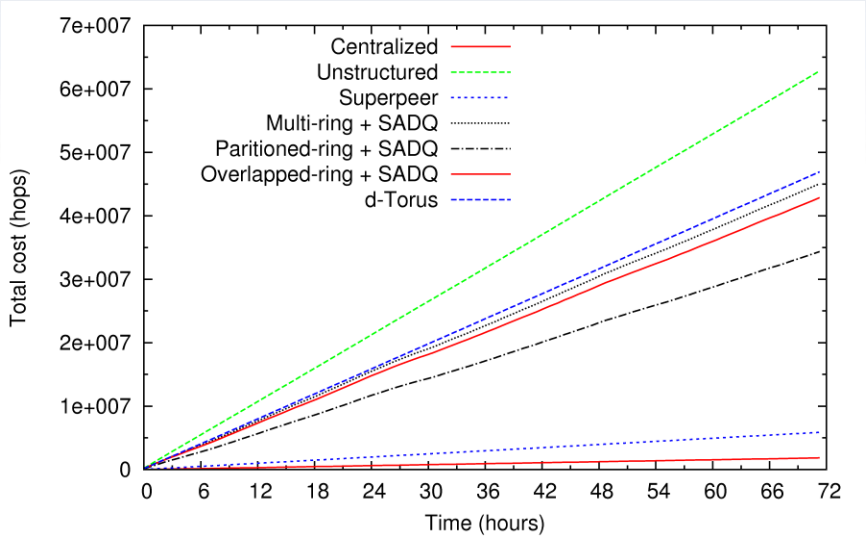
- Evaluate fundamental design choices for resource discovery
- Used node & query traces from PlanetLab

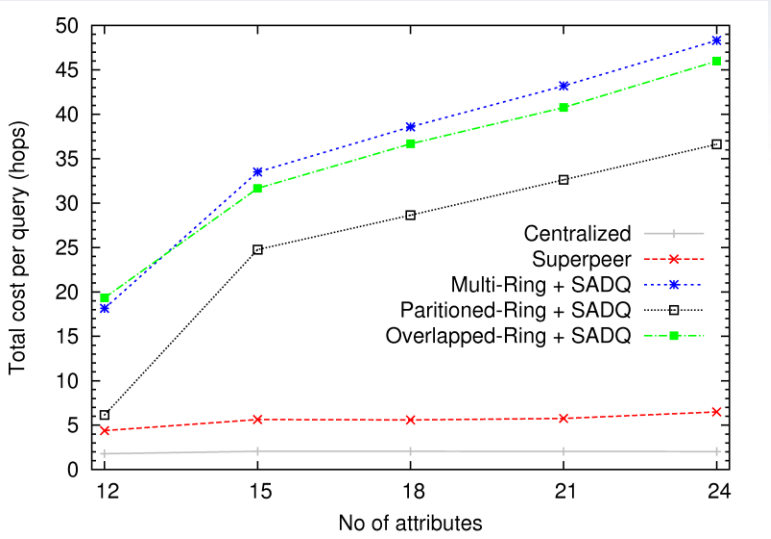




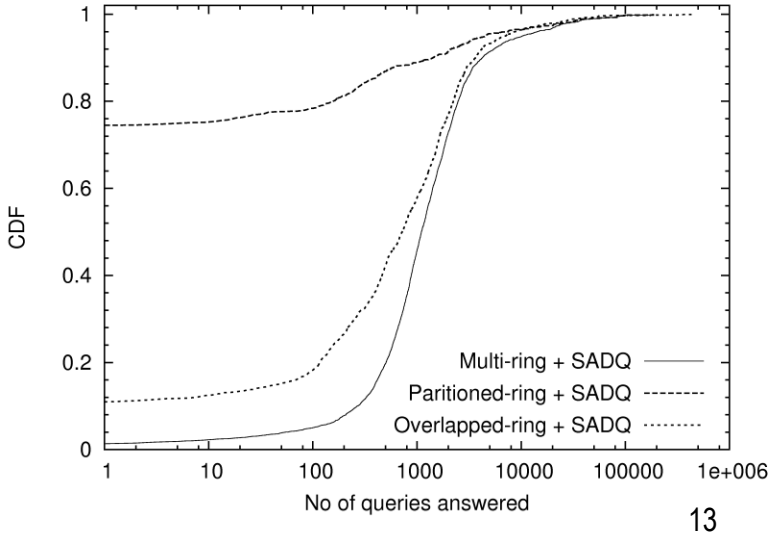


Design Choices for P2P-Based Resource Discovery – Performance Analysis [6, 12]





N	Multi-Ring + SADQ			Partitioned-Ring + SADQ			Overlapped-Ring + SADQ		
	Min	Ave	Max	Min	Ave	Max	Min	Ave	Max
250	0	9.2	239.1	0	3.7	19.4	0	9.1	238.4
527	0	13.7	509.0	0	4.6	27.6	0	13.5	506.0
750	0	16.2	719.1	0	4.9	36.6	0	16.5	719.9
1000	0	19.8	975.5	0	5.3	45.3	0	20.4	963.8



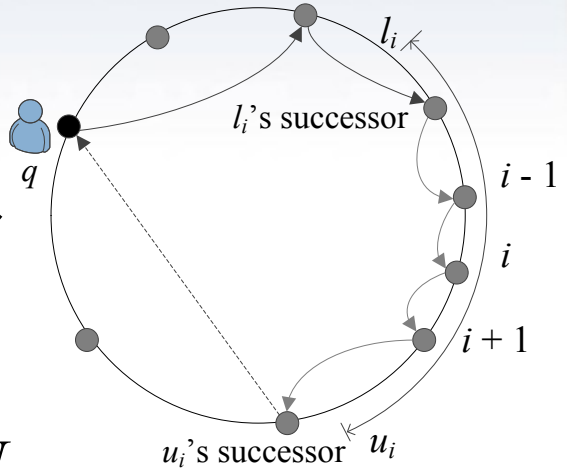
- Real-world queries are relatively easier to resolve
- Ring-based designs
  - Advertising & query cost – $O(A_{Dynamic})$ & $O(N)$
  - Load balancing problem

Outline

- Motivation
- Problem statement
- Multi-attribute resource & query characteristics
- Resource & query aware resource discovery
- Multi-attribute resource & range query generation
- Community-based caching
- NDN for DCAS
- Summary & future work

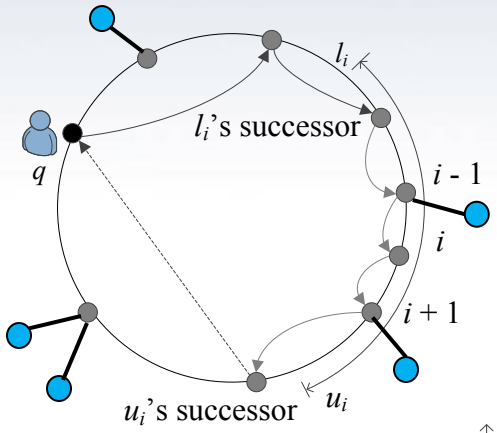
Resource & Query Aware Resource Discovery [3]

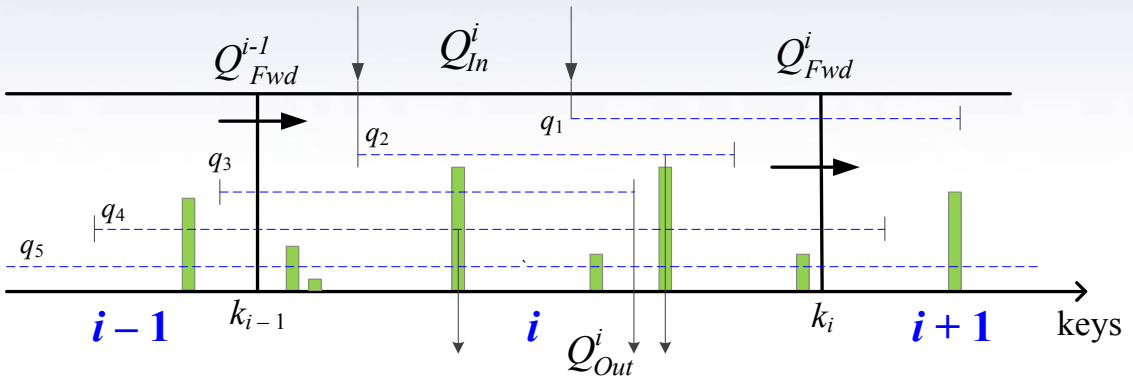
- Ring-based resource discovery solutions
  - Pros – Scalable & performance guarantees
  - Cons – High query ($O(N)$) & advertising cost, & unbalanced load
    - Conventional solutions assume $D_i \gg N$
    - Add more nodes to balance load $\sim R/N$ & $\sim Q/N$
- Domain of some attributes is small $D_i \ll N$
  - E.g., CPU cores, CPU architecture, & OS
- Less specific queries
  - Not useful to advertise even attributes with large D_i at high resolutions
    - E.g., Free CPU 40-100%, Free Disk 5-1000 GB
  - Effectively, $D_i \ll N$
- $N = \max(D_i)$
  - How to reduce N while balancing load?

