Pacific Northwest
NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

StreamWorks – A System for Real-Time Graph Pattern Matching on Network Traffic

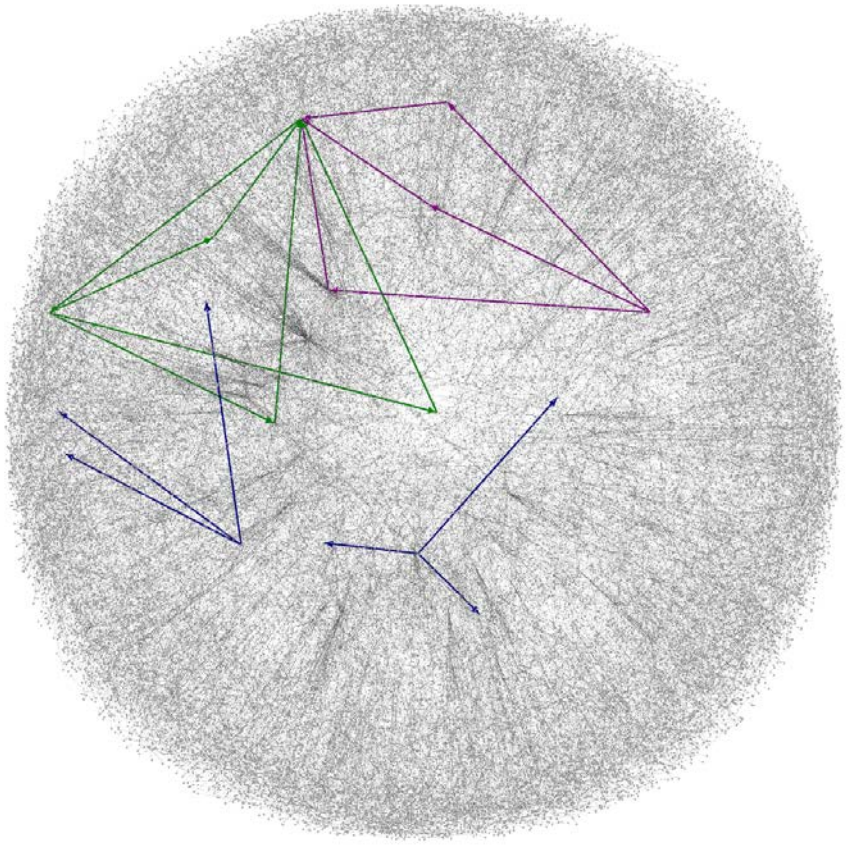
GEORGE CHIN, SUTANAY CHOUDHURY AND KHUSHBU AGARWAL
Pacific Northwest National Laboratory

Emerging Graph Patterns

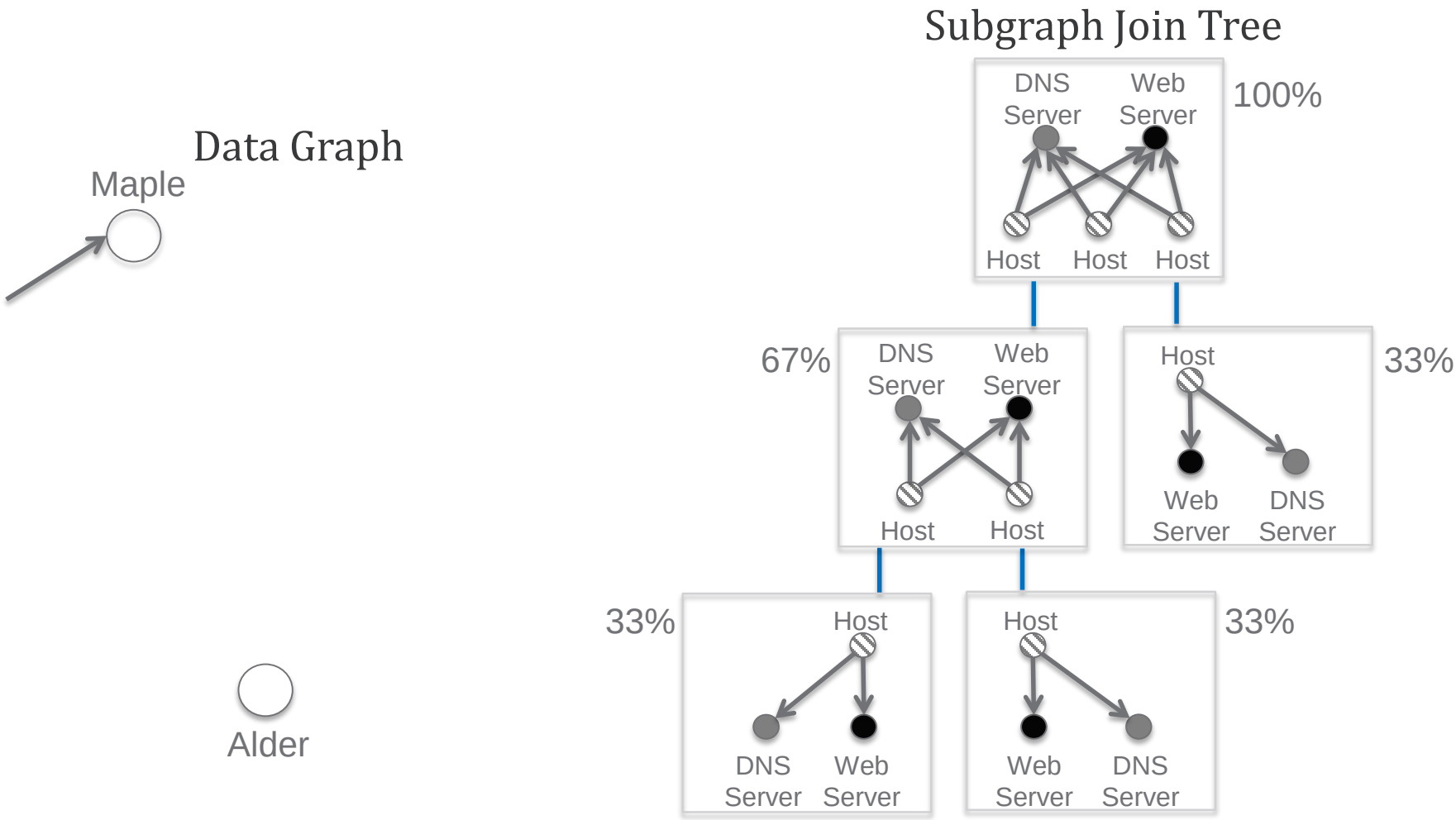
- Goal:

  Detect and identify precursor events and patterns as they emerge in complex networks such that events or threats may be mitigated or acted upon before they are fully realized

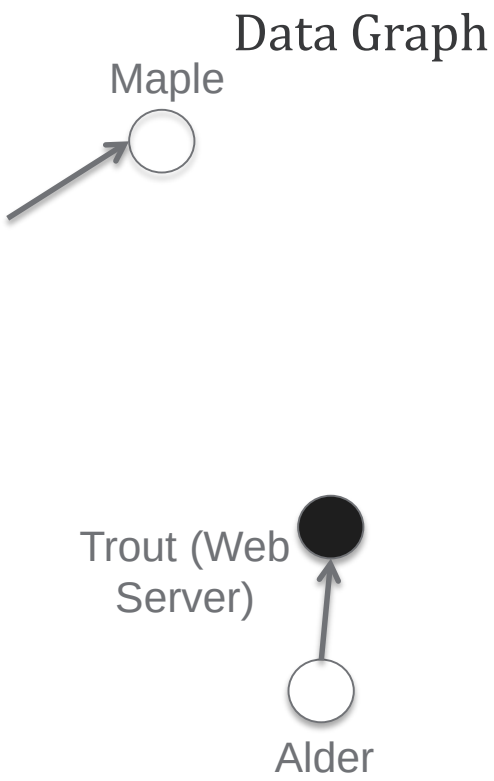
- Capture evolution of critical graph patterns
- Devise optimal search strategy to identify emerging pattern
- Consider cases where target subgraph patterns may or may not be known
- Subgraph pattern matching is a well-studied NP-hard problem. Some work on scalable algorithms
- Limited work on subgraph matching in dynamic networks
- Application areas:
  - Computer network intrusions and threats
  - Social media and network analysis
  - Financial and stock market analysis
  - Distributed sensor networks

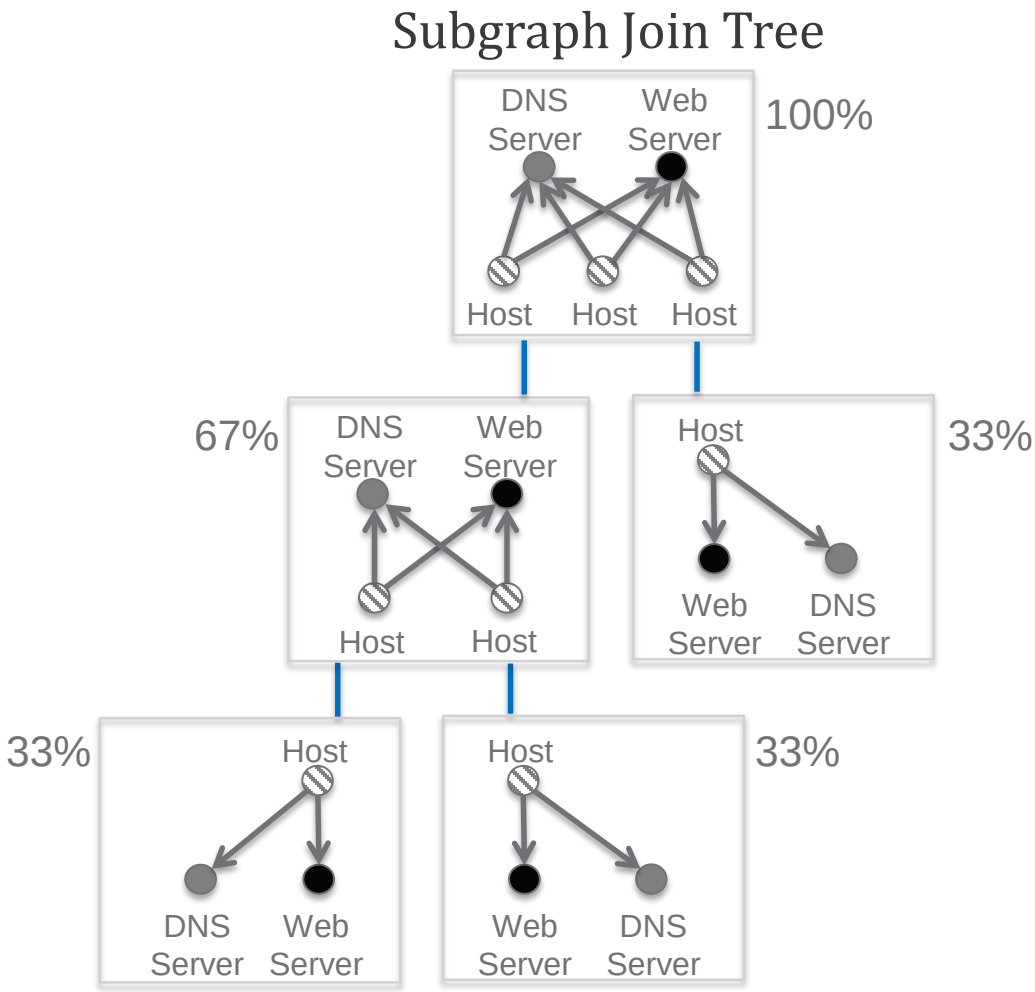


Emerging Graph Pattern Algorithm in Action

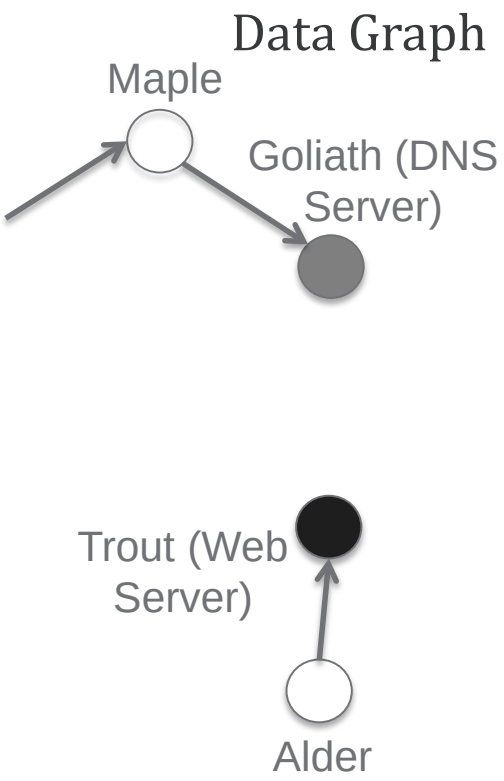


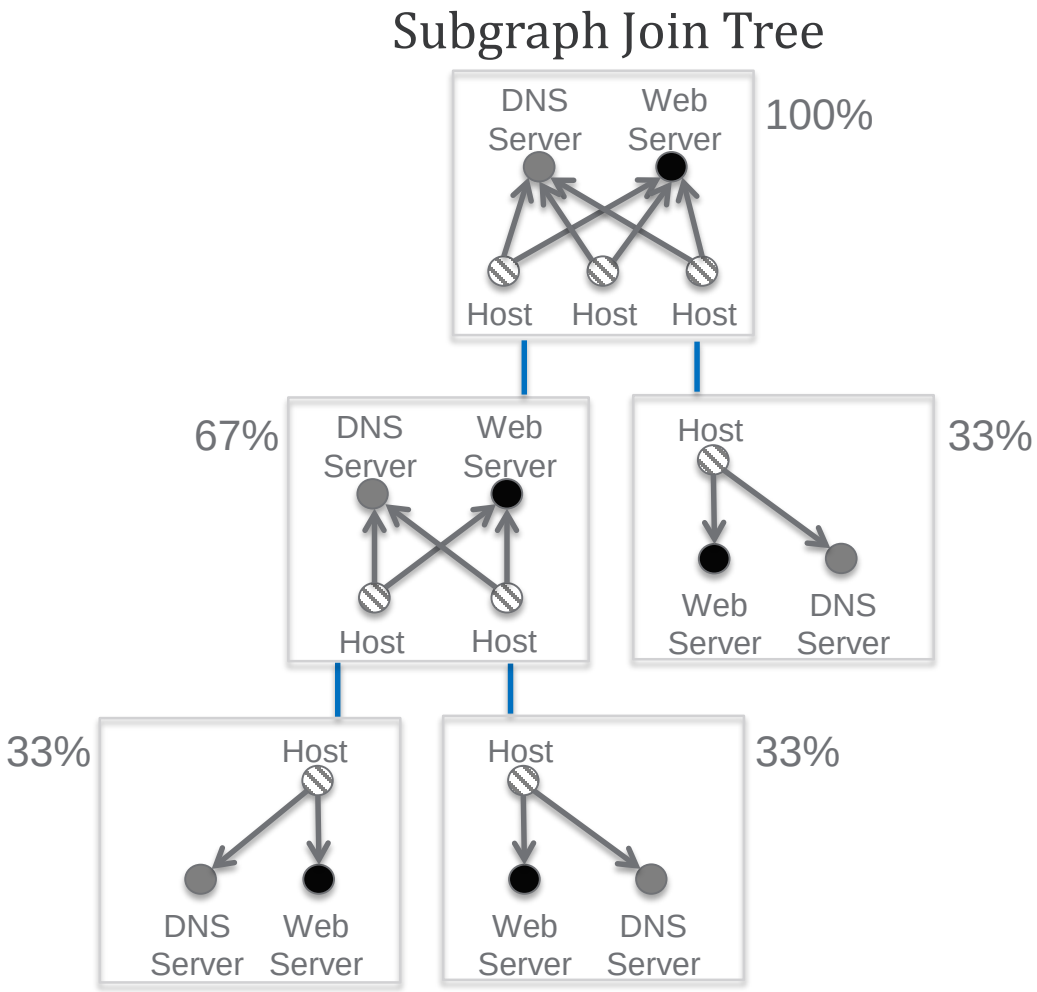
Emerging Graph Pattern Algorithm in Action



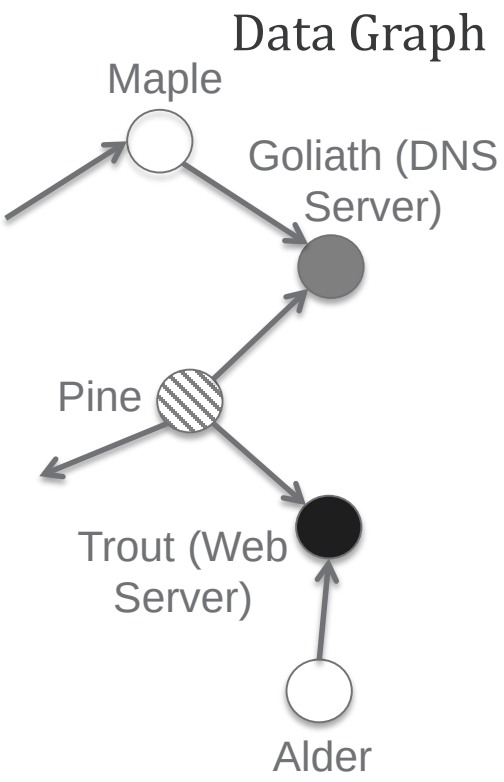


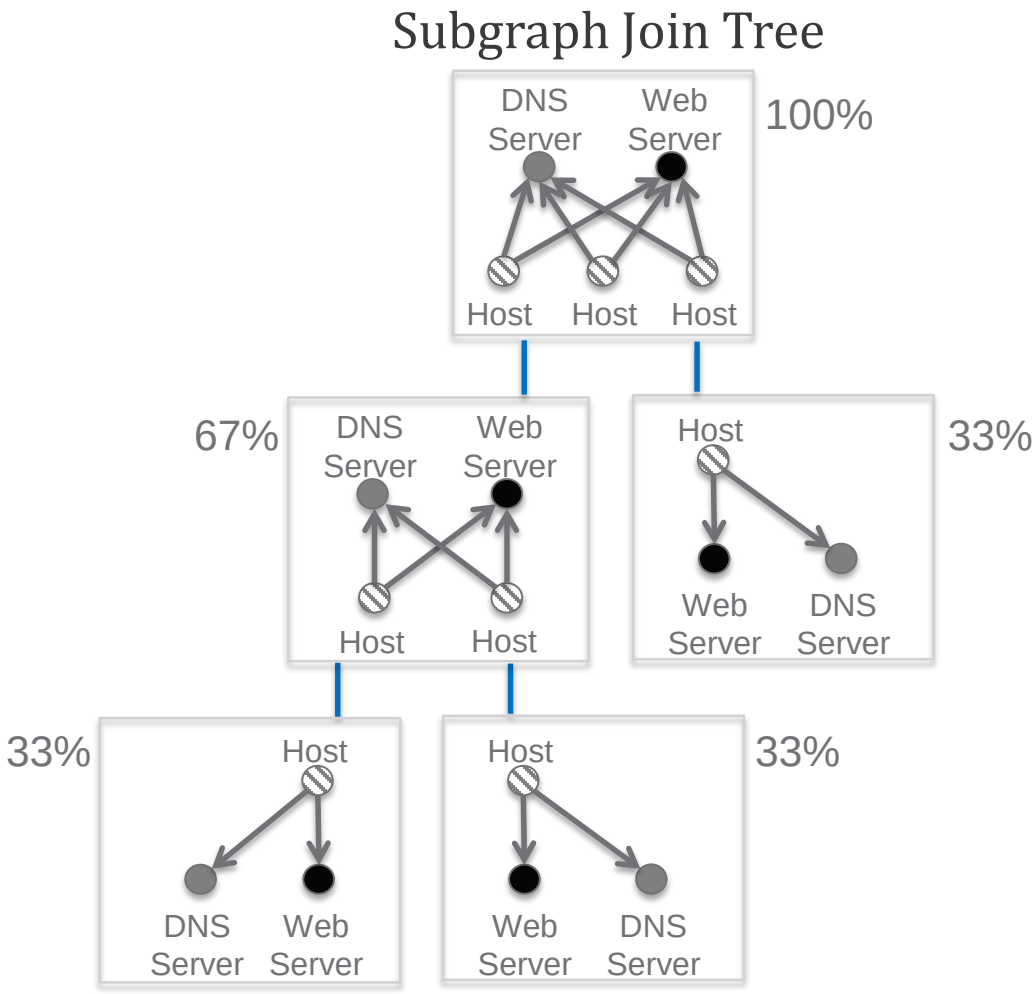
Emerging Graph Pattern Algorithm in Action



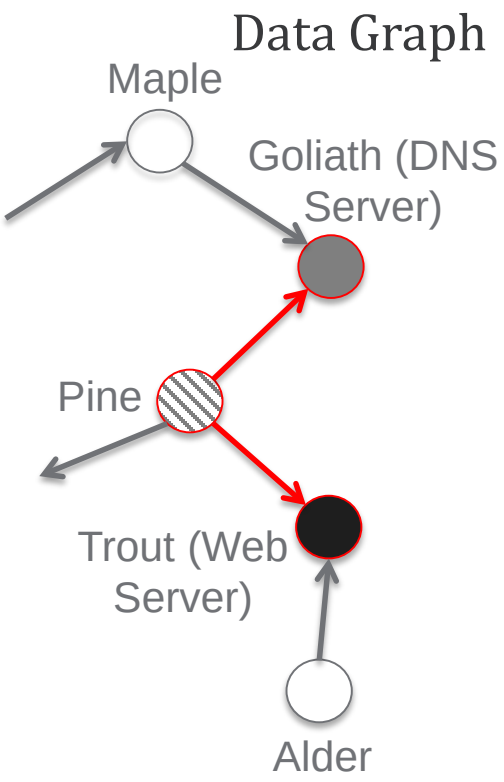


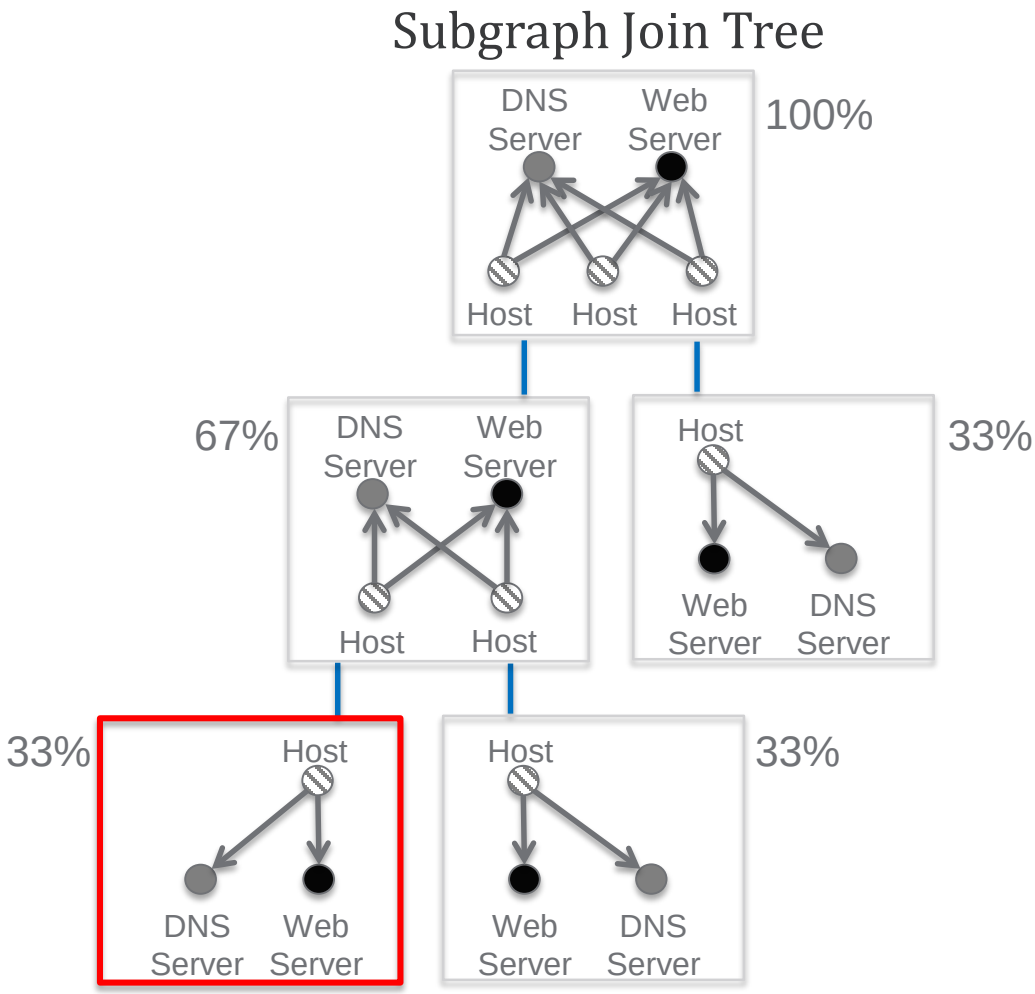
Emerging Graph Pattern Algorithm in Action



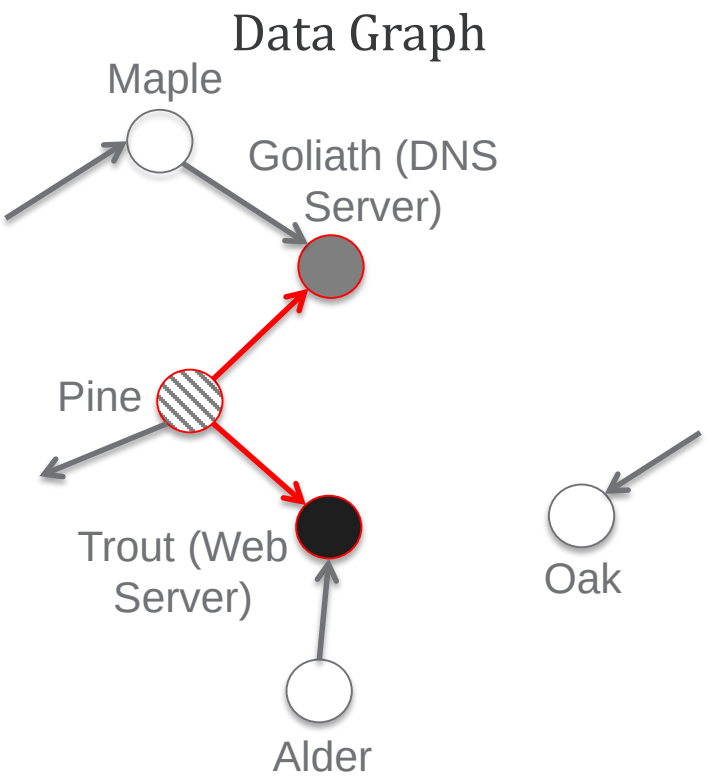


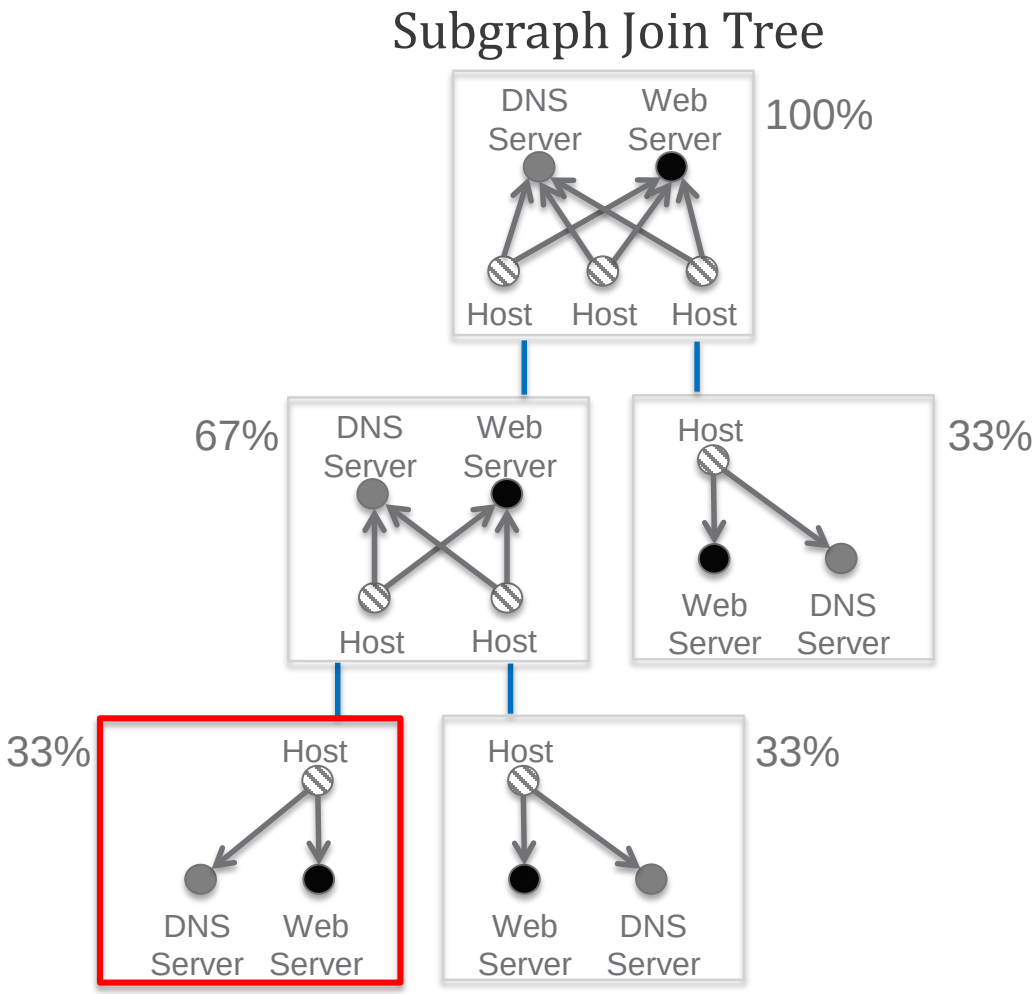
Emerging Graph Pattern Algorithm in Action



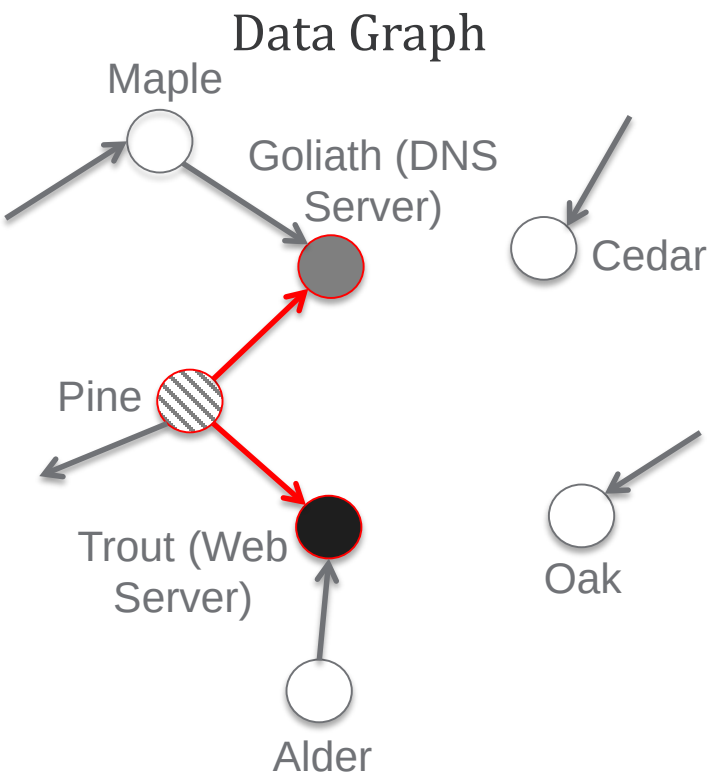


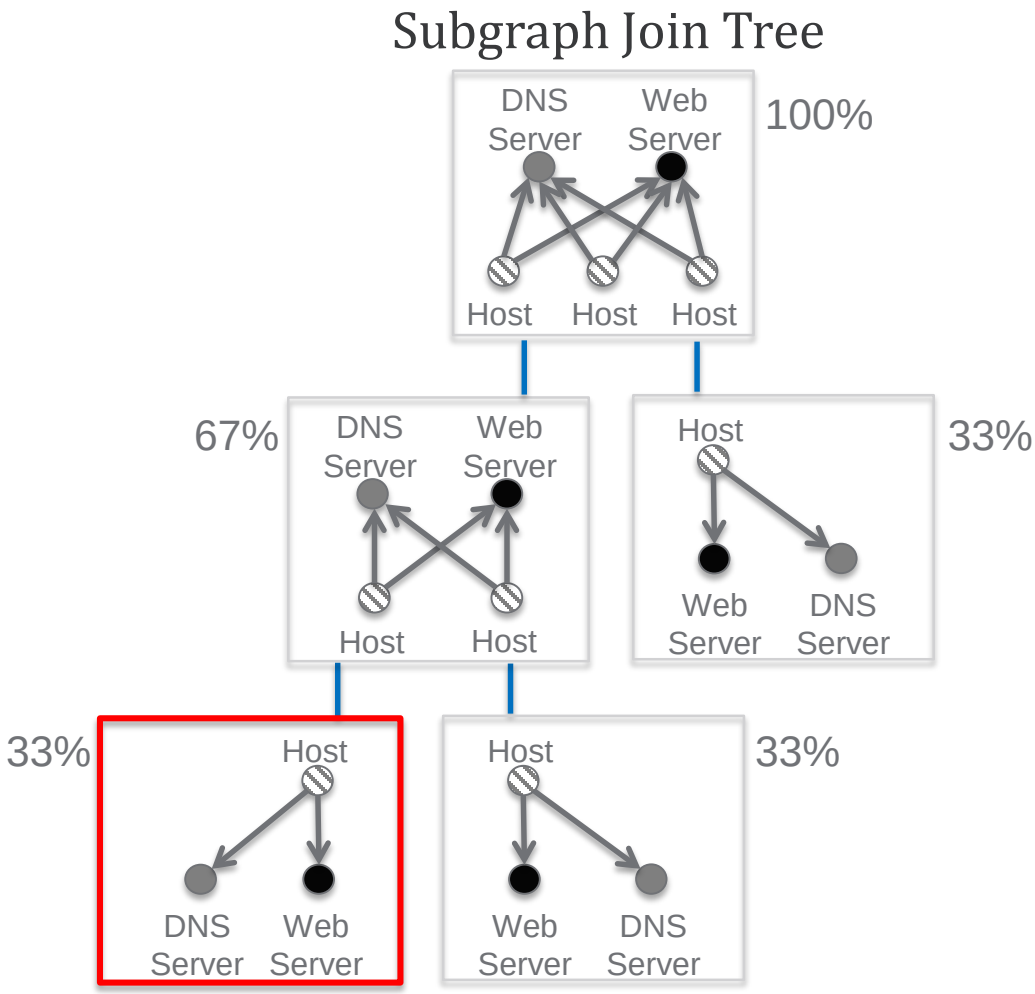
Emerging Graph Pattern Algorithm in Action



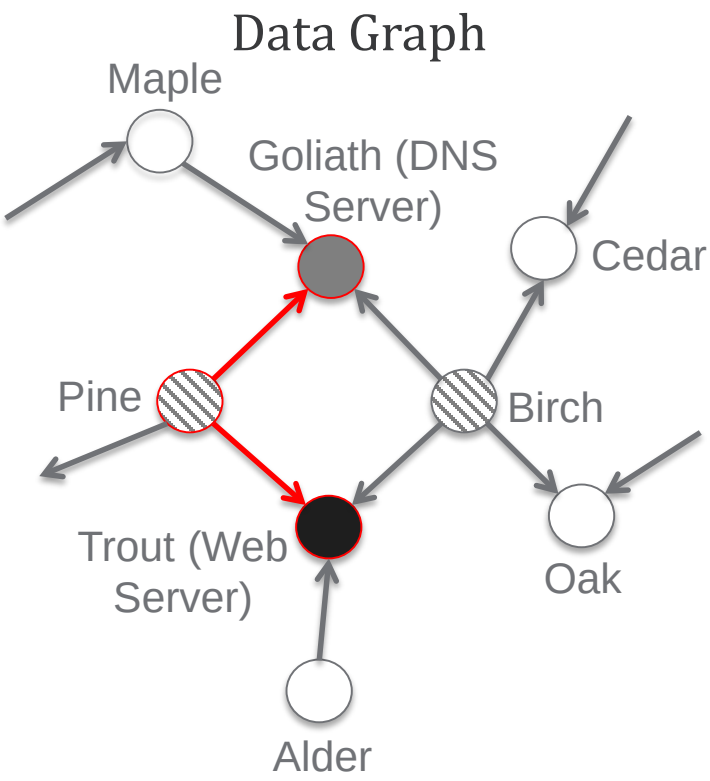


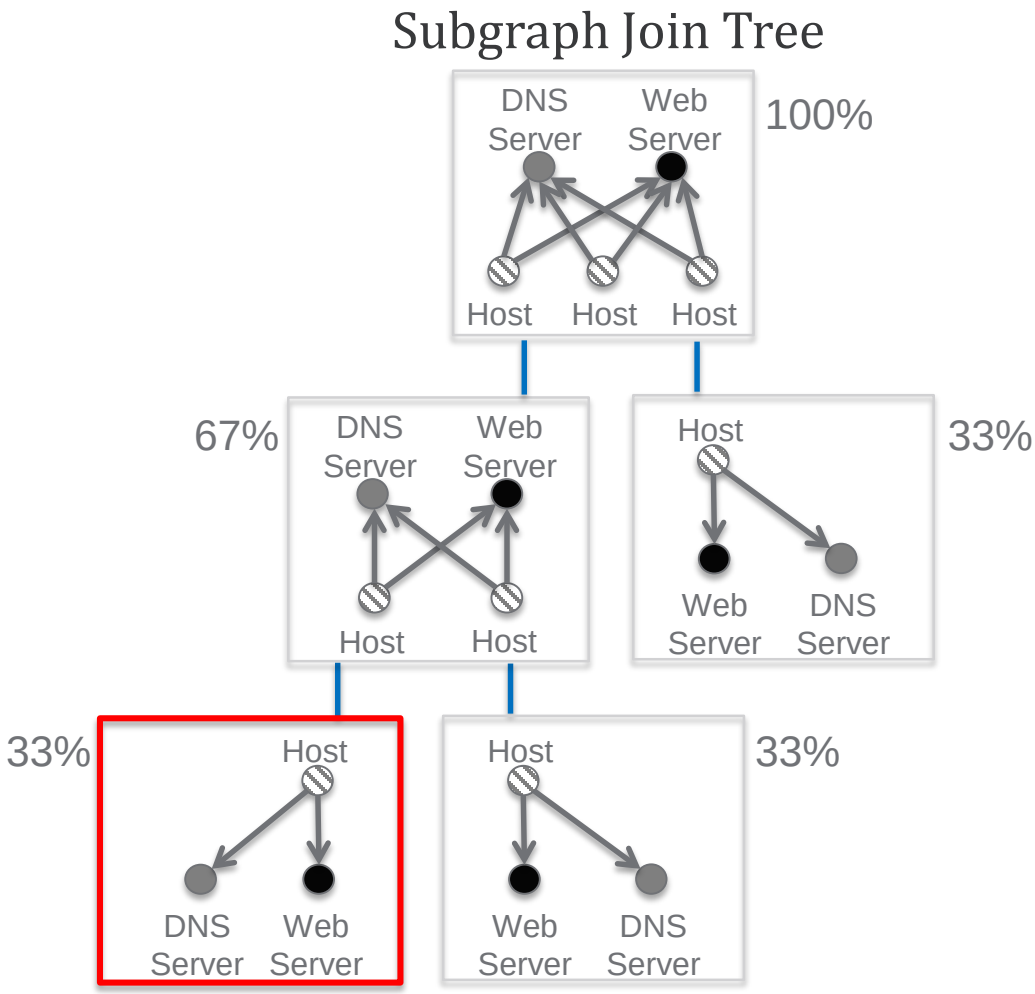
Emerging Graph Pattern Algorithm in Action