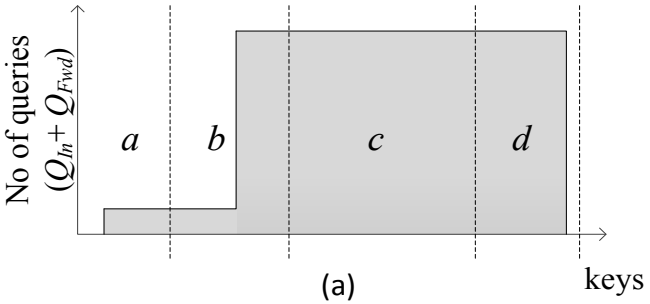


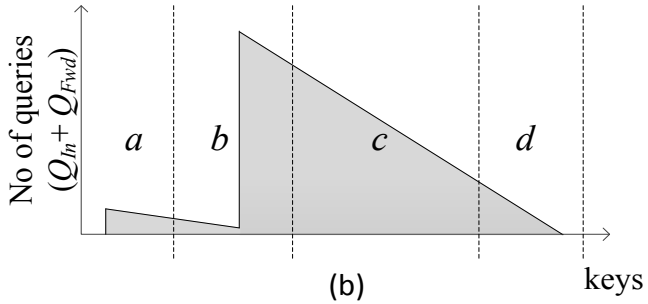
$$C^q_{Query} = \sum_{i \in \mathbf{a_q}} \left(h^i_{Query} + \left\lceil \frac{(u_i - l_i)}{D_i} N \right\rceil - 1 \right)$$

Heuristic 1 – Prune Nodes With Lower Contribution









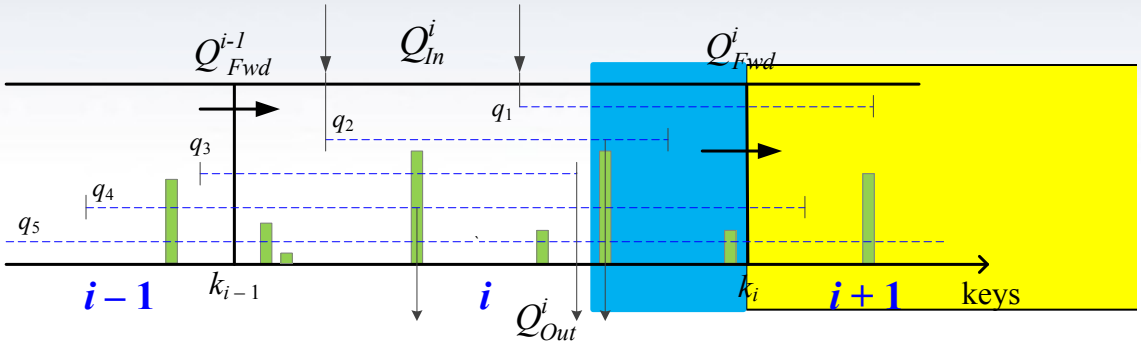
Heuristic 1

- Nodes have fixed resource index & query capacity

a) Remove c → Reduce query cost $\quad Q^{c}_{Out} = 0$
   - Can b or d accept any resources indexed at c?
   - d is preferred as no changes are required to overlay network

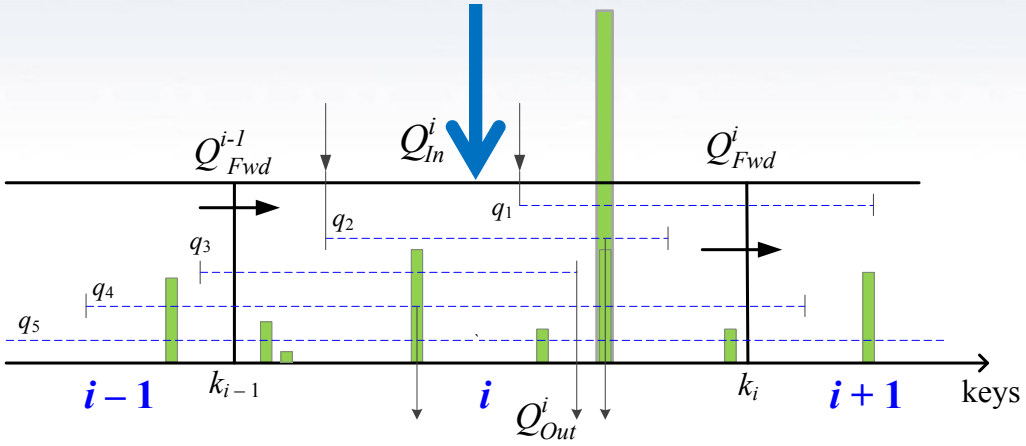
b) Remove a, b, or d → Reduce query cost $\quad Q^{i}_{Out} < Q^{i}_{Thr}$
   - Can neighbors accept resource index & query load?

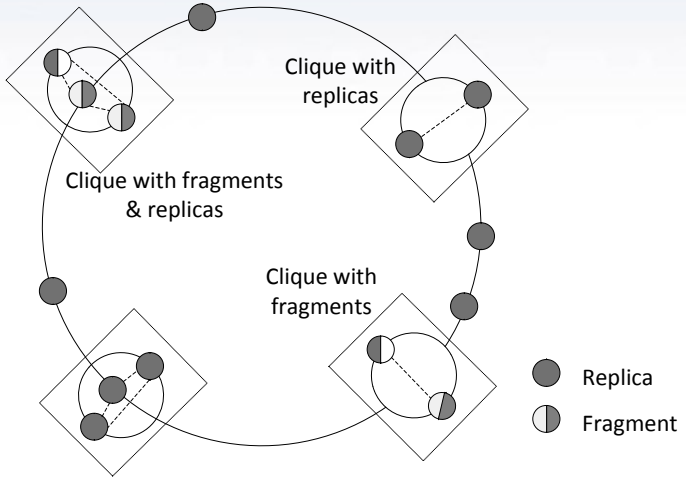
Heuristics 2 & 3 – Key Transfer



- Nodes are already contributing & overloaded
- Heuristic 2
  - i is overloaded
  - Move keys/resources to successor or predecessor – If it can accept
  - Successor is preferred
  - Minor changes to overlay
- Heuristic 3
  - i is overloaded & successor & predecessor not willing to accept load
  - Add new successor or predecessor – Load must not exceed capacity of a node
  - Successor is preferred
  - Some changes to overlay

Heuristics 4 & 5 – Replication & Fragmentation

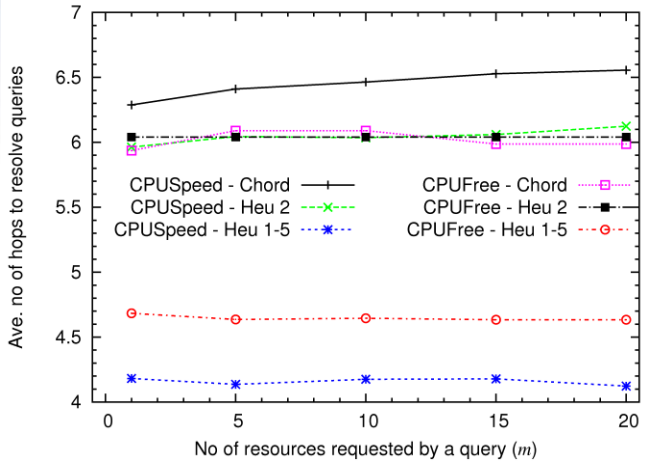


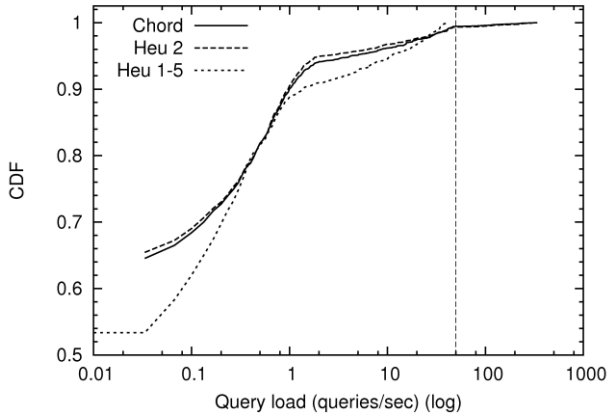


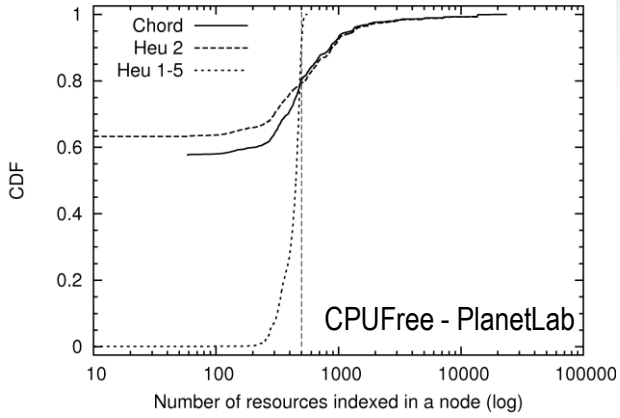
- Heuristics 2 & 3 will fail if load is too much for a single node
- Heuristic 4
  - Query load is too high
  - Add new node & replicate index
  - Don't increase query cost
  - More changes to overlay
- Heuristic 5
  - Resource index is too large
  - Add new node & fragment index
  - Rarely increase query cost
  - More changes to overlay

In practice, nodes can index many resources & answer many queries/second → Cliques are not large

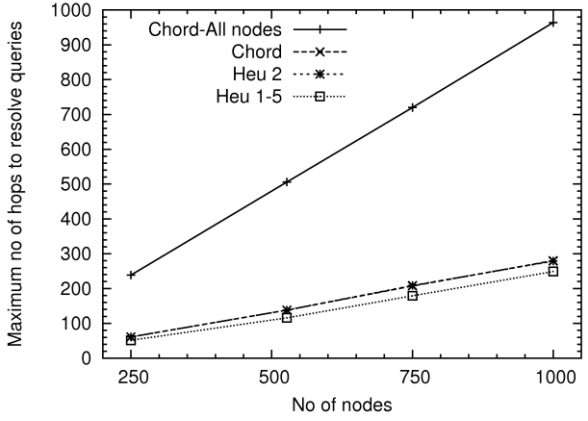
Resource & Query Aware Resource Discovery – Performance Analysis

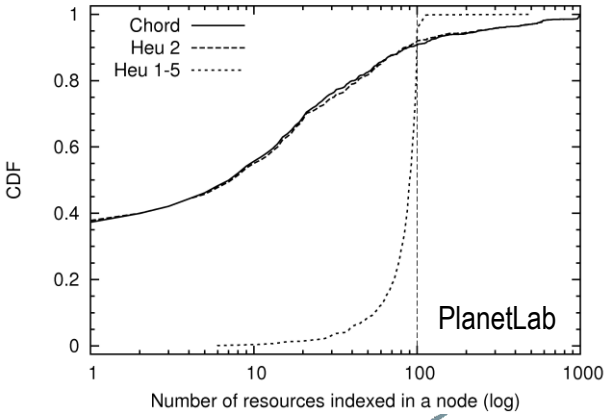






- Each heuristic addresses a specific problem
- More efficient & load balanced solution when all 5 heuristics are combined
  - Work with both single & multi-attribute resources





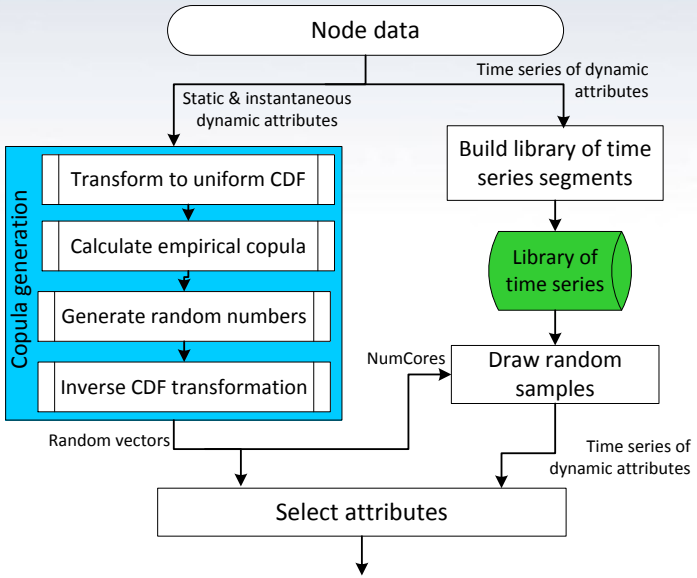
Outline

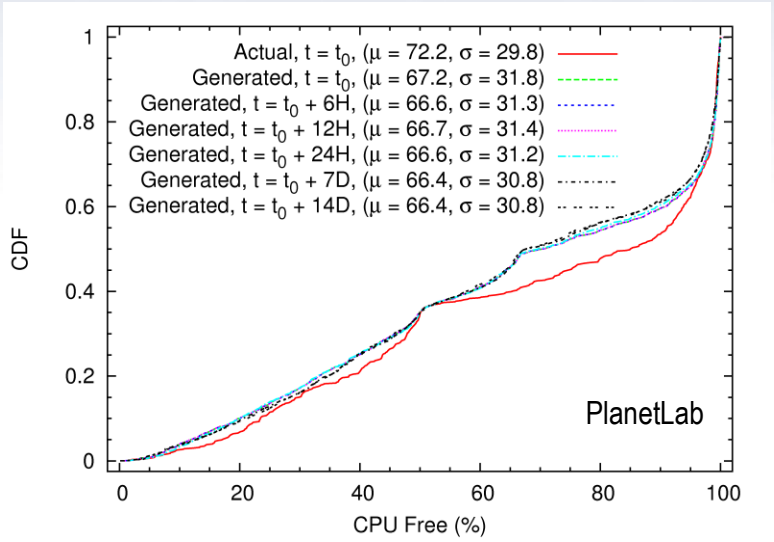
- Motivation
- Problem statement
- Multi-attribute resource & query characteristics
- Resource & query aware resource discovery
- Multi-attribute resource & range query generation
- Community-based caching
- NDN for DCAS
- Summary & future work