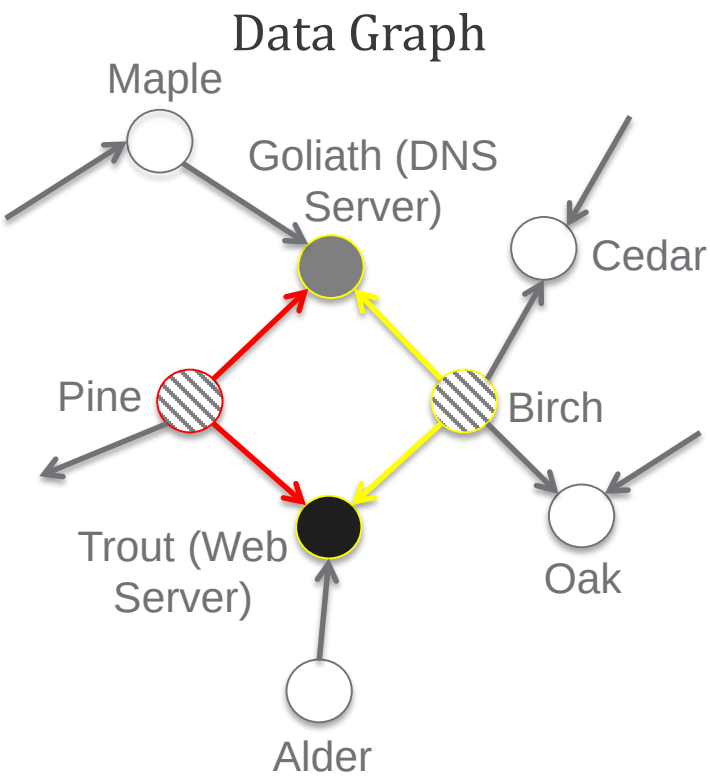


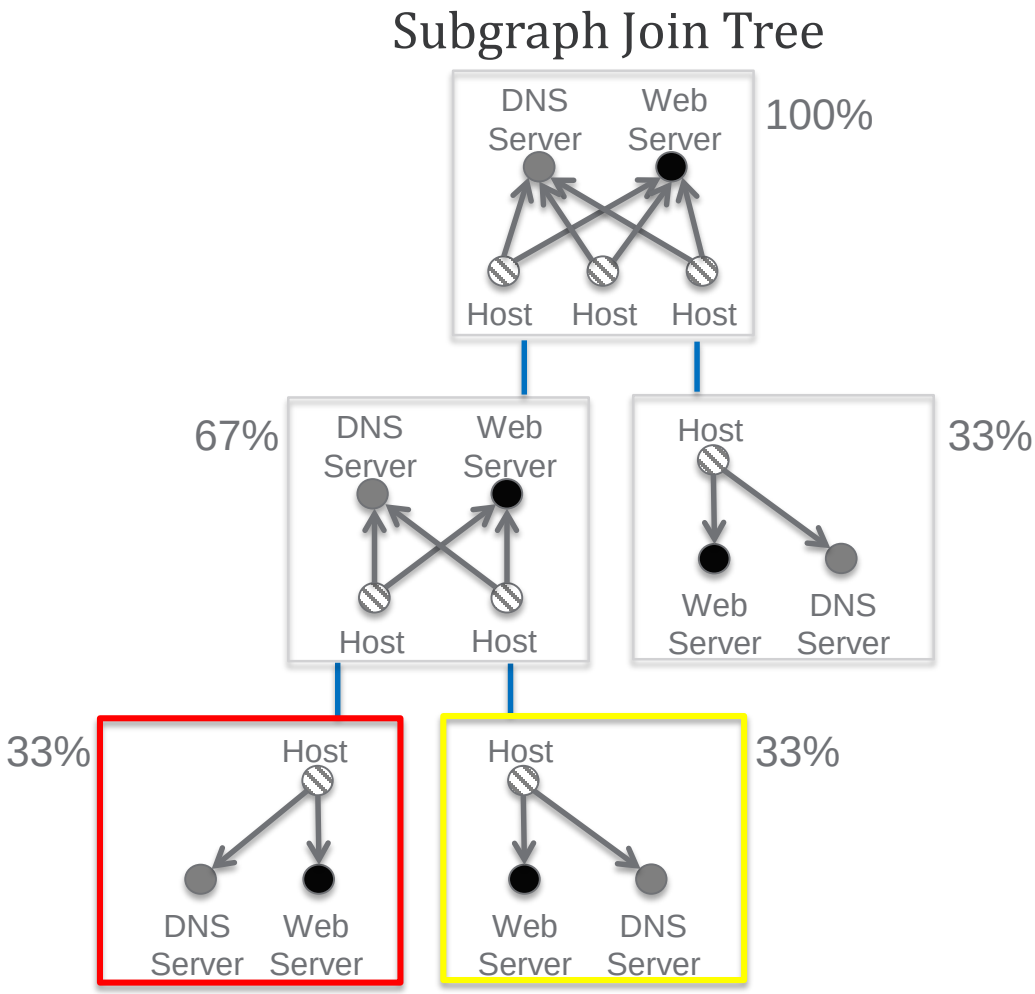
Emerging Graph Pattern Algorithm in Action





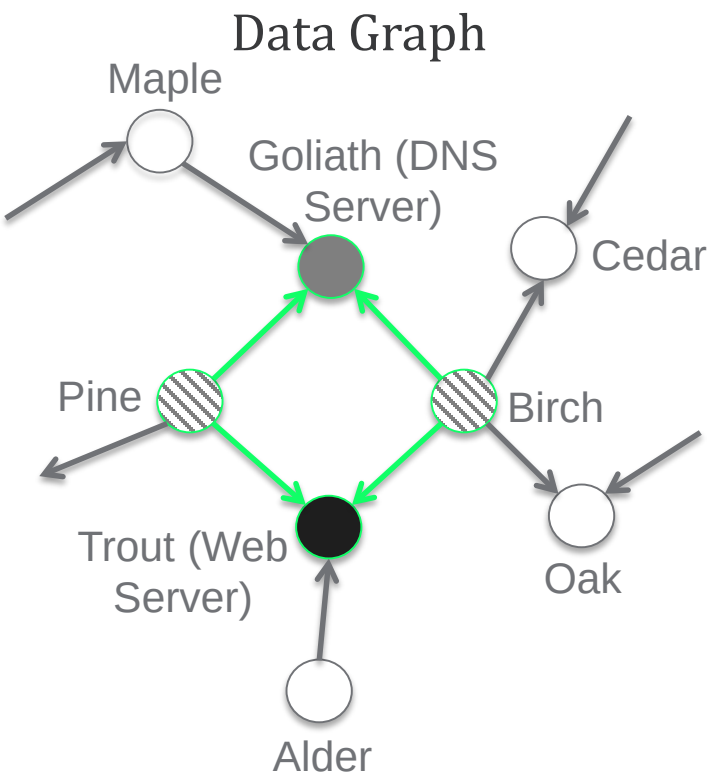
Emerging Graph Pattern Algorithm in Action

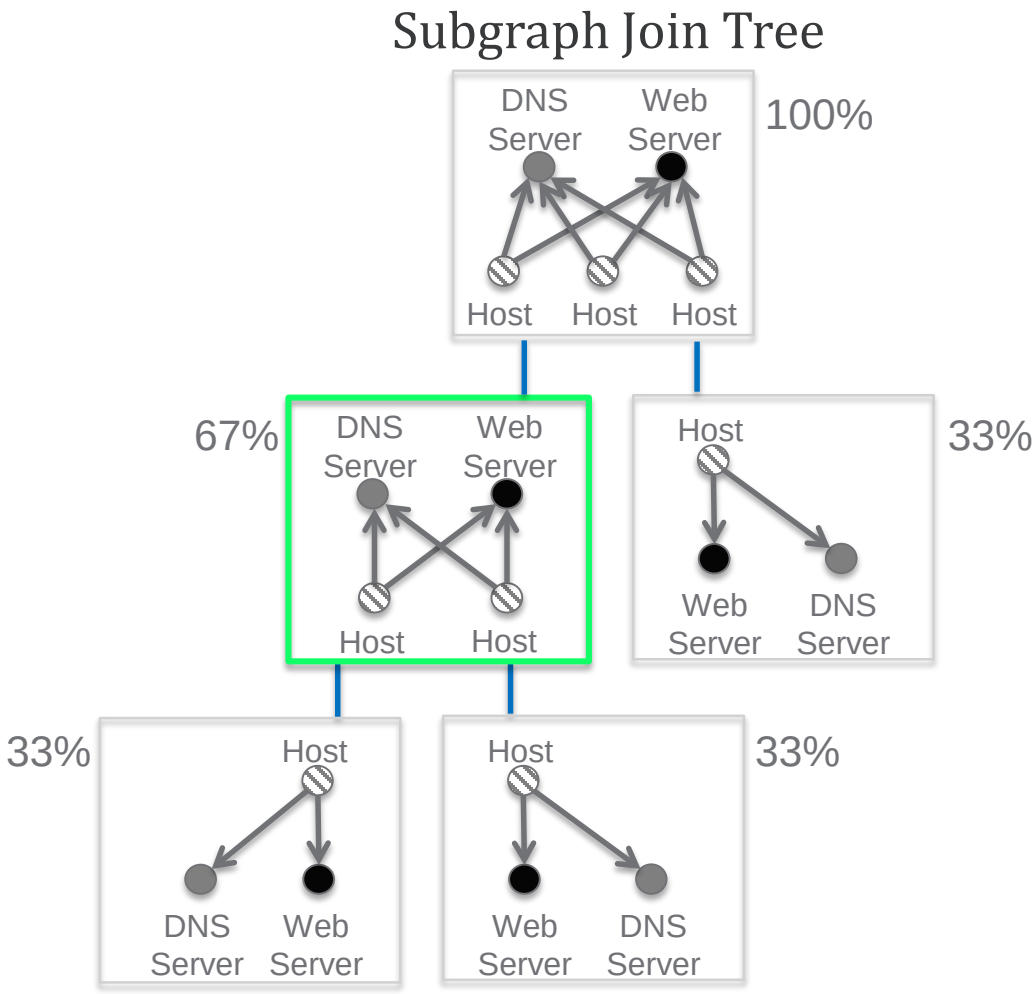




Emerging Graph Pattern Algorithm in Action

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

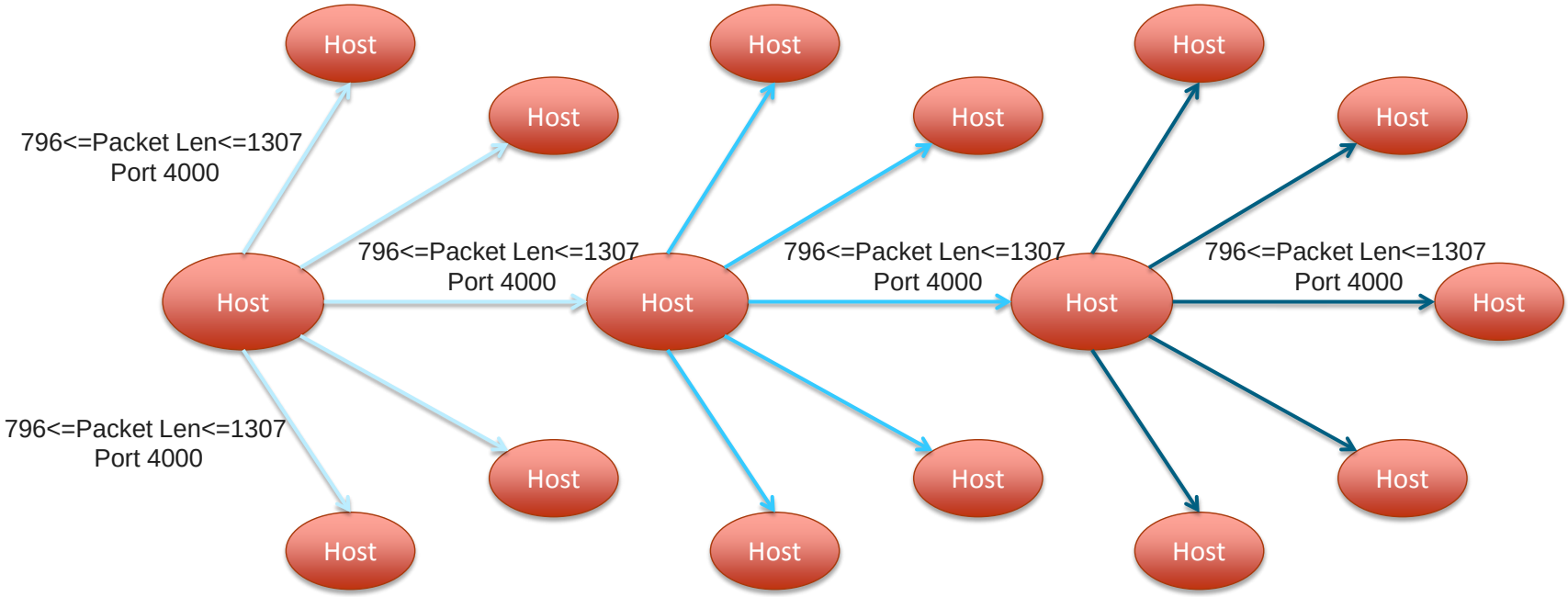




Detecting Emerging Cyber Attacks

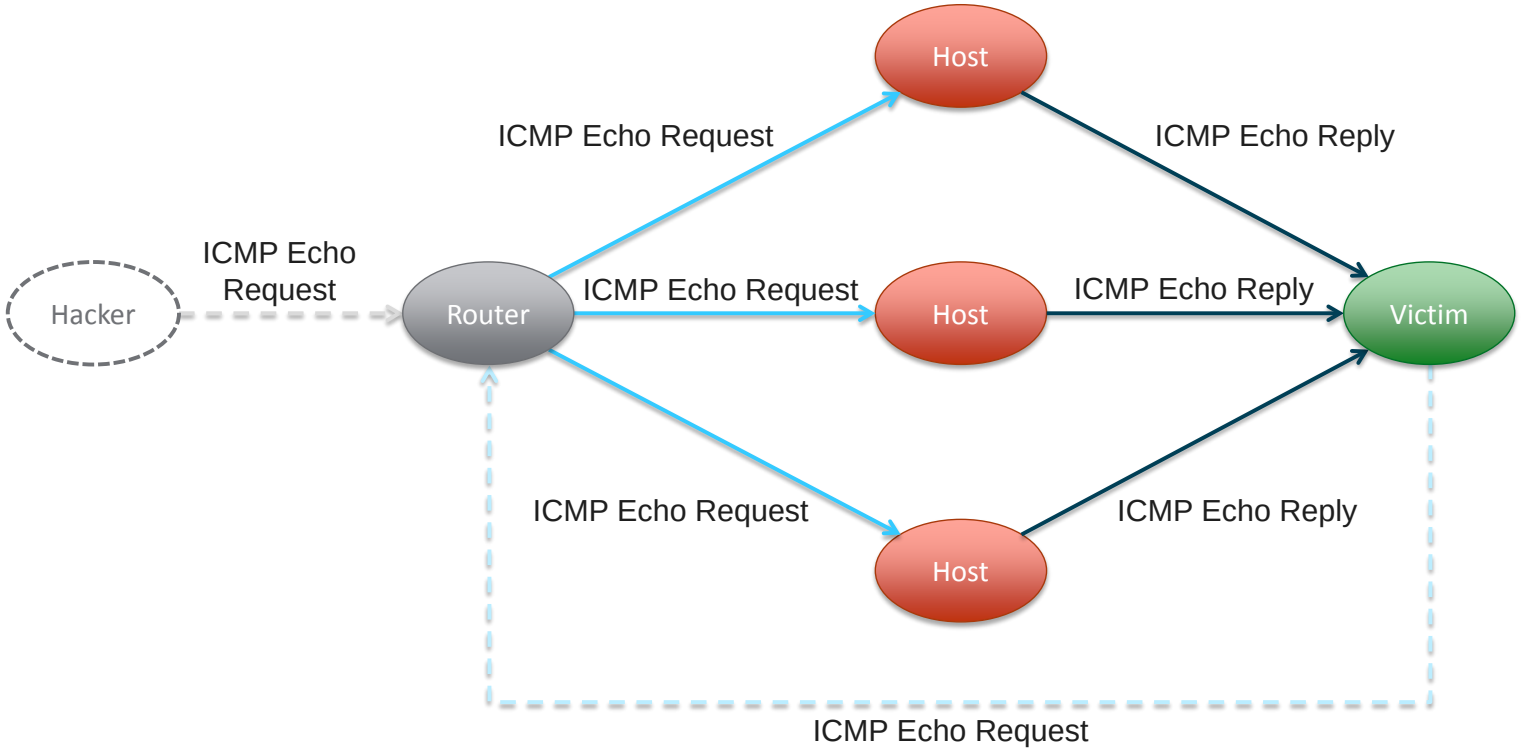
- Developing emerging subgraph pattern algorithm in a package we call StreamWorks to detect cyber intrusions and attacks in computer network traffic
- Constructing set of cyber attack graph patterns related to network scans, reflector attacks, flood attacks, viruses, worms, etc. in collaboration with PNNL cybersecurity analysts
- Utilizing anonymized internet traces data curated by CAIDA (The Cooperative Association for Internet Data Analysis) at SDSC/UCSD and simulated intrusion detection datasets from the University of New Brunswick's Information Security Centre of Excellence

Witty Worm



- Internet worm that began to spread on March 19, 2004
- Targeted buffer overflow vulnerability in internet security systems (ISS) products
- Payload contained phrase “(^.^) insert witty message here (^.^)”
- Attacked port 4000 with packets of sizes between 796 and 1307