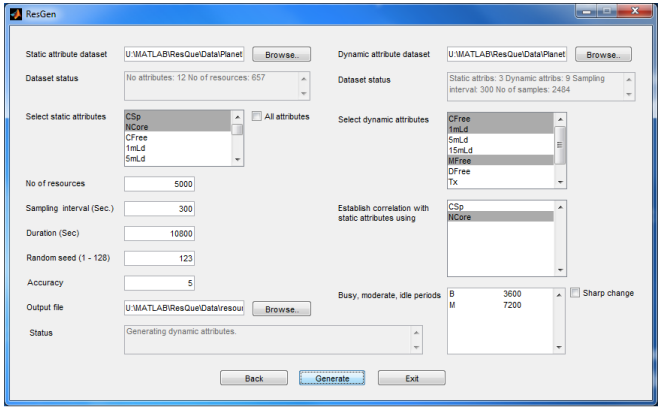
ResQue – Resource & Query Generator [8, 13]

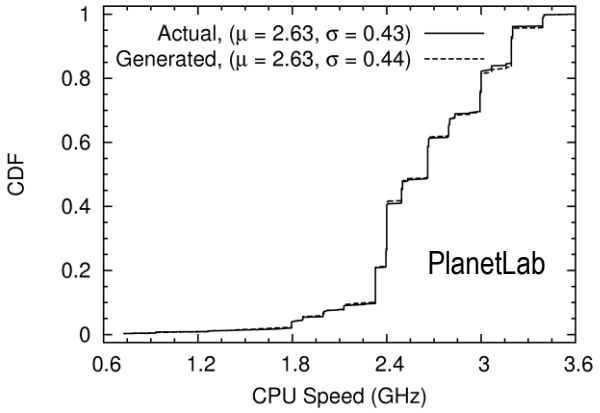
- Large-scale performance studies need large datasets
  - Neither practical nor economical to capture large datasets at high resolution
- Generate large synthetic traces using information gathered from small real-world traces
  - Resources
    - Large no of resources, many attributes, & temporal changes
    - Vectors of static attributes – Empirical copula
    - Time series of dynamic attributes – Library of time series segments
      - Detect structural changes in time series
  - Multi-attribute range queries
    - Probabilistic finite state machine
  - Preserve statistical characteristics, dependencies, & temporal patterns
  - Dataset neutral

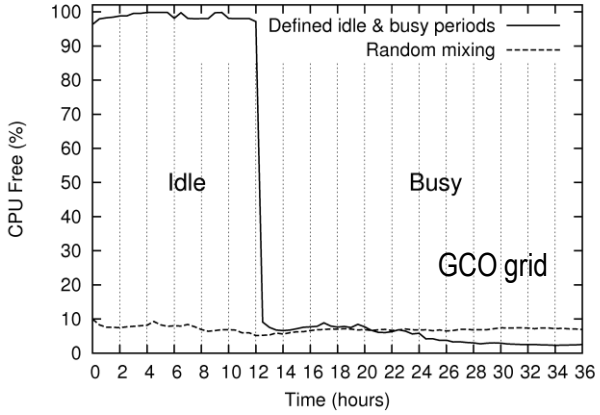
ResQue – Multi-Attribute Resource Generation











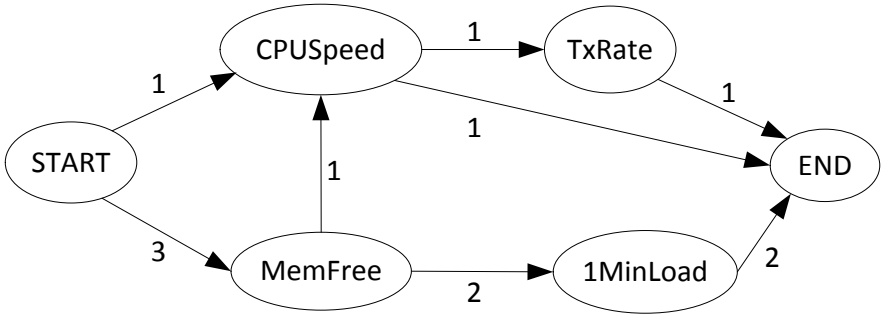
- Satisfy KS-test with a significance level of 0.05
- Available www.engr.colostate.edu/cnrl/Projects/CP2P/

ResQue – Multi-Attribute Range Query Generation

$Q_1 = \{CPUSpeed\}$ $\times 1$

$Q_2 = \{MemFree, 1MinLoad\}$ $\times 2$

$Q_3 = \{MemFree, CPUSpeed, TxRate\}$ $\times 1$



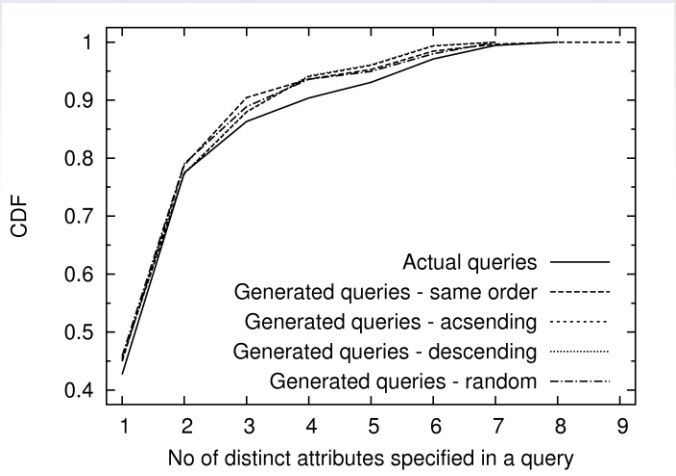
$q_1 = \{CPUSpeed\}$ 1/8

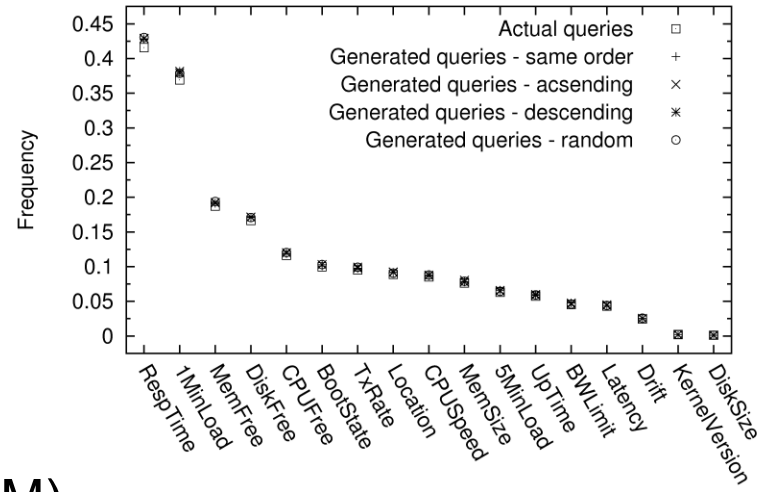
$q_2 = \{CPUSpeed, TxRate\}$ 1/8

$q_3 = \{MemFree, 1MinLoad\}$ 1/2

$q_4 = \{MemFree, CPUSpeed\}$ 1/8

$q_5 = \{MemFree, CPUSpeed, TxRate\}$ 1/8





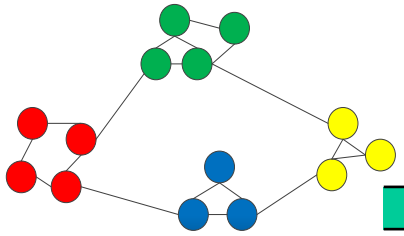
- Probabilistic Finite State Machine (PFSM)
- No of attributes, popularities, & occurrences of attribute pairs are similar
- Satisfy KS-test with a significance level of 0.05

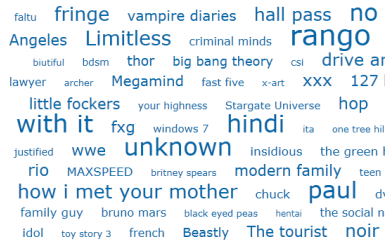
Outline

- Motivation
- Problem statement
- Multi-attribute resource & query characteristics
- Resource & query aware resource discovery
- Multi-attribute resource & range query generation
- Community-based caching
- NDN for DCAS
- Summary & future work

Community-Based Caching [1, 9]

- Small communities are emerging within large P2P systems
  - Based on semantic, geographic, & organizational interests
    - BitTorrent communities
  - Objective – Gain better performance while being in a large system
- Ways to improve query/lookup performance
  1. Satisfy only the most dominant queries
  2. Form clusters of communities

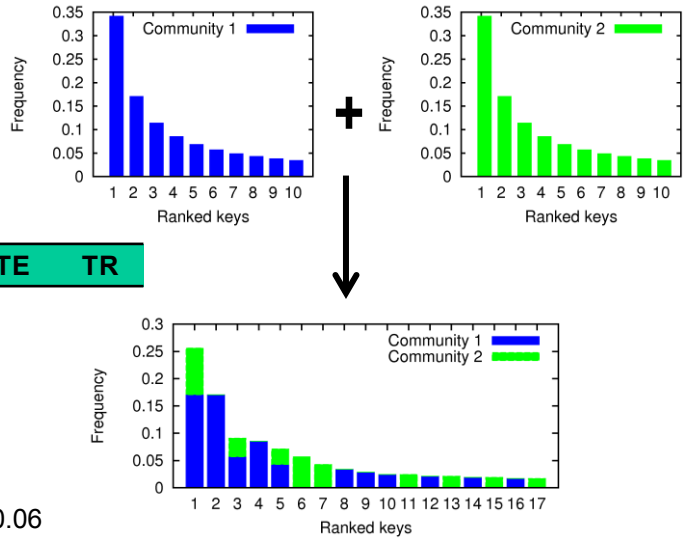




Community*	EX	FE	SP	TB	TS	TE	TR
fenopy.com (FE)	0.38						
seedpeer.com (SP)	0.00	0.00					
torrentbit.net (TB)	0.40	0.29	0.00				
torrentscan.com (TS)	0.48	0.33	0.00	0.48			
torrentsection.com (TE)	0.53	0.23	0.00	0.31	0.25		
torrentreactor.net (TR)	0.10	0.08	0.00	0.06	0.09	0.06	
youbittorrent.com (YB)	0.36	0.35	0.00	0.29	0.42	0.20	0.04

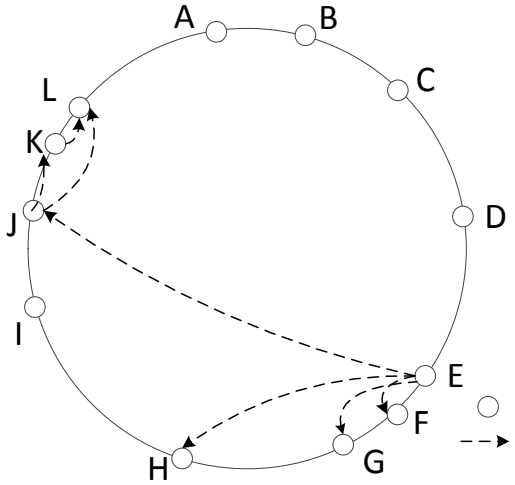
Cosine similarity among search clouds of communities

1, 3, & 4 are same as 2, 5, & 6

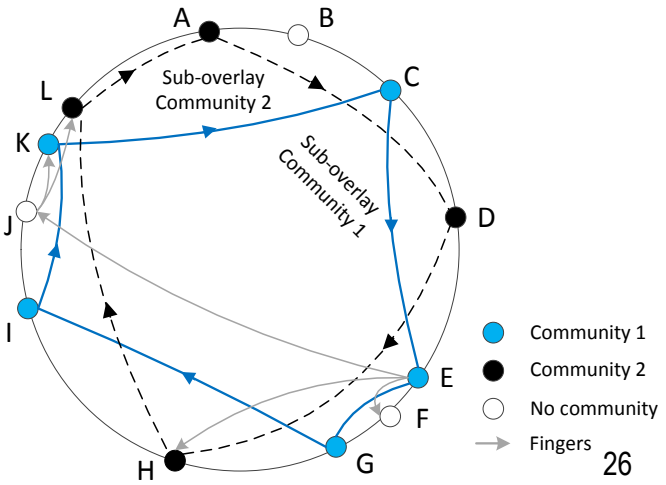


Community-Based Caching (Cont.)

- Reduce mixing among communities while in same overlay
  1. Sub-overlay among community members
     - Nodes indicate their communities using Community IDs
     - Find community members by sampling routing tables of nodes pointed by fingers
     - Maintain fingers to those community members
     - Overlay properties are preserved
  2. Cache contents based on community interest
     - "*What is important to me is also important to other community members*"
     - Local Knowledge-based Distributed Caching (LKDC)



- Path length $O(\log N)$
- By probing i-th finger & its successor $2(i + 2\log_2 N - b) - 1$ nodes can be found
  - $1 \le i \le b$
- Community of size M has $M/2^{b-i+1}$ peers within the range of i-th finger



Distributed Local Caching

- Each overlay node
  - Independently decides what keys to cache based on the queries it forwards
  - Tries to minimize average query cost

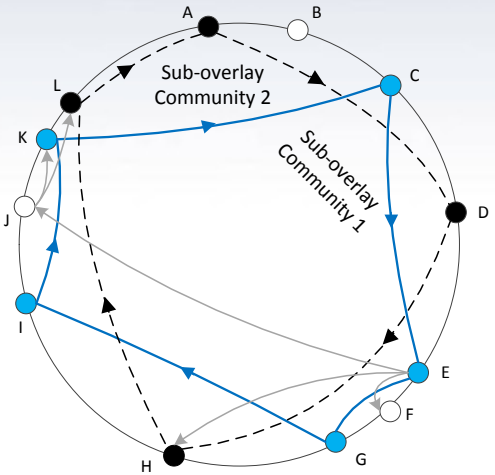
$$h_{ave}^{n} = \frac{\text{Totalhops}}{\text{Totalqueries}} = \frac{\sum_{k \in \mathbf{K}} \lambda_k^n h_k^n \left(1 - x_k^n\right)}{\sum_{k \in \mathbf{K}} \lambda_k^n}$$

  - Maximize hop count reduction while satisfying its cache capacity C_n

$$\text{maximize } R = \sum_{k \in \mathbf{K}} \lambda_k^n h_k^n - \sum_{k \in \mathbf{K}} \lambda_k^n h_k^n \left(1 - x_k^n\right) = \sum_{k \in \mathbf{K}} \lambda_k^n h_k^n x_k^n$$

$$\text{subject to } \sum_{k \in \mathbf{K}} S_k x_k^n \le C_n$$

  - NP complete
  - For improving lookup performance ok to assume $S_k = 1$ → Cache keys with largest $\lambda_k^n h_k^n$



k – key

n – node

λ_k^n – Demand for key k

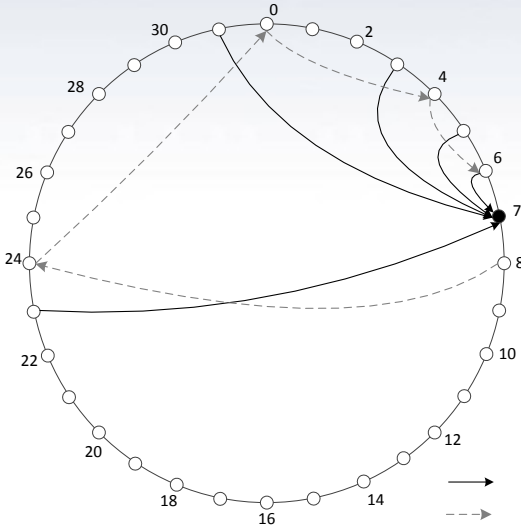
h_k^n – Hops to resolve query at n

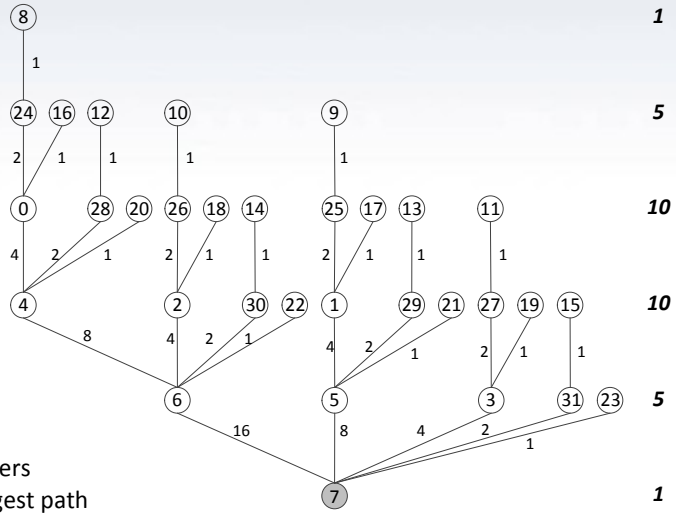
$x_k^n \in \{0,1\}$ – k cached at n

S_k – Size of key/content k

Local Knowledge-based Distributed Caching (LKDC)

Distributed Local Caching (Cont.)