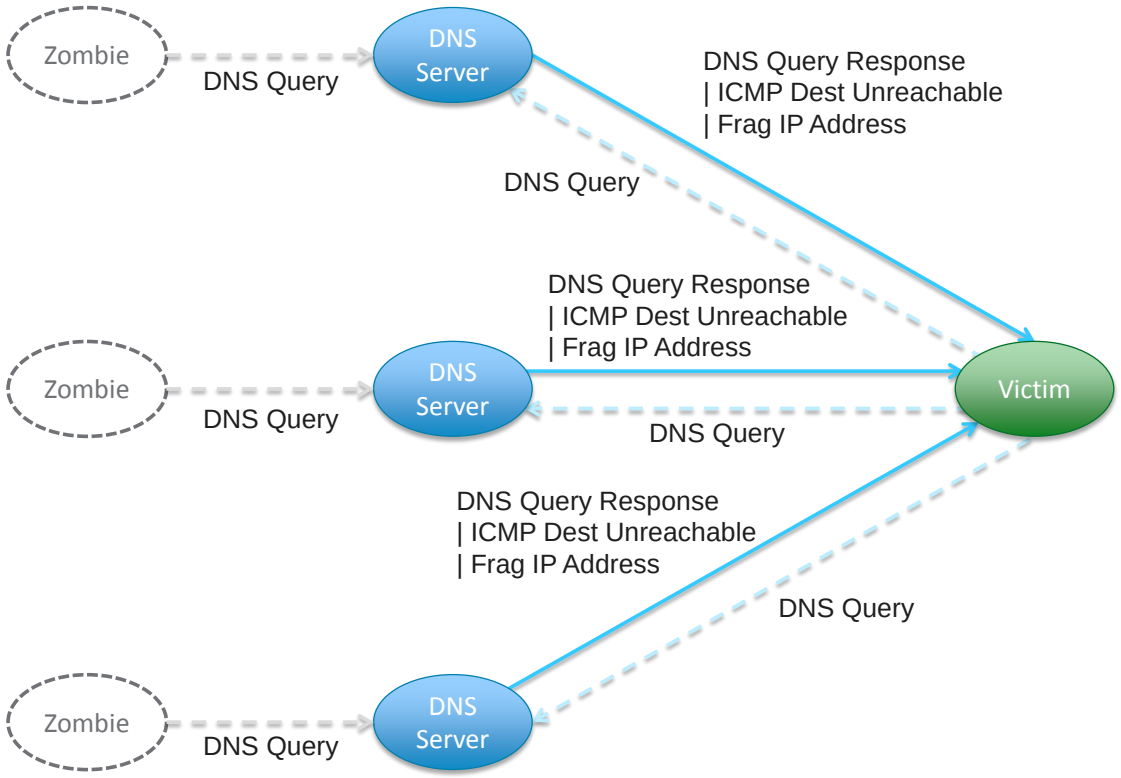
Distributed Denial-of-Service Smurf Attack



- Attacker sends packets to broadcast IP address with spoofed source address of victim's
- Packets delivered to intermediate hosts
- Intermediate hosts reply to return address of victim

Distributed Denial-of-Service DNS Amplification Attack

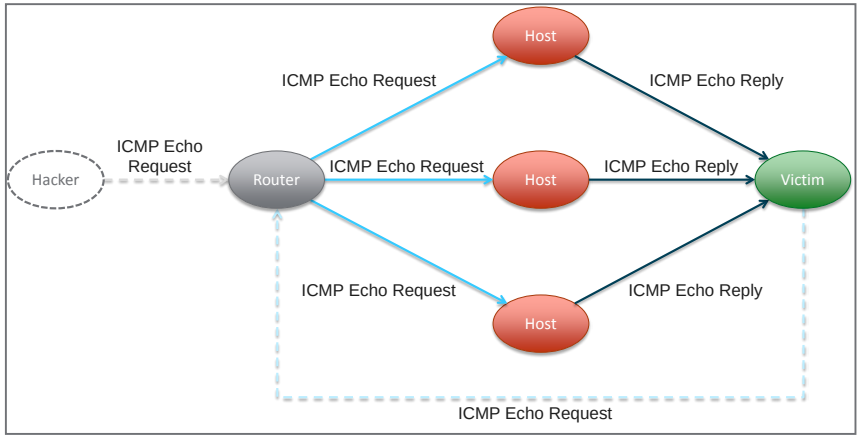
- Agents or zombies generate large number of DNS requests with spoofed source address
- DNS servers send 3 different types of responses to victim
- DNS response packets may be significantly larger than DNS request packets



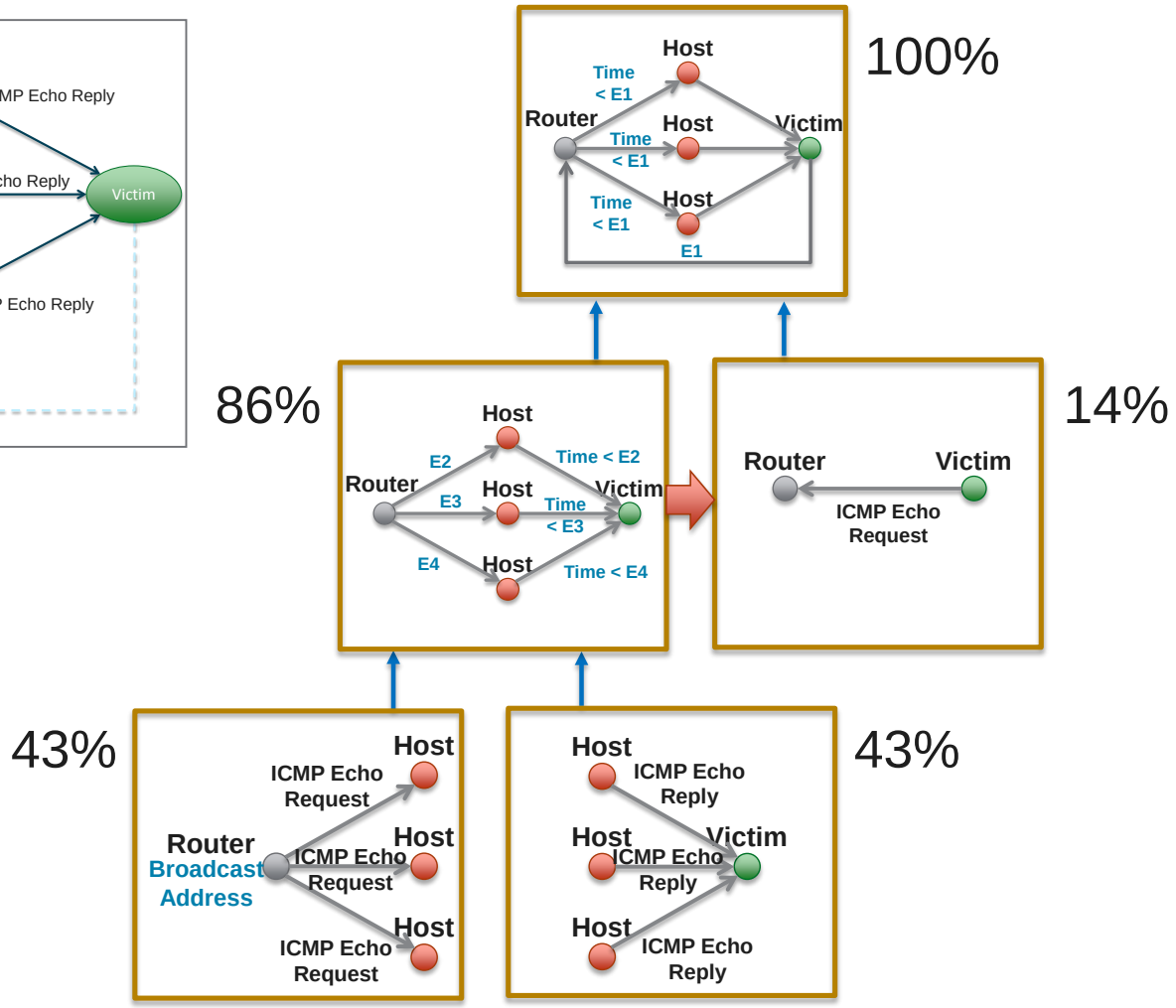
Subgraph Join Tree for DDoS Smurf Attack

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

Cyberattack Pattern



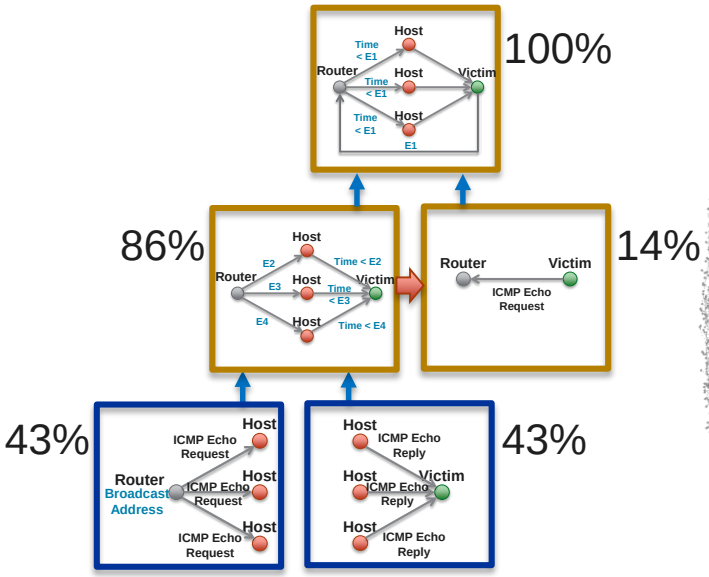
Subgraph Join Tree (Breadth-First)