- Longest path $\log_2 N$
- Ave. path ½ $\log_2 N$
- Bell shaped distribution of path lengths

- Where to place cache entries?
  - At nodes that forward most number of messages
  - 6, (4, 5), (0, 1, 2, 3), …
  - Hops reduce 16, 8, 8, 4, 4, 4, 4, 2, …
  - Hops reduce by placing c_k entries

$$g(c_k) = \frac{N}{2} \sum_{i=1}^{c_k} 1/2^{\lfloor \log_2 i \rfloor}$$

- How many entries to create?
  - Problem

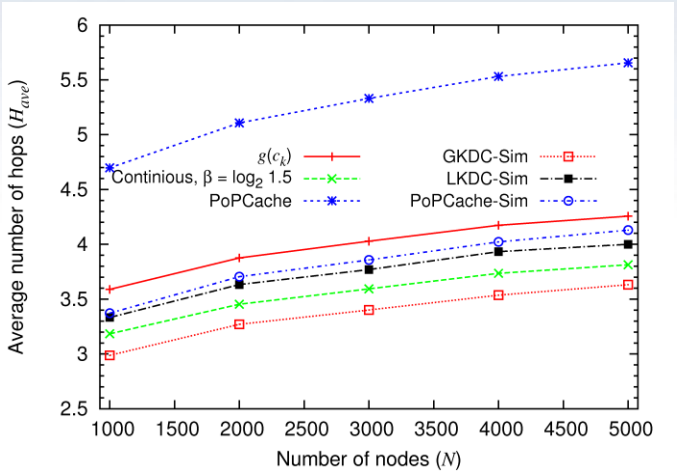
$$\text{minimize } H_{ave} = h_{ave} - \frac{1}{N} \sum_{k \in \mathbf{K}} f_k g(c_k)$$

$$\text{subject to } \sum_{k \in \mathbf{K}} c_k = B, \quad c_k \le N-1 \quad \forall k \in \mathbf{K}$$

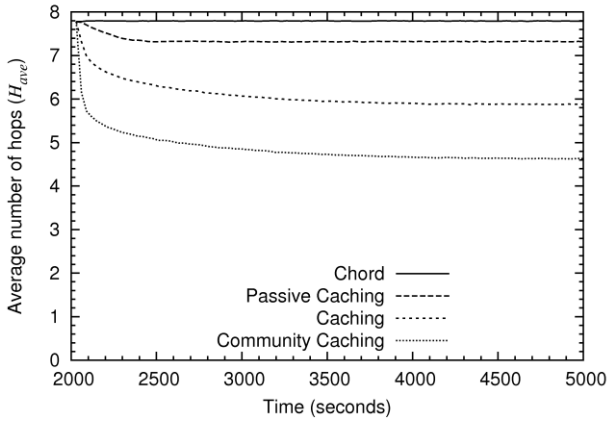
  - Solution
    - Allocate in proportion to popularity

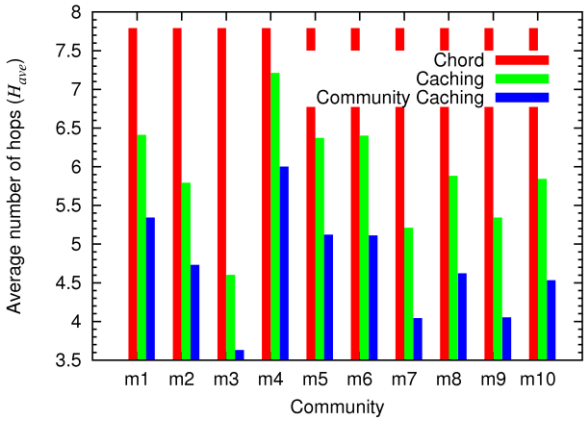
$$c_k = \begin{cases} N-1 & \text{if } k \le l \\ \dfrac{f_k \left(B - l(N-1)\right)}{P(l, K, \alpha)} & \text{else} \end{cases}$$

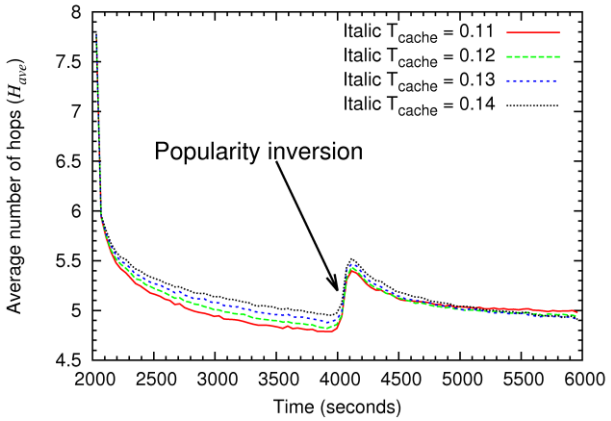
Community-Based Caching – Performance Analysis



Community	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}
No of nodes (apx.)	600	600	600	1,200	1,200	1,200	1,200	1,200	2,400	4,800
Zipf's parameter	0.85	0.95	1.10	0.5	0.80	0.80	1.0	0.90	0.90	0.75
No of distinct keys	40,000	30,000	30,000	40,000	40,000	40,000	50,000	50,000	50,000	50,000
Similarity with community (x)	0.2 (C_8)	0	0.1 (C_7)	0.2 (C_9)	0.3 (C_8) 0.5 (C_7)	0	0.1 (C_3) 0.5 (C_5)	0.3 (C_5) 0.2 (C_1)	0.4 (C_1) 0.2 (C_4) 0.3 (C_{10})	0.3 (C_9)
Queries for rank 1 key	4,516	8,535	17,100	603	6,454	6,454	21,059	11,956	23,911	17,030







- Model provides a lower bound & more accurate than previous approaches
- 40% reduction in average path length
  - Most popular communities – 48-53%
  - Least popular community – 23% (7% with caching)
- Quickly adapt to rapid changes in popularity

Outline

- Motivation
- Problem statement
- Multi-attribute resource & query characteristics
- Resource & query aware resource discovery
- Multi-attribute resource & range query generation
- Community-based caching
- NDN for DCAS
- Summary & future work

NDN for Data Fusion in DCAS Systems [10]

- Internet
  - Designed to share resources → End-to-end
  - Users value ability to access contents → End irrelevant
  - Traffic aggregation, location dependence, & security
- Named Data Networking (NDN/CCN)
  - Access & route contents based on application layer names
  - In-network caching, duplicate message suppression, on demand data generation, better security, & incremental deployment

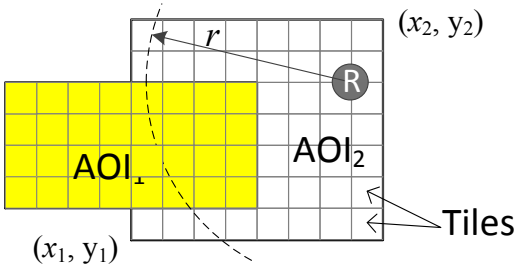
– Distributed Collaborative Adaptive Sensing (DCAS) systems
  – Multiple redundant sensors, multi-application, & multi-user systems
  – Data pull – Users' information needs determine how system is used
  – Sensor specific data names
    - *"Reflectivity data from CSU CHILL"*
  – Users are concerned about a particular event(s) occurring within an Area Of Interest (AOI)
    - *"Reflectivity over Fort Collins"* or *"Wind speed in southwestern Oklahoma"*

Why NDN for CASA?