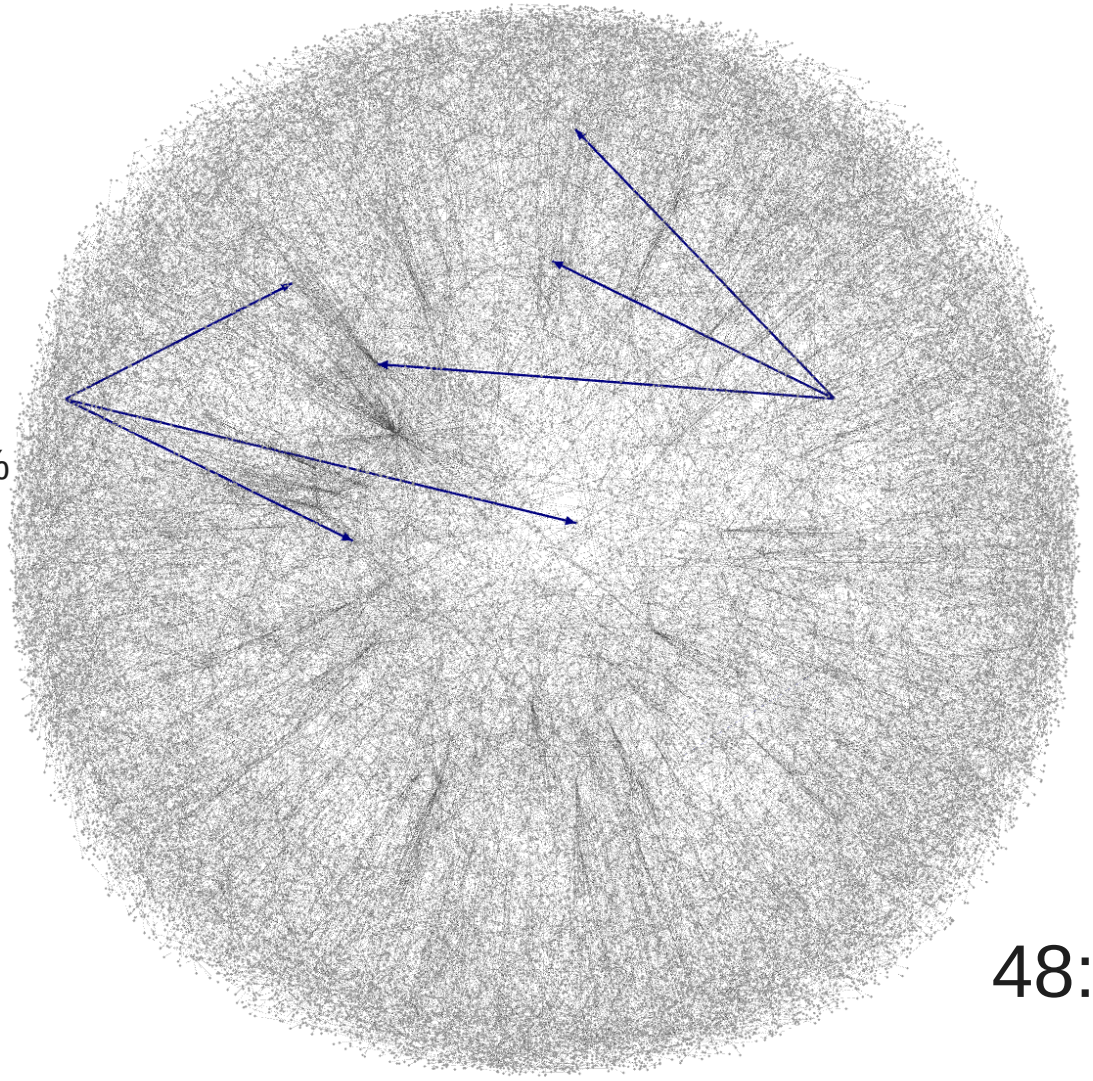


DDoS Smurf Attack Query

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

Breadth-First SJT



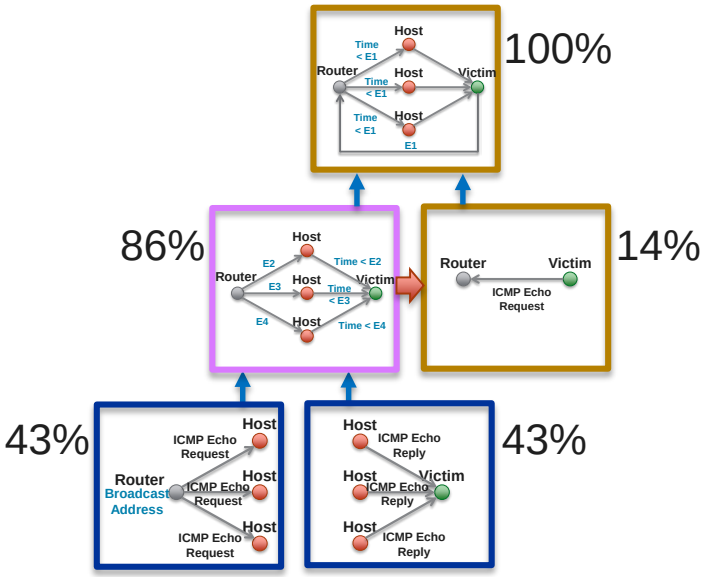


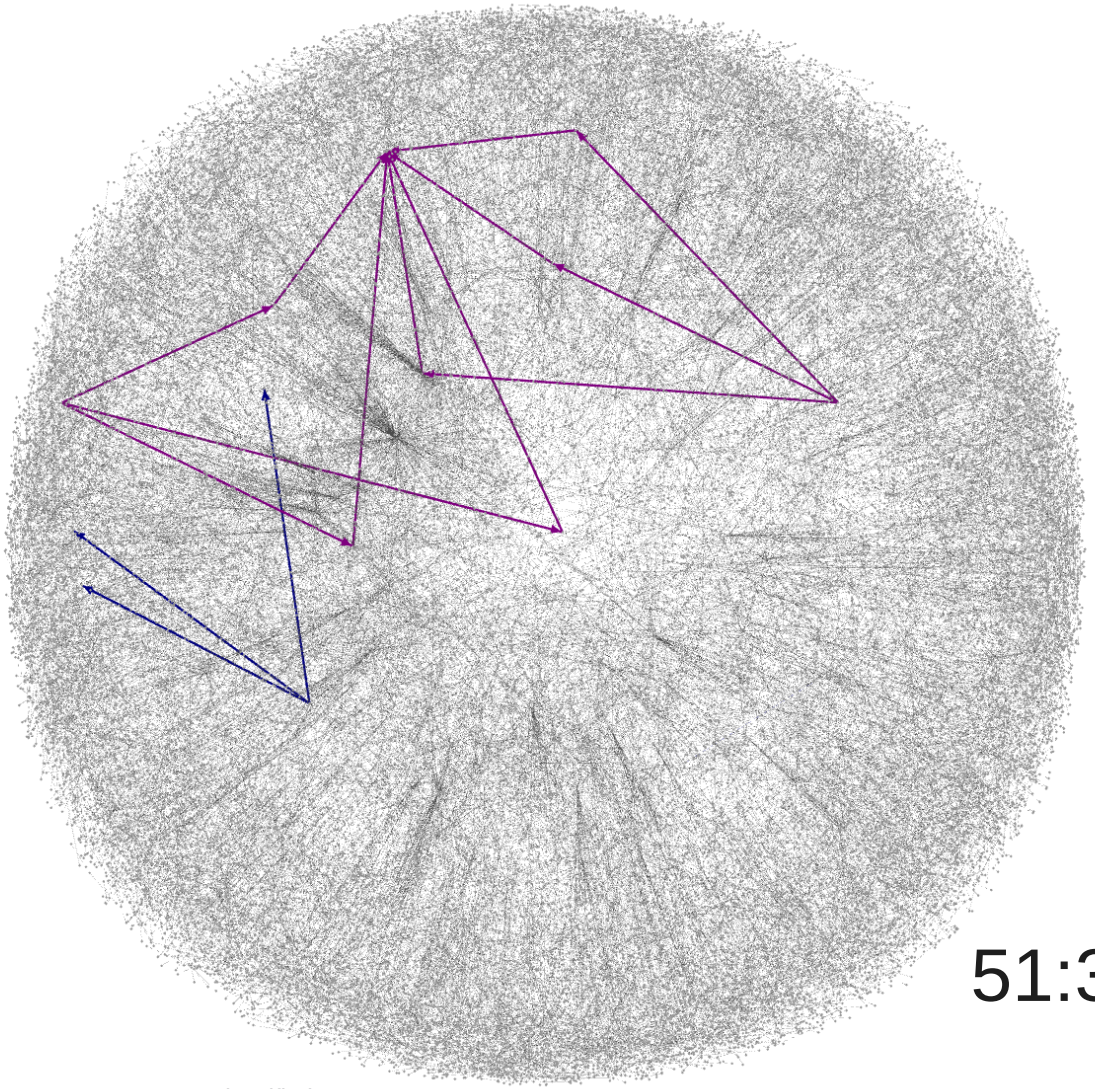
48:06

DDoS Smurf Attack Query

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

Breadth-First SJT



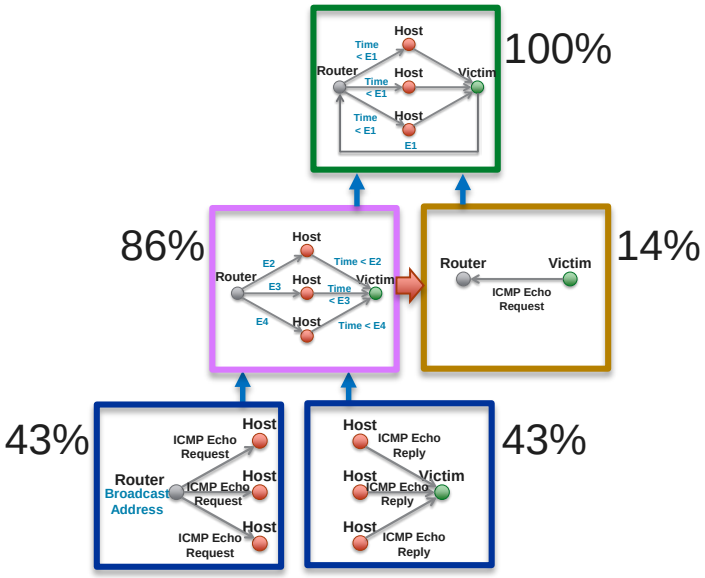


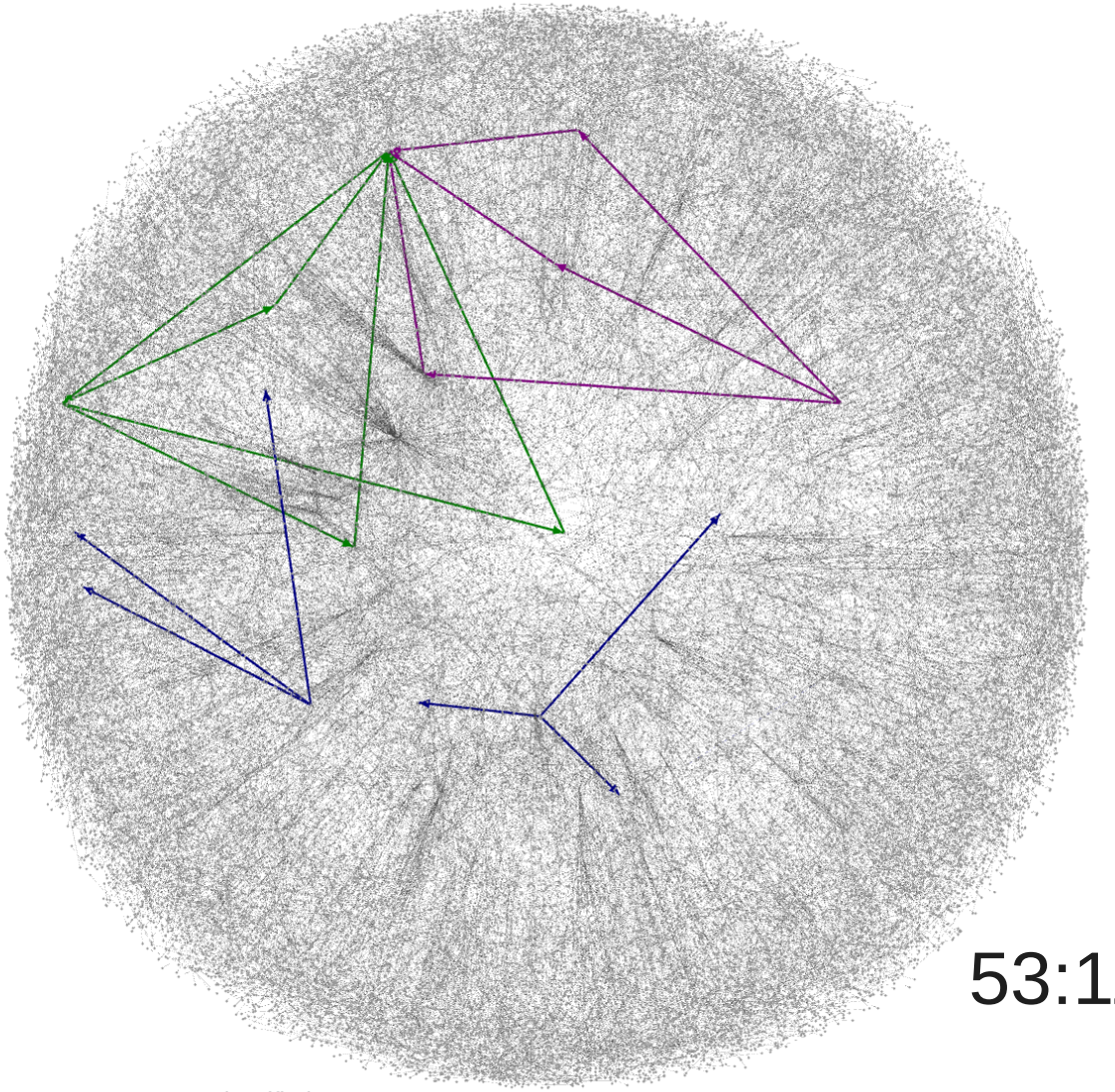
51:39

DDoS Smurf Attack Query

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

Breadth-First SJT

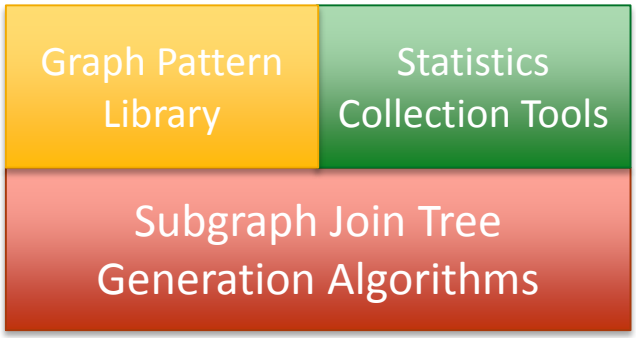




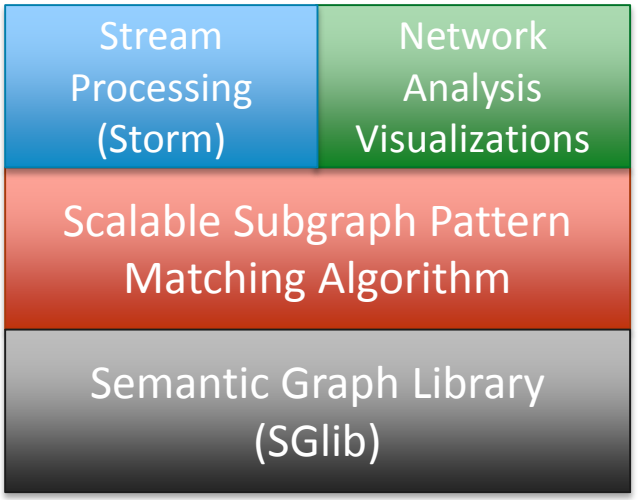
53:11

StreamWorks Components

Graph Pattern Definition and Join Tree Modeling



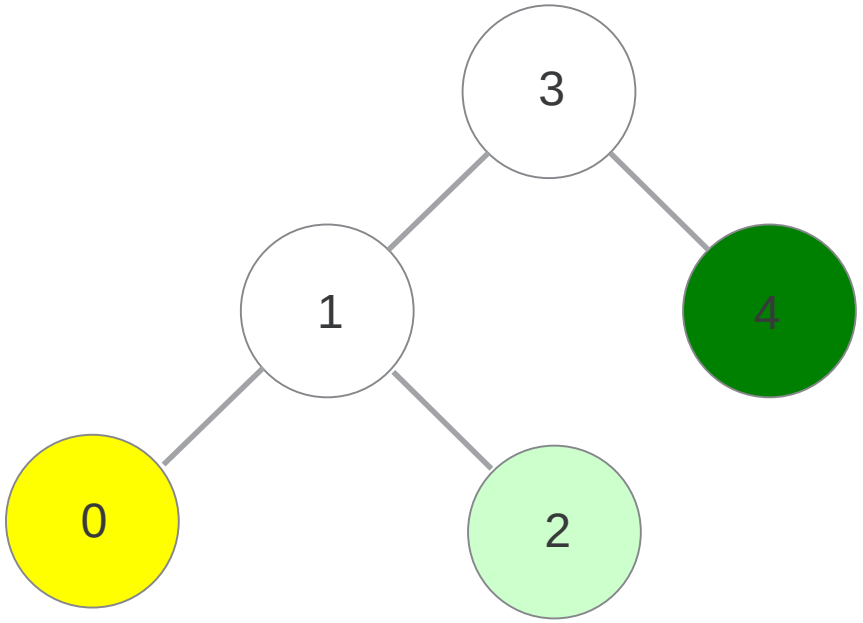
Run-Time System



Scalable Subgraph Matching Algorithm

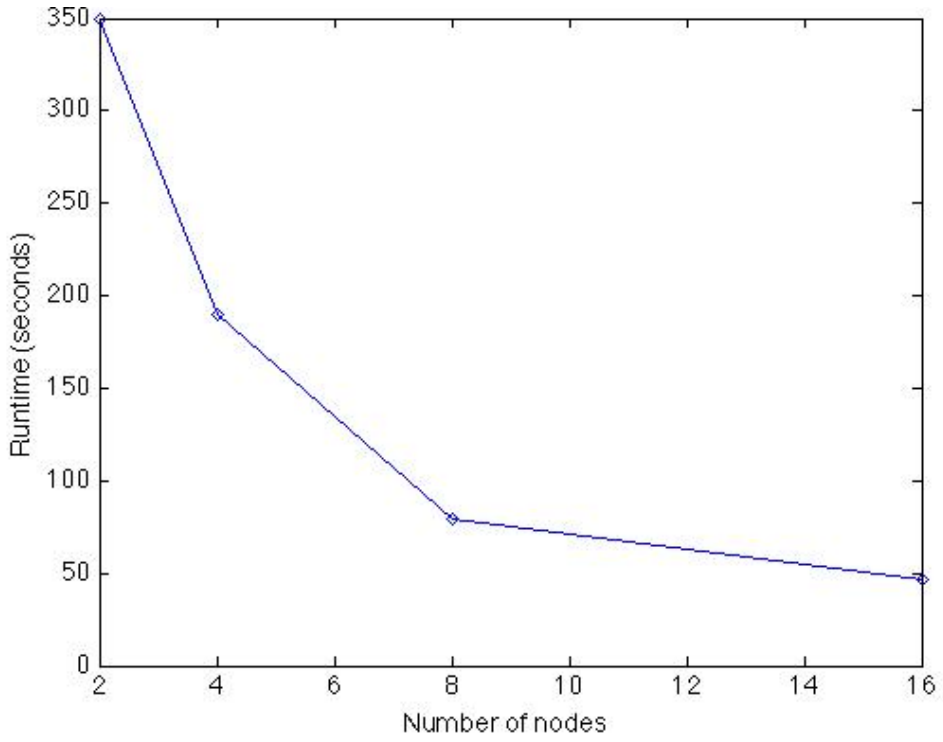
Distributed Implementation of Dynamic Graph Search

- Update the distributed graph with new edges in parallel
- Search the updated graph in parallel for unique sub-queries
  - Colors represents unique sub-queries in SJ-Tree
  - Each node in SJ-Tree maintains a match collection
  - Each nodes receive the new set of matches
- Perform parallel hash join of new and old matches in SJ-Tree at each level



Scalable Subgraph Matching Algorithm

Scalability Results for Distributed Implementation of Dynamic Graph Search

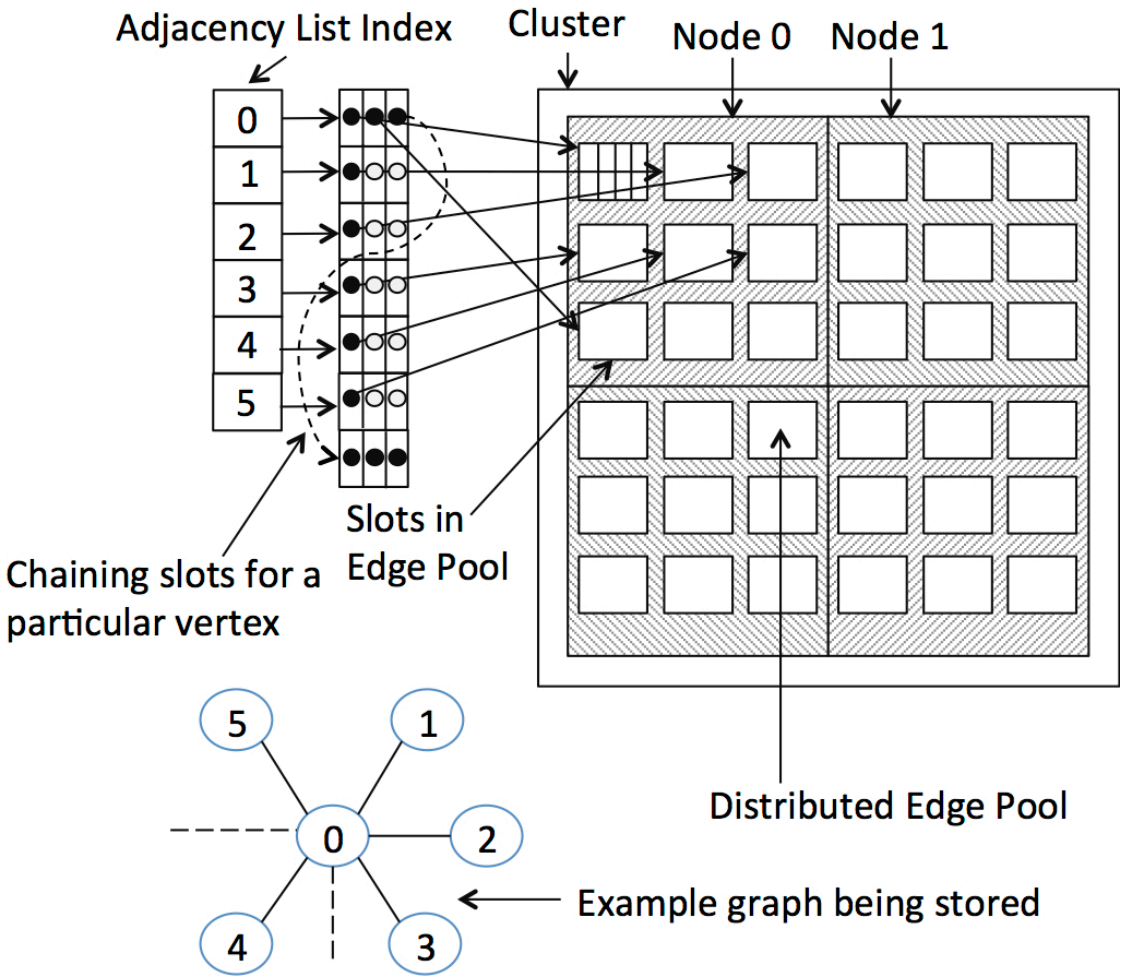


CAIDA Network Traffic
(2.49M nodes, 19.55M edges)

PNNL institutional computing (PIC) cluster: 692 nodes, AMD Interlagos processors, dual socket, 16 cores per socket, 64 GB memory per node, QDR InfiniBand

Scalable Subgraph Matching Algorithm

Distributed Dynamic Graph Data Structure



Scalable Subgraph Matching Algorithm

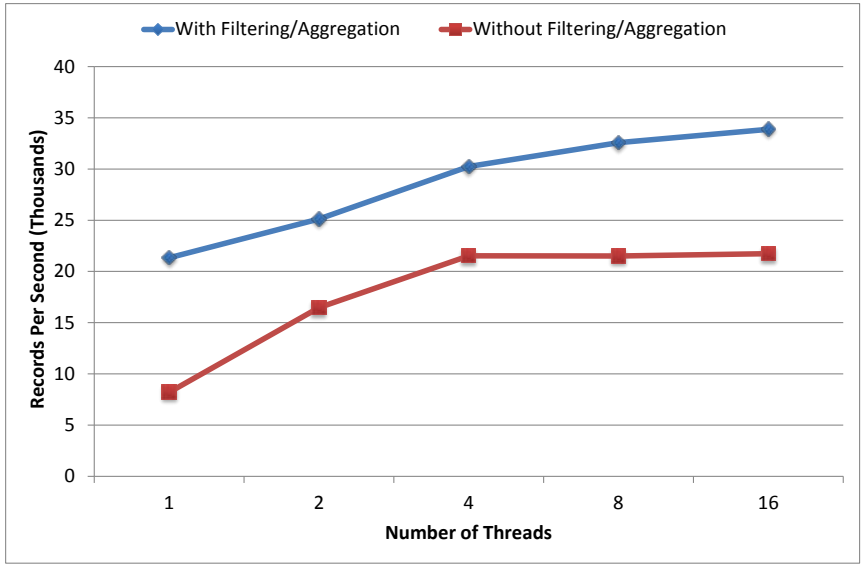
Concurrent Graph Queries via Multiple Subgraph Join Trees

- Conduct parallel searching across all subgraph patterns of all subgraph join trees
- Leverage locality in terms of operations: Identify common subgraph patterns across subgraph join trees and search once for multiple queries
- Leverage locality in terms of data: Identify graph regions that apply to multiple subgraph searches and track and manage once for multiple queries

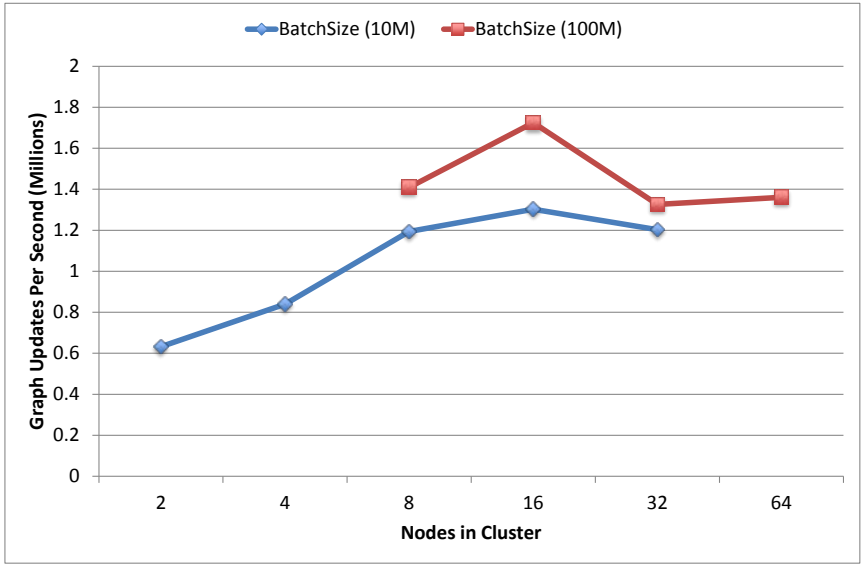
Stream Processing

- Developed various Apache Storm Bolts to filter/aggregate Netflow data
- Filtered and aggregated data is passed to emerging subgraph algorithm using Apache ActiveMQ
- Tuning of primitive subgraph matching between Storm and emerging subgraph algorithm is ongoing

1M Netflow records through Storm, 1 record per message through ActiveMQ (on PIC)

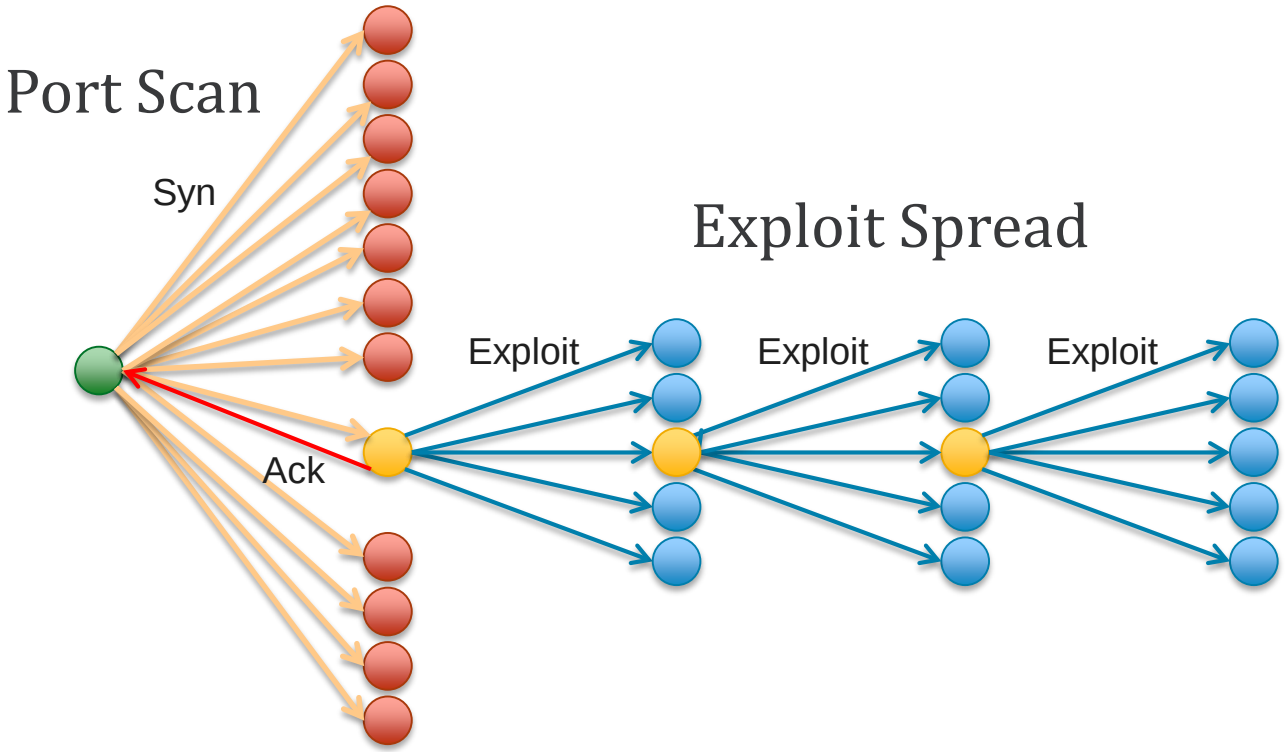


10M and 100M power law graph updates through emerging subgraph algorithm (on PIC)

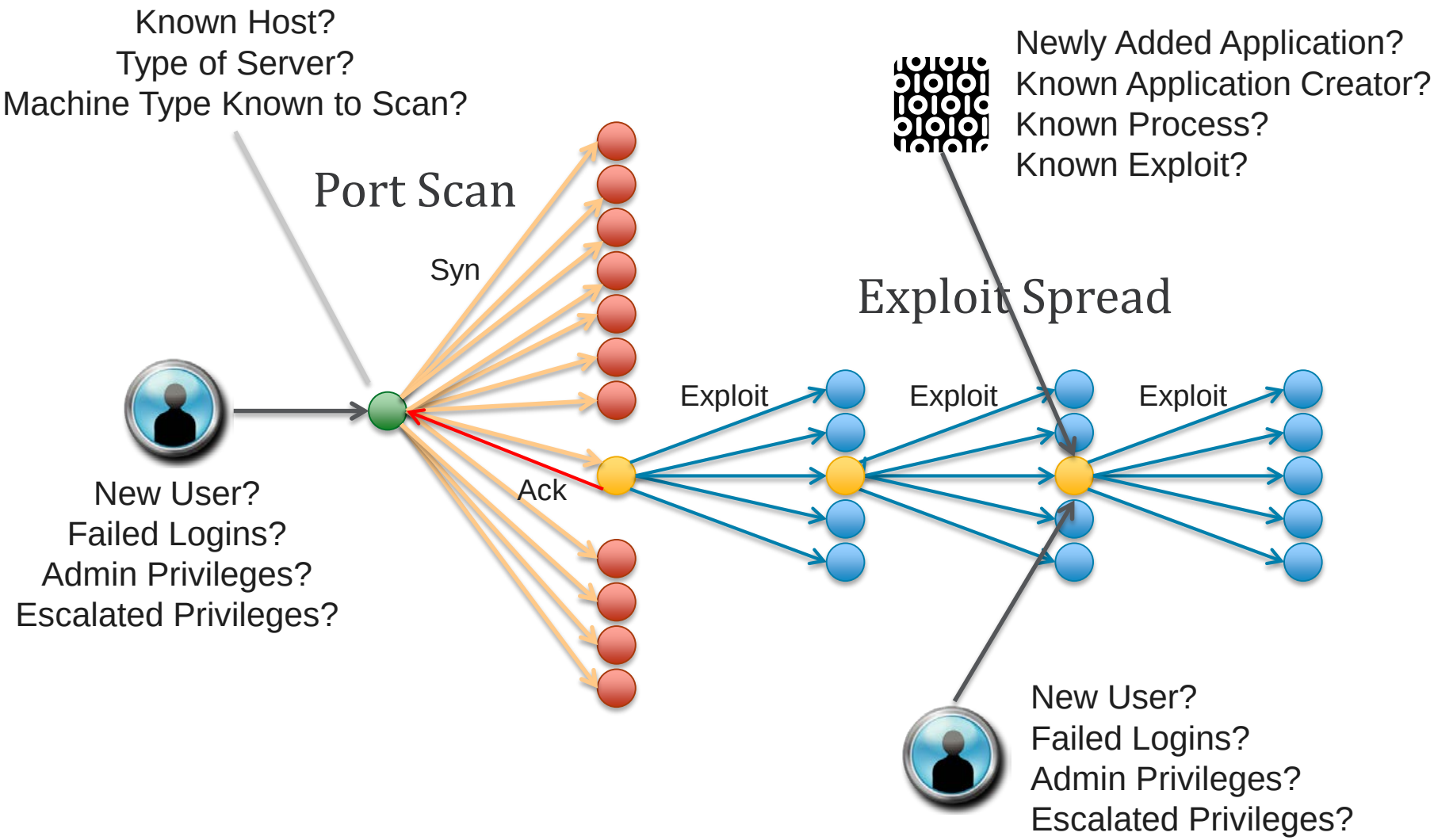


Cyberattack Graph Patterns

Is Netflow data and patterns enough to effectively detect emerging cyberattacks?



Cyberattack Graph Patterns

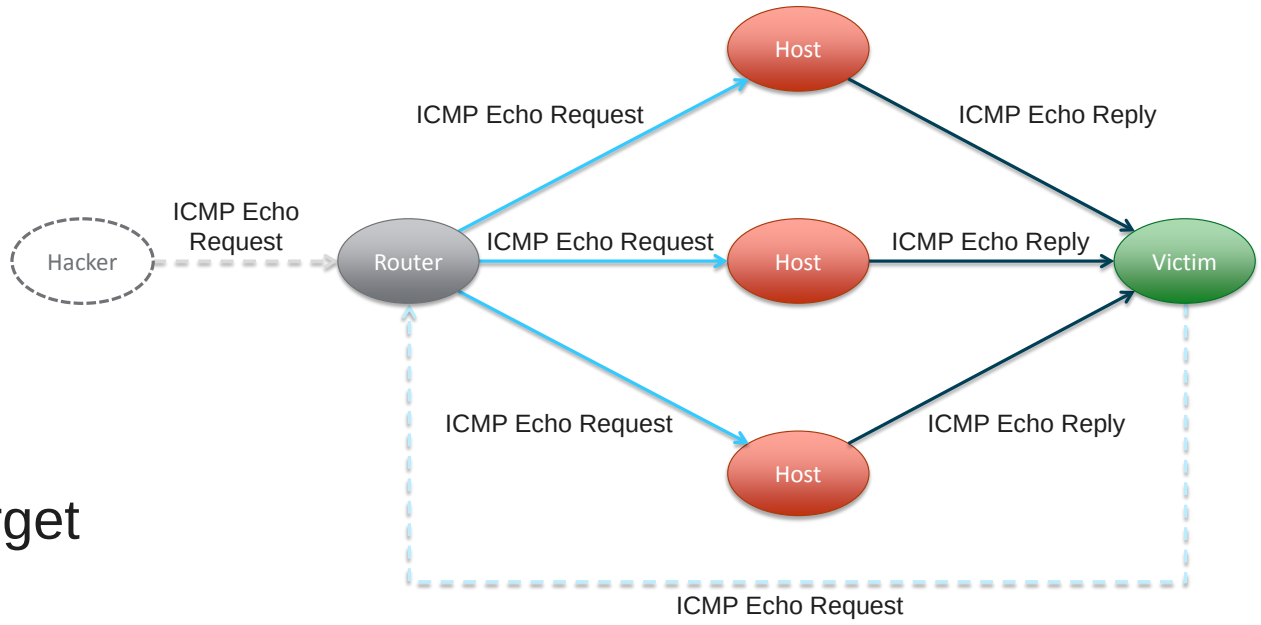


Multi-Source Cyberattack Graph Patterns

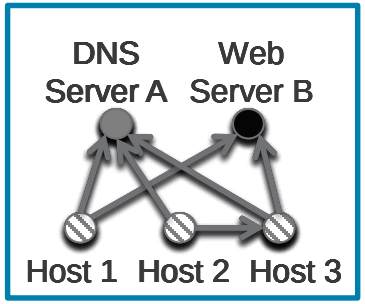
- Look to fuse streaming Netflow data with other streaming data sources such event logs, host scan logs, firewall logs, and anti-malware reports
- Enrich the semantic graph with more attributes to collect information from the fused data stream
- Derive additional candidate graph patterns and their associated subgraph join trees with fuller attributes to better elaborate specific cyberattacks

Automatic Subgraph Join Tree Generation

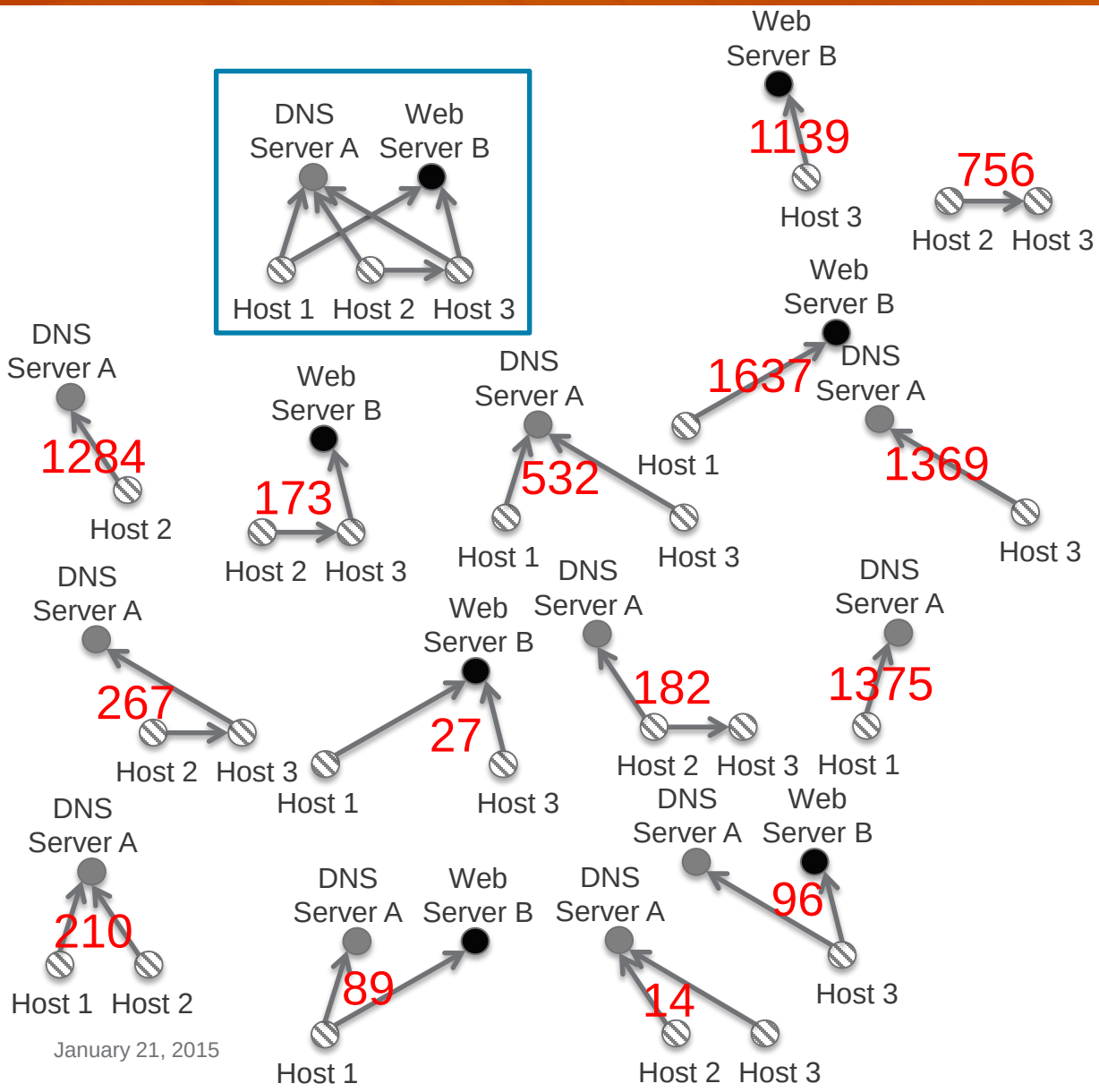
- With known target graph patterns
  - Breadth-first traversal
  - Depth-first traversal
  - Frequency-based
- With unknown target graph patterns
  - Frequency-based



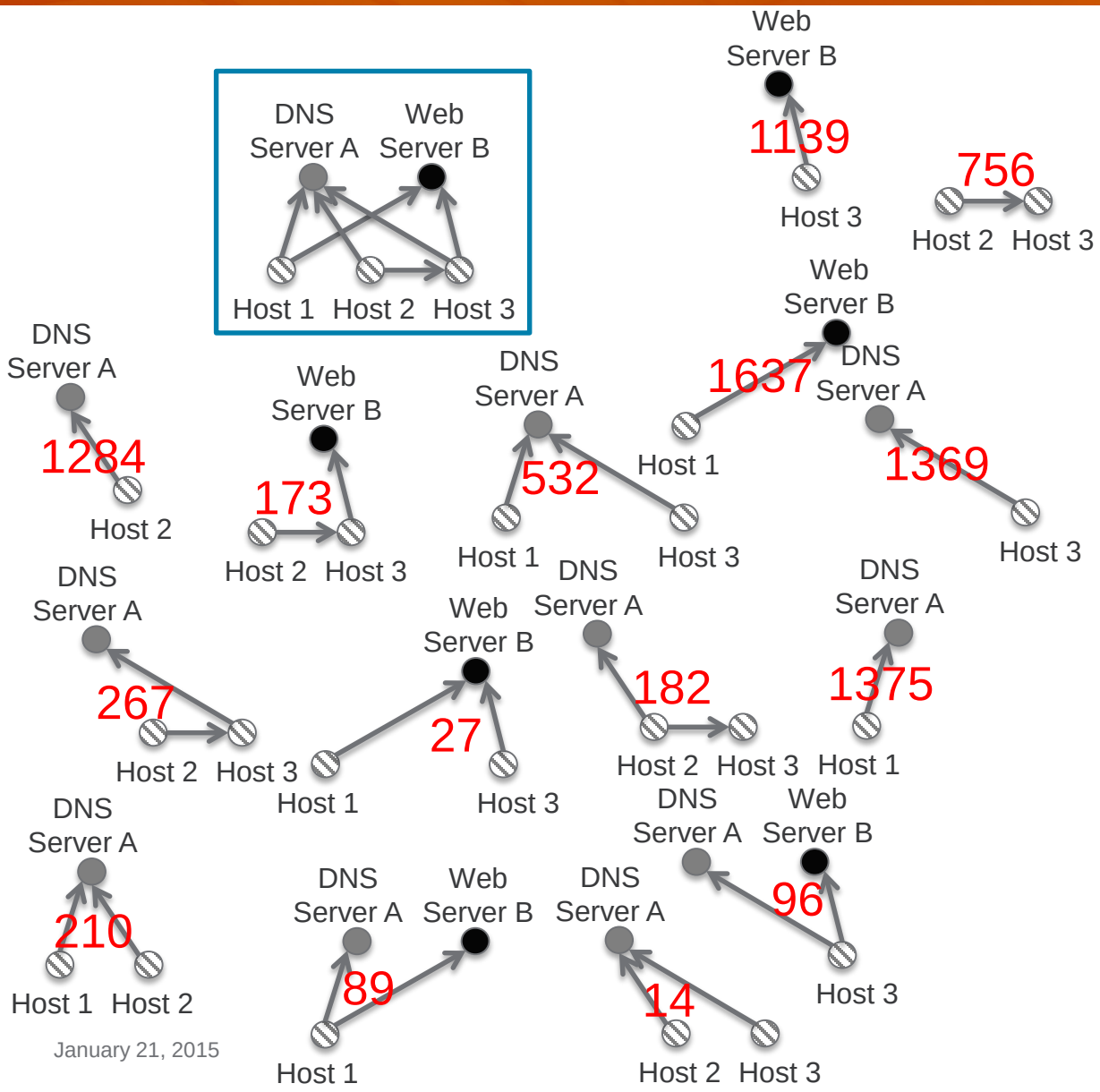
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



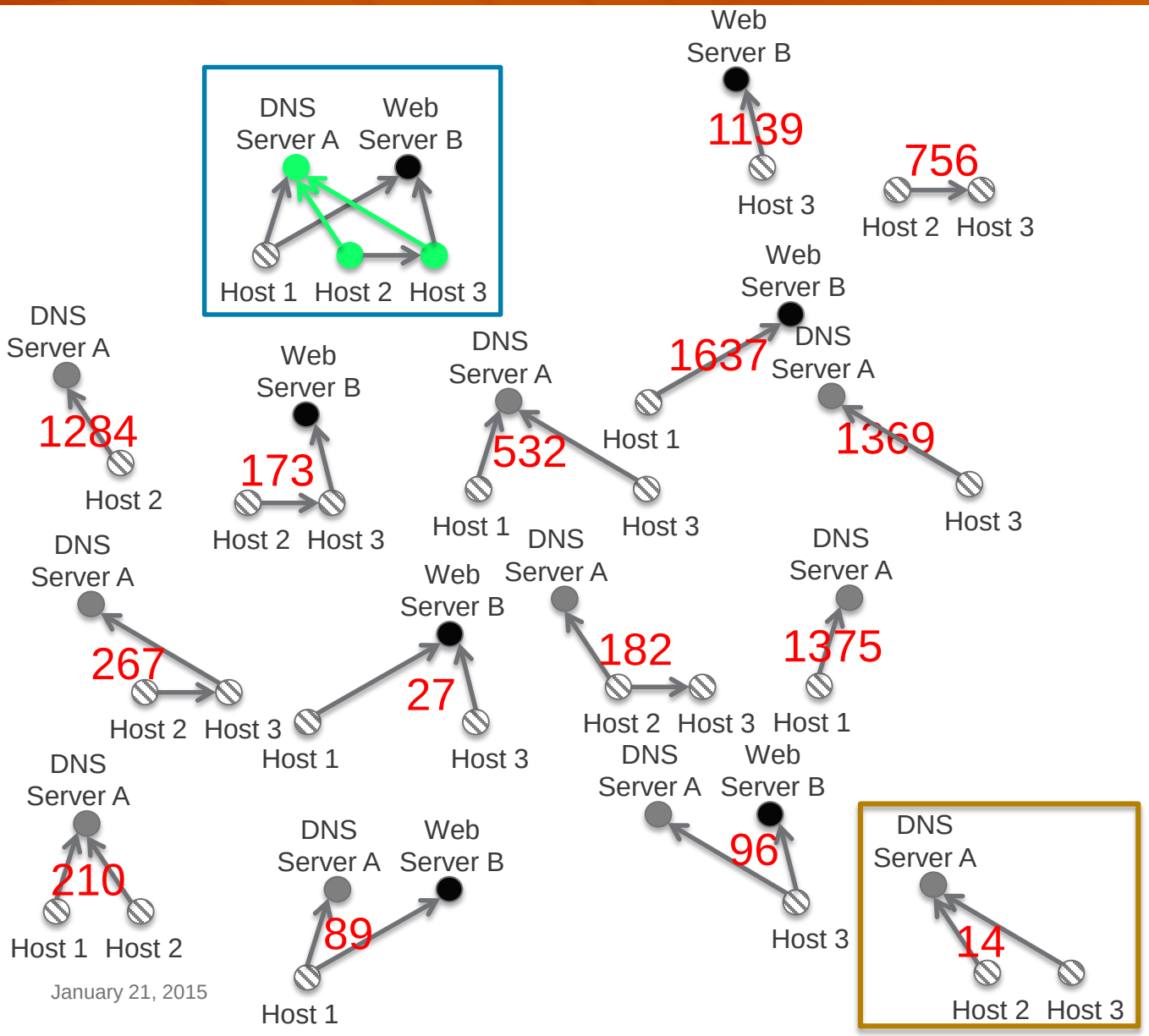
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