Geographic location & weather event specific names • Queries & data • Aliases for same data	Content dependent names • 2 packet types – Interests & data • `/FortCollins/Reflectivity/13:32/` • Multiple names
Decouple data, security, & access from sensor • Use any available sensor	Decouple identity, security, & access from end point
High temporal & spatial locality	Exploit temporal & spatial locality
Pull based • End-user information needs determine what & how resources are used	Receiver driven communication • On demand data generation
Overlay routing	Multiple routing schemes
Load balancing, resilience, & security • Multi-path routing & mobility	Better reliability & security • Multi-path routing & mobility

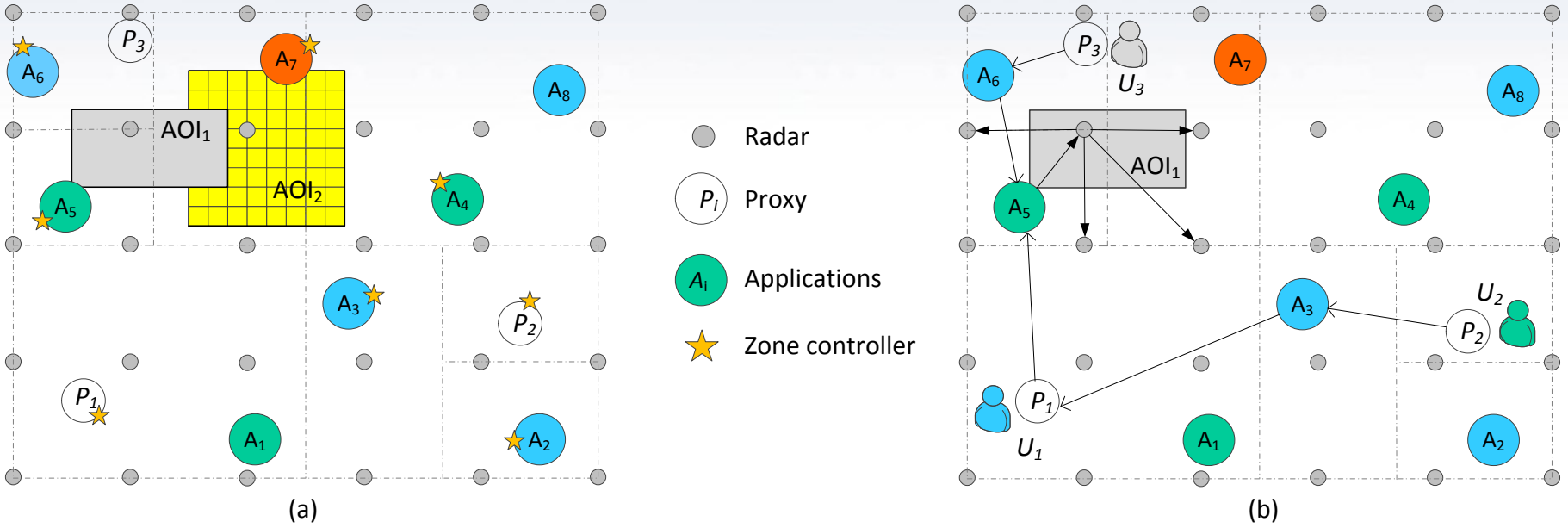
NDN for DCAS – Naming Data

- End users specify an AOI, application, & time
  - /AOI/application/time/
  - Interest packet is looking for an application near AOI
    - Process data close to source → Save bandwidth
  - AOI is typically expressed as a rectangular area
    - /x_1/y_1/x_2/y_2/application/time
    - Larger AOIs are broken into smaller ones



- Application needs to subscribe to radars
  - CASA radars negotiate among themselves on how to provide data
  - /x_1/y_1/x_2/y_2/radar/time/subscription/n/dataType
  - PIT is modified to accept up to n data packets per tile
- Application pull data from selected radars
  - /x_R/y_R/x_R/y_R/radar/time/x_1/y_1/x_2/y_2/bitmap/dataType

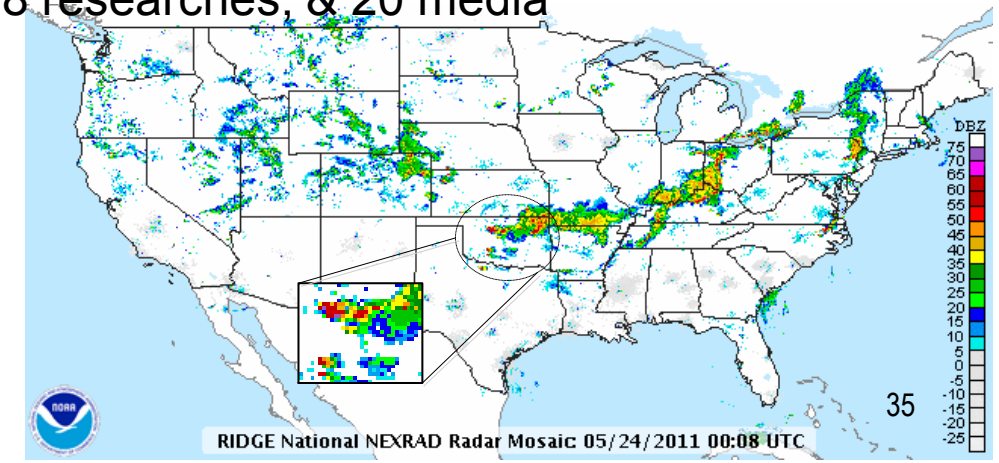
NDN for DCAS – Overlay Construction & Query Resolution



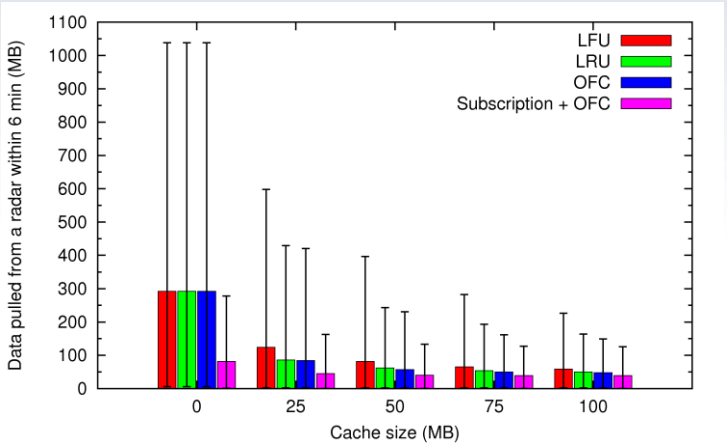
- Overlay routing – Content Addressable Network (CAN)
  - Maps to 2D space while preserving locality
  - No local minimas as in other greedy routing solutions
- End users connect to overlay using a set of proxies
- In network caching & duplicate interest suppression