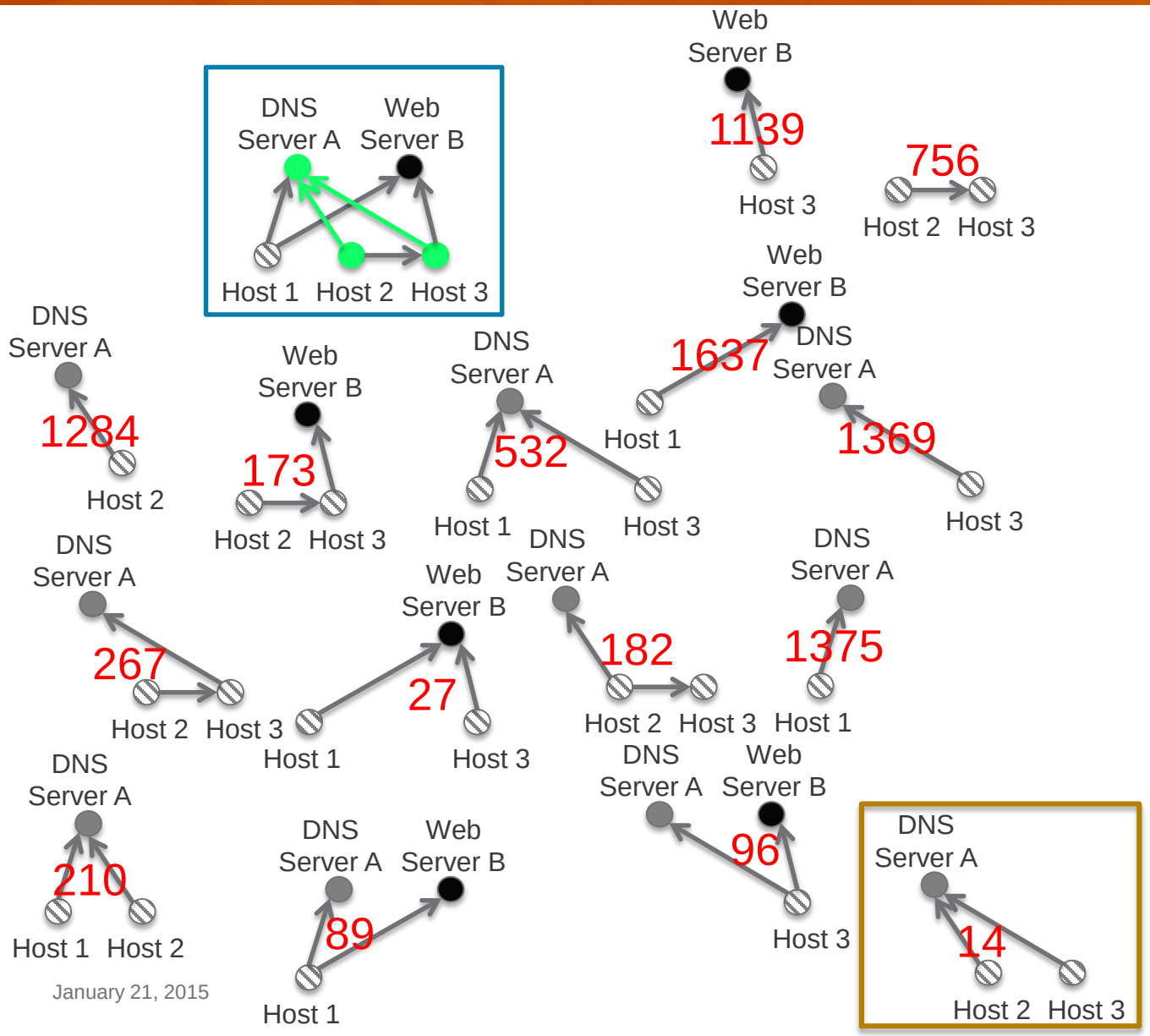
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



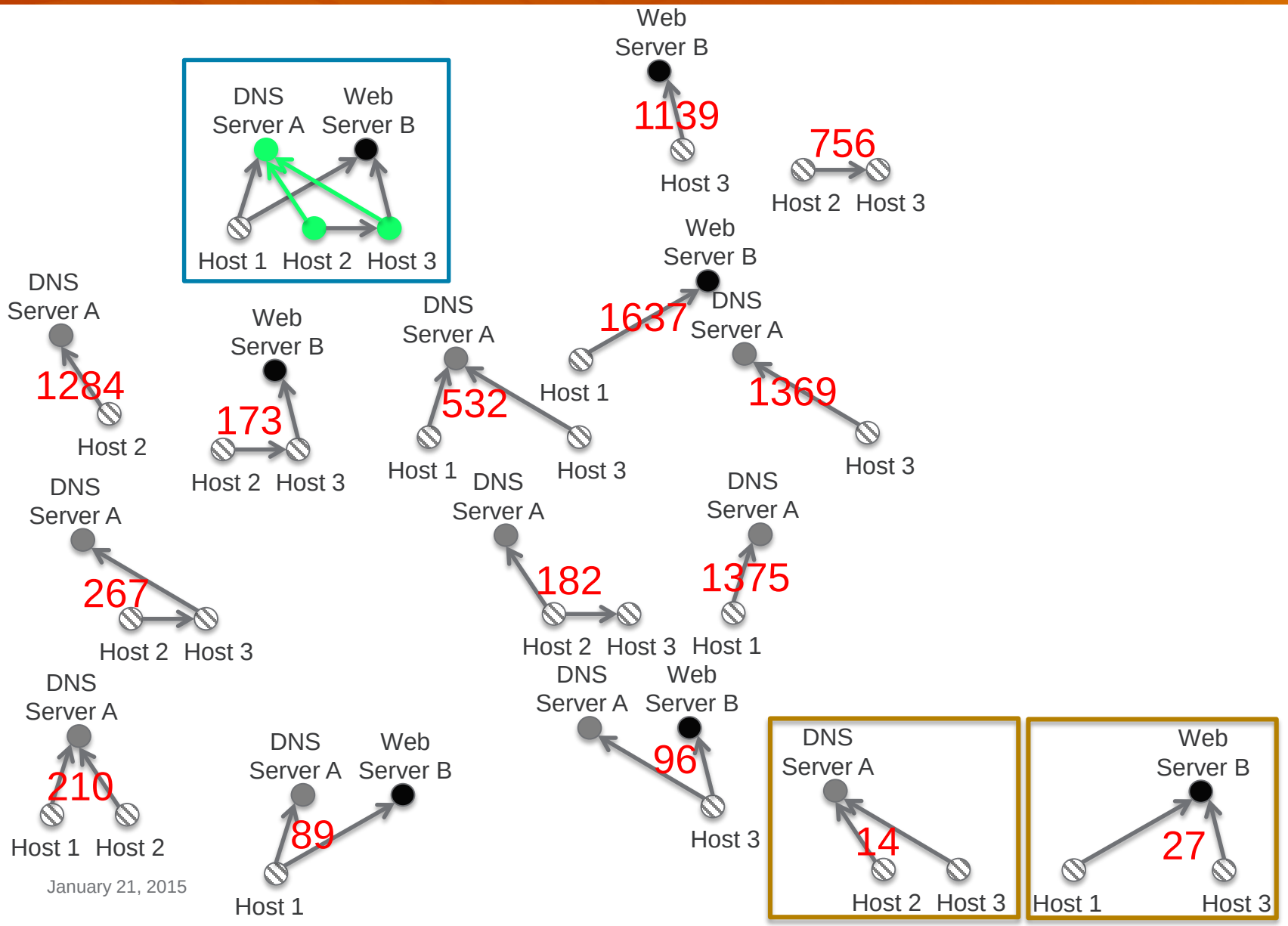
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



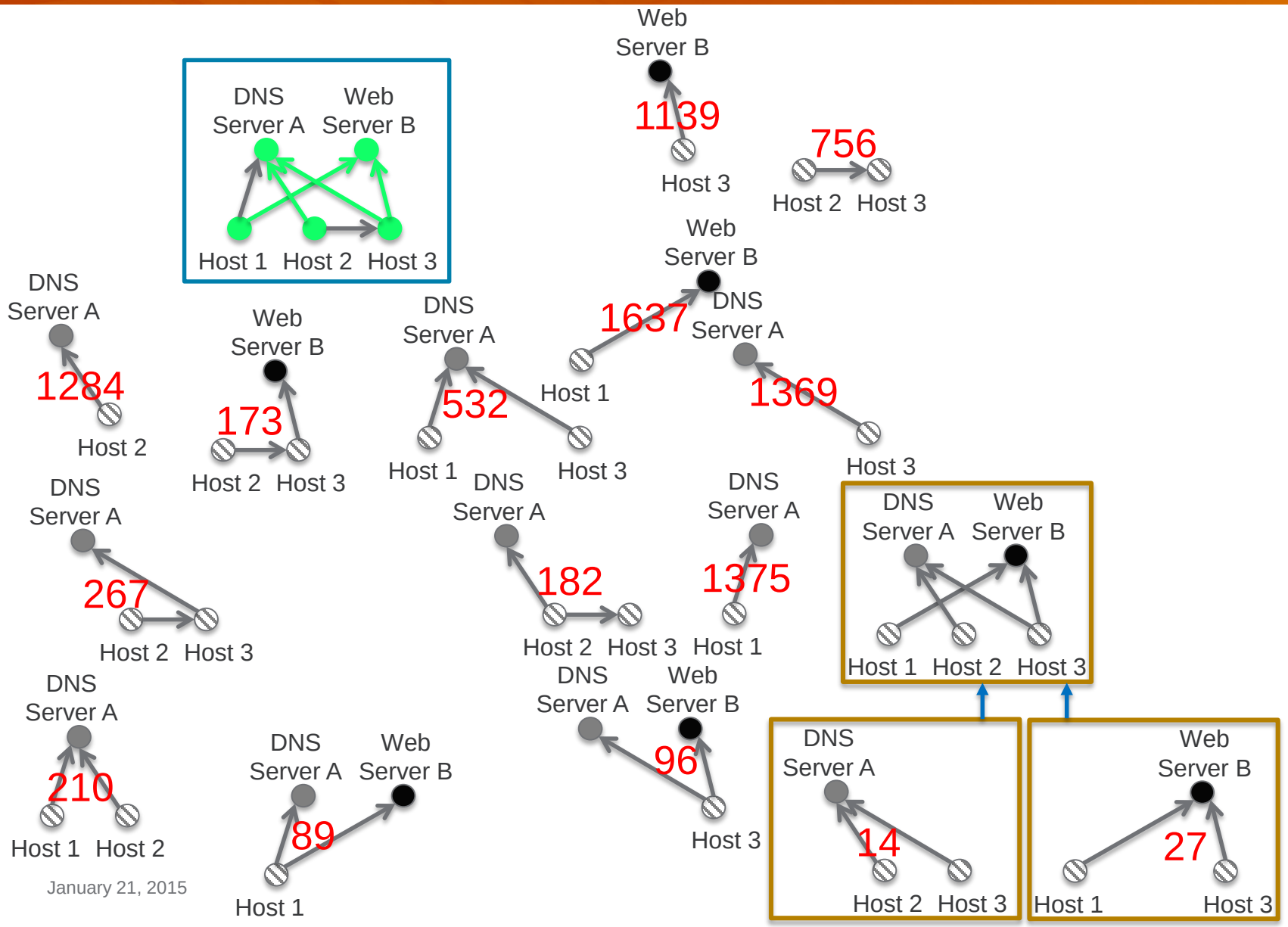
Automatic Join Tree Generation with Known Graph Pattern



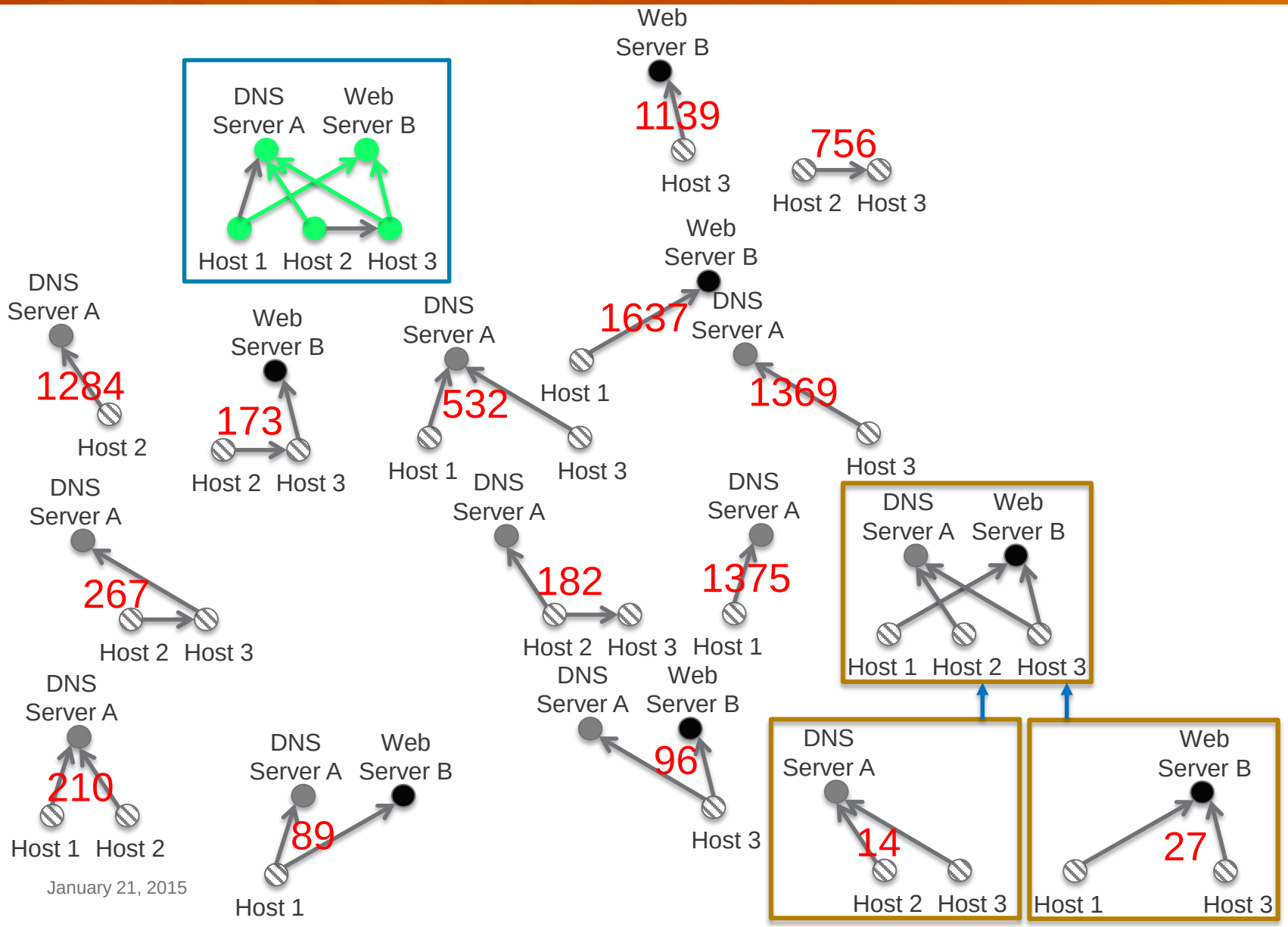
Automatic Join Tree Generation with Known Graph Pattern



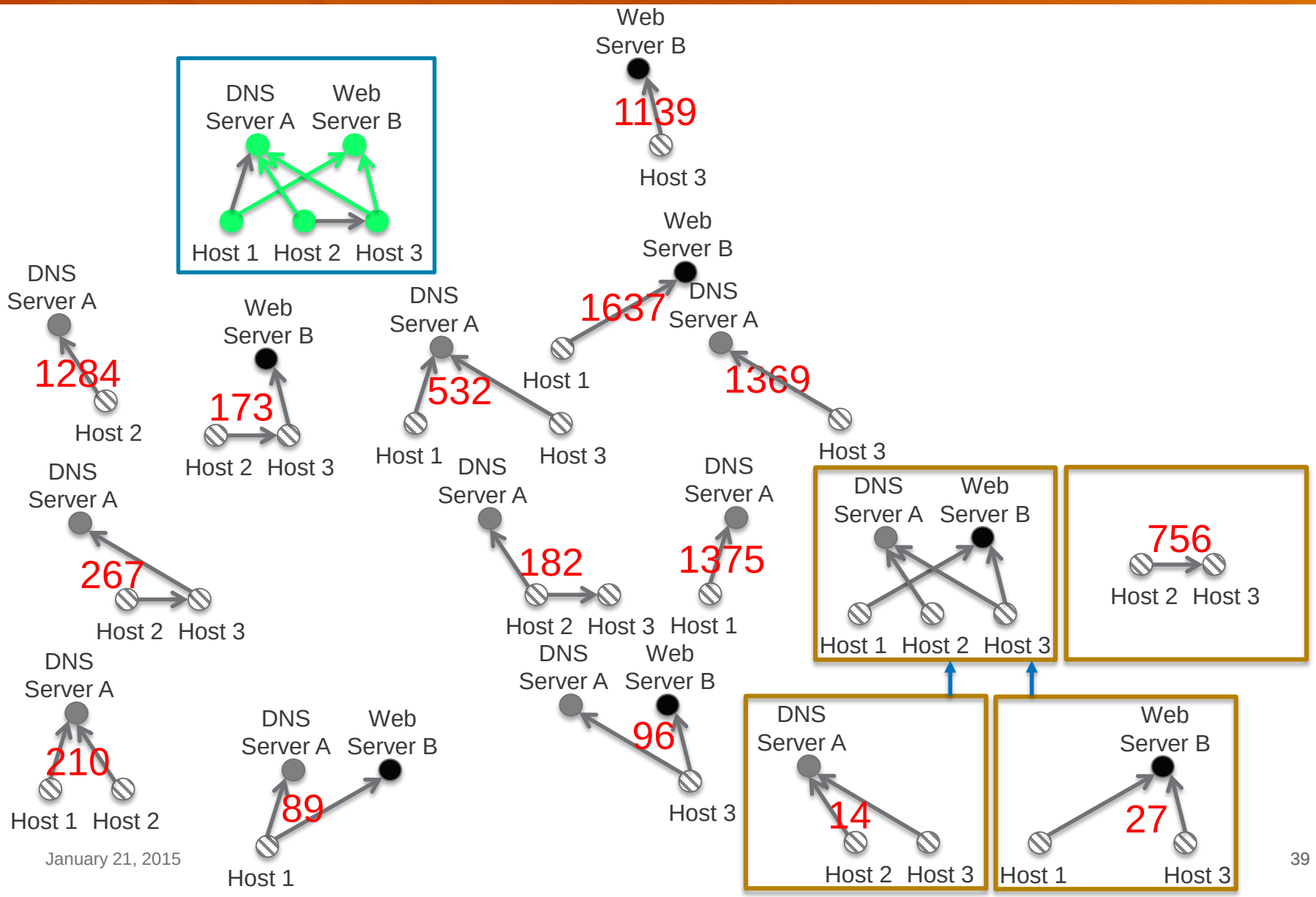
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