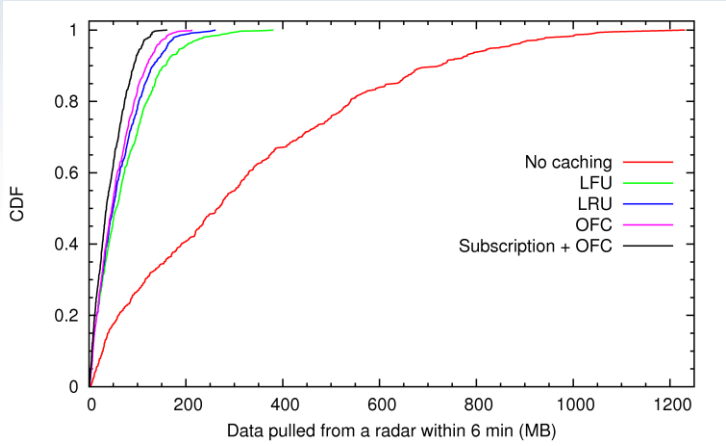
NDN for DCAS – Simulation Setup

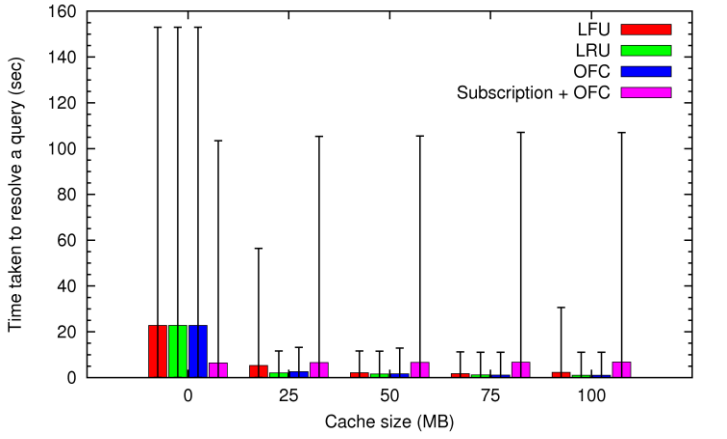
- Parameters from CASA IP1 test bed
  - 121 radars placed on a 300 km x 300 km area, 30 km apart, 40 km range
  - 30 PPI scans, unsynchronized radars
  - 4 bytes per data type per tile (tile 100 m x 100 m)
  - 5 proxies, 16 x 2 reflectivity & velocity, & 4 x 3 NBRR, nowcasting, & QPE
  - 1 Gbps links
- Reflectivity data from a large-scale weather event over Oklahoma
  - 05/23/2011 10:00pm to 05/24/2011 2:00am
  - AOI – Active weather if reflectivity ≥ 25 dBz
  - End users – 2 NWS, 30 EMs, 8 researches, & 20 media

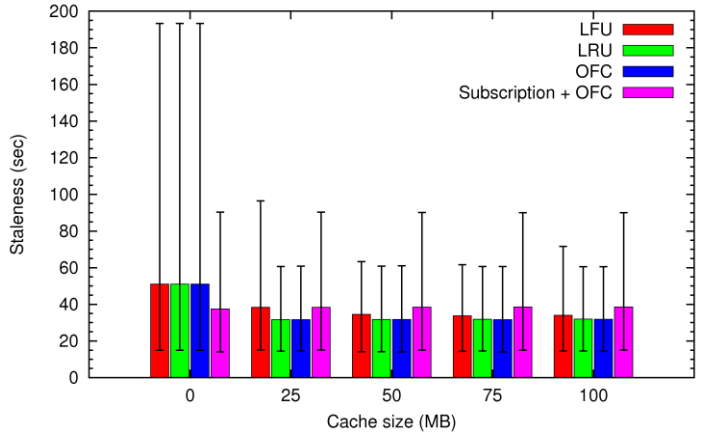


NDN for DCAS – Performance Analysis









- Bandwidth requirements are reduced
  - Subscription scheme – 61%, Oldest First Caching (OFC) – 87%
  - Better load distribution
- Better quality data – Waiting time & staleness is reduced
  - Waiting time – 88%, Staleness – 69%

Summary

- Proposed a P2P-based approach to enable collaboration of group of heterogeneous resources
- Achieved goal of enabling integration of groups of resources & data fusion
  - Real-world datasets exhibit several noteworthy features that affect performance of resource aggregation
  - Resource & query aware P2P-based resource discovery solution
  - Tool to generate synthetic resource & query traces
  - Community-based caching for large P2P systems
  - Demonstrated applicability of NDN for DCAS

Collective power of P2P communities & their resources → Globally distributed virtual clouds for many applications

Future Work

- Support all key phases of resource aggregation [4, 12]
  - Extend resource & query aware resource discovery solution
  - Hybrid between DHT & superpeer
    - Superpeers – Good for resource matching & binding
- Identify semantic-based P2P communities within overlay [1]
  - Compare with cluster-based solutions, alternative routing, & churn
- Aggregate data from heterogeneous sensors in NDN
  - Integrate other sensors & enhance event-specific queries
  - Reference implementation based on CCNx
- Supporting incentives, trust, security, & privacy [4]
  - Determine ultimate success
  - With right tools & incentives in place, it will be more efficient & rewarding to accomplish a greater task through collaboration

Publications

1. H. M. N. D. Bandara and A. P. Jayasumana, “Community-based caching for enhanced lookup performance in P2P systems,” IEEE Transactions on Parallel & Distributed Systems, 2012, DOI: 10.1109/TPDS.2012.270.
2. H. M. N. D. Bandara, A. P. Jayasumana, and M. Zink, “Radar networking in collaborative adaptive sensing of atmosphere: State of the art and research challenges,” In Proc. IEEE GLOBECOM Workshop on Radar and Sonar Networks (RSN ‘12), Dec. 2012, To appear.
3. H. M. N. D. Bandara and A. P. Jayasumana, “Resource and query aware, peer-to-peer-based multi-attribute resource discovery,” In Proc. 37th IEEE Conf. on Local Computer Networks (LCN ‘12), Oct. 2012, To appear.
4. H. M. N. D. Bandara and A. P. Jayasumana, “Collaborative applications over peer-to-peer systems – Challenges and solutions,” Peer-to-Peer Networking and Applications, Springer New York, 2012, DOI: 10.1007/s12083-012-0157-3.
5. P. Lee, A. P. Jayasumana, H. M. N. D. Bandara, S. Lim, and V. Chandrasekar, “A peer-to-peer collaboration framework for multi-sensor data fusion,” Journal of Network and Computer Applications, vol. 35, no. 2, May 2012, pp. 1052-1066.
6. H. M. N. D. Bandara and A. P. Jayasumana, “Evaluation of P2P resource discovery architectures using real-life multi-attribute resource and query characteristics,” In Proc. IEEE Consumer Communications and Networking Conf. (CCNC ‘12), Jan. 2012.
7. H. M. N. D. Bandara and A. P. Jayasumana, “Characteristics of multi-attribute resources/queries and implications on P2P resource discovery,” In Proc. Int. Conf. on Computer Systems and Applications (AICCSA ‘11), Dec. 2011.

Publications (Cont.)

8. H. M. N. D. Bandara and A. P. Jayasumana, “On characteristics and modeling of P2P resources with correlated static and dynamic attributes,” In Proc. IEEE Global Communications Conference (GLOBECOM ‘11), Dec. 2011.
9. H. M. N. D. Bandara and A. P. Jayasumana, “Exploiting communities for enhancing lookup performance in structured P2P systems,” In Proc. IEEE Int. Conf. on Communications (ICC ‘11), June 2011.
10. H. M. N. D. Bandara and A. P. Jayasumana, “Distributed multi-sensor data fusion over named data networks,” In review.
11. P. Lee, A. P. Jayasumana, H. M. N. D. Bandara, S. Doshi, and V. Chandrasekar, "Analysis of multi-sensor, data-fusion latency in Internet-based distributed collaborative adaptive systems," In review.
12. H. M. N. D. Bandara and A. P. Jayasumana, “Multi-attribute resource and query characteristics of real-world systems and implications on peer-to-peer-based resource discovery,” In preparation.
13. H. M. N. D. Bandara and A. P. Jayasumana, “On characteristics and generation of multi-attribute resources and queries with correlated attributes,” In preparation.

Acknowledgments

- Prof. Anura Jayasumana
- Prof. V. Chandrasekar, Prof. Daniel Massey, & Prof. Indrajit Ray
- CASA & NSF (award number 0313747)
- Dr. Michael Zink, Veeresh Rudrappa, & Sudharshan Varadarajan
- Dr. Panho Lee, Dr. Sanghun Lim, Vidarshana, Saket, Dulanjalie, Pritam, Yi, Negar, & many other colleagues at CNRL
- Parents, wife, & son

Thank you!