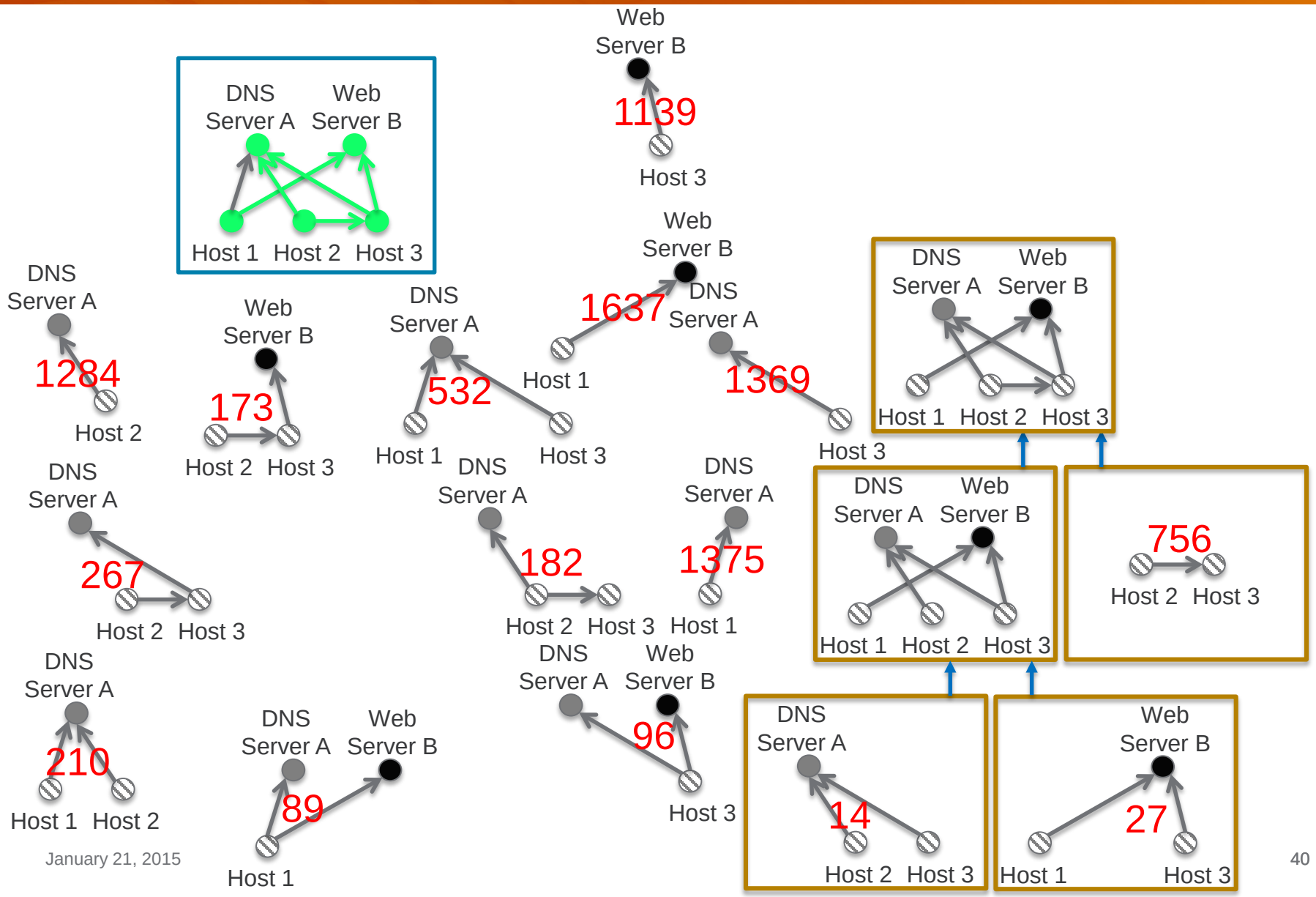
Automatic Join Tree Generation with Known Graph Pattern



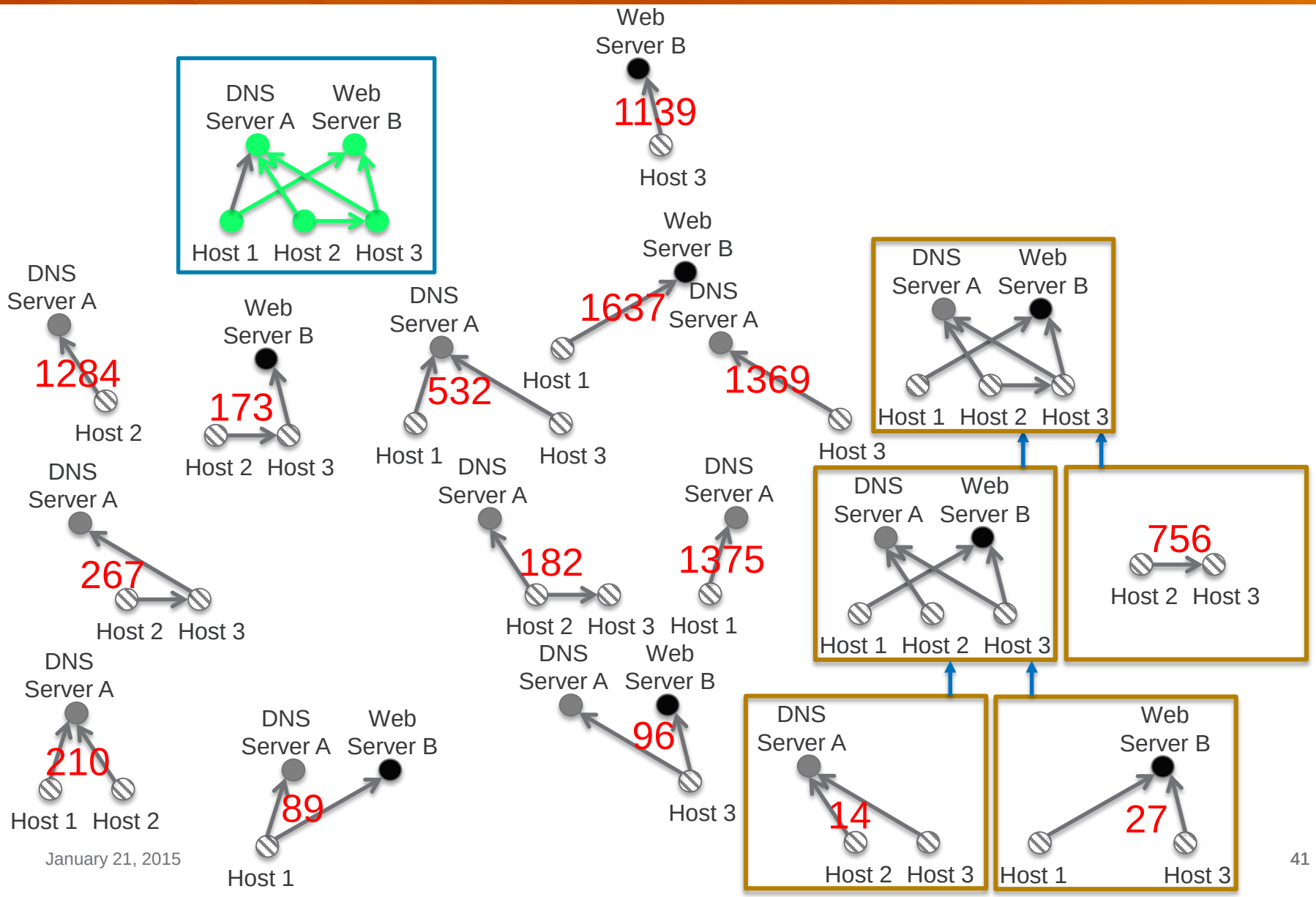
Automatic Join Tree Generation with Known Graph Pattern



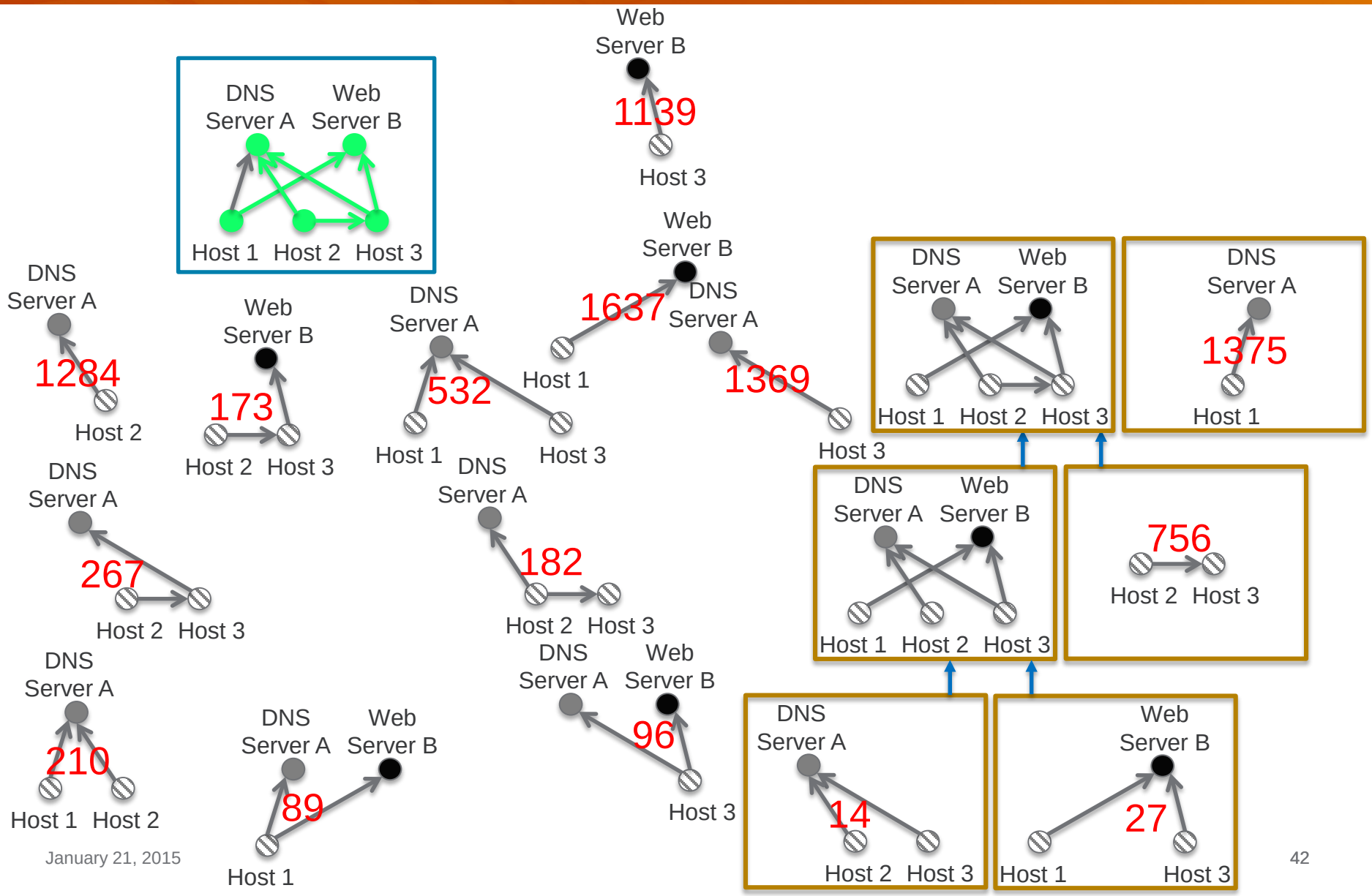
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



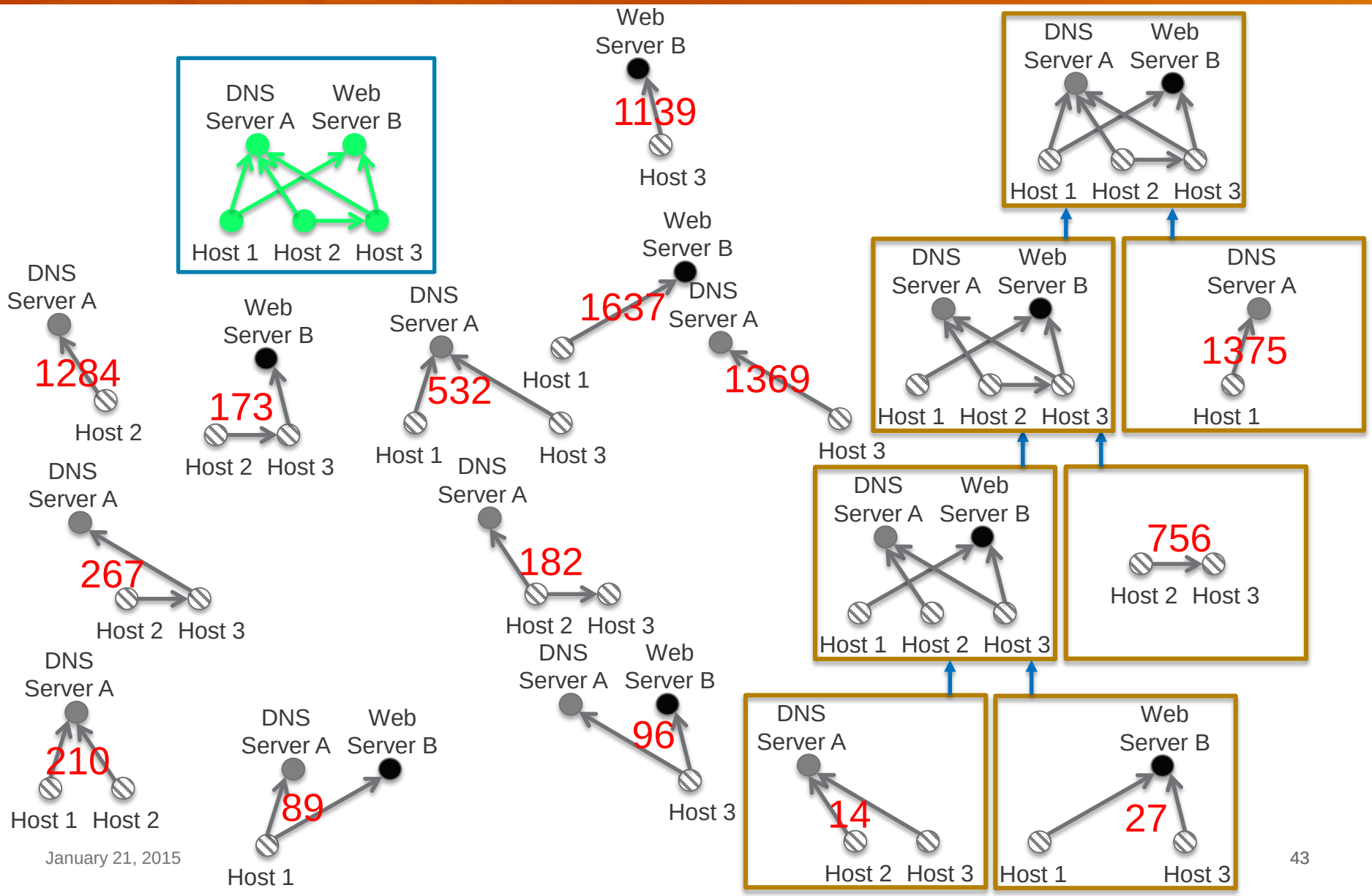
Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



Automatic Frequency-Based Join Tree Generation with Known Graph Pattern



Automatic Frequency-Based Join Tree Generation with Known Graph Pattern

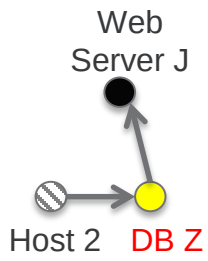


Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern

Emergent Infrequent
Subgraph Patterns

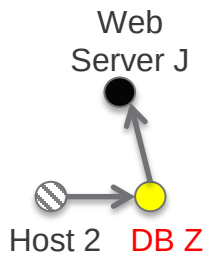
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern

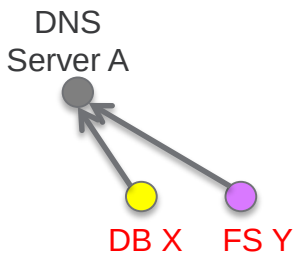
Emergent Infrequent Subgraph Patterns



Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern

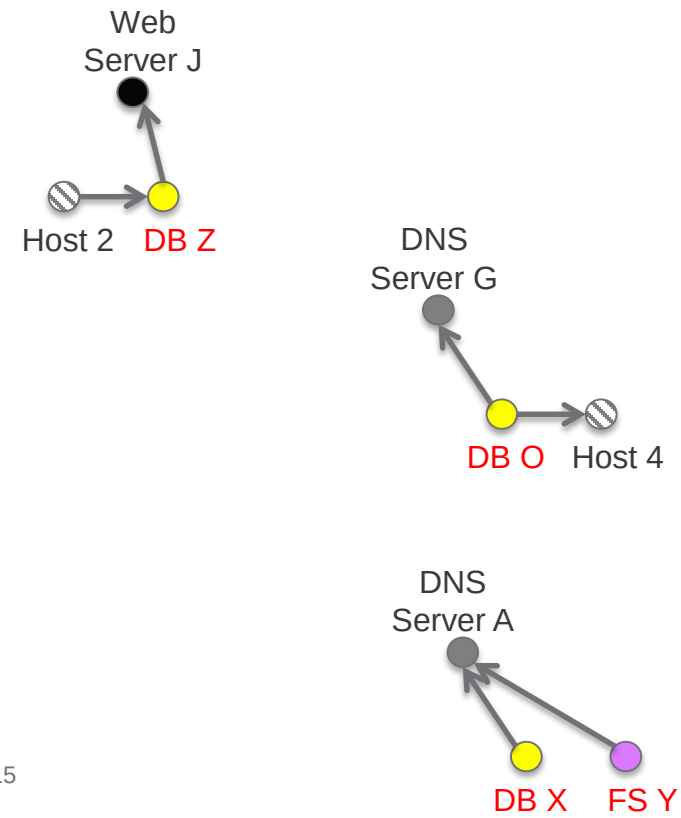
Emergent Infrequent Subgraph Patterns





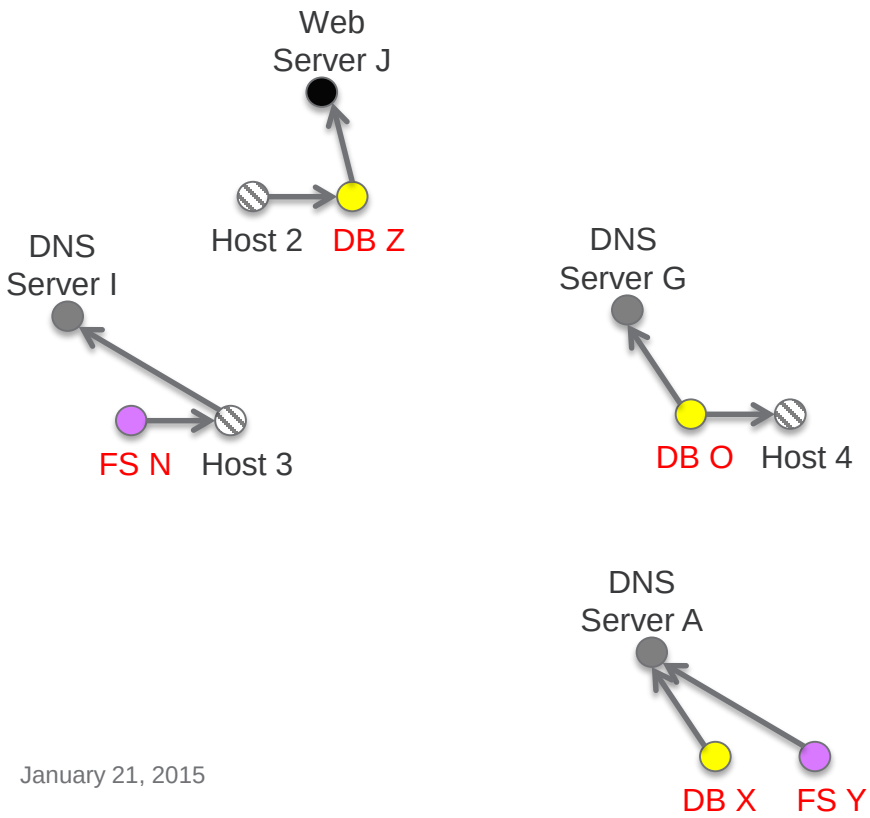
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern

Emergent Infrequent Subgraph Patterns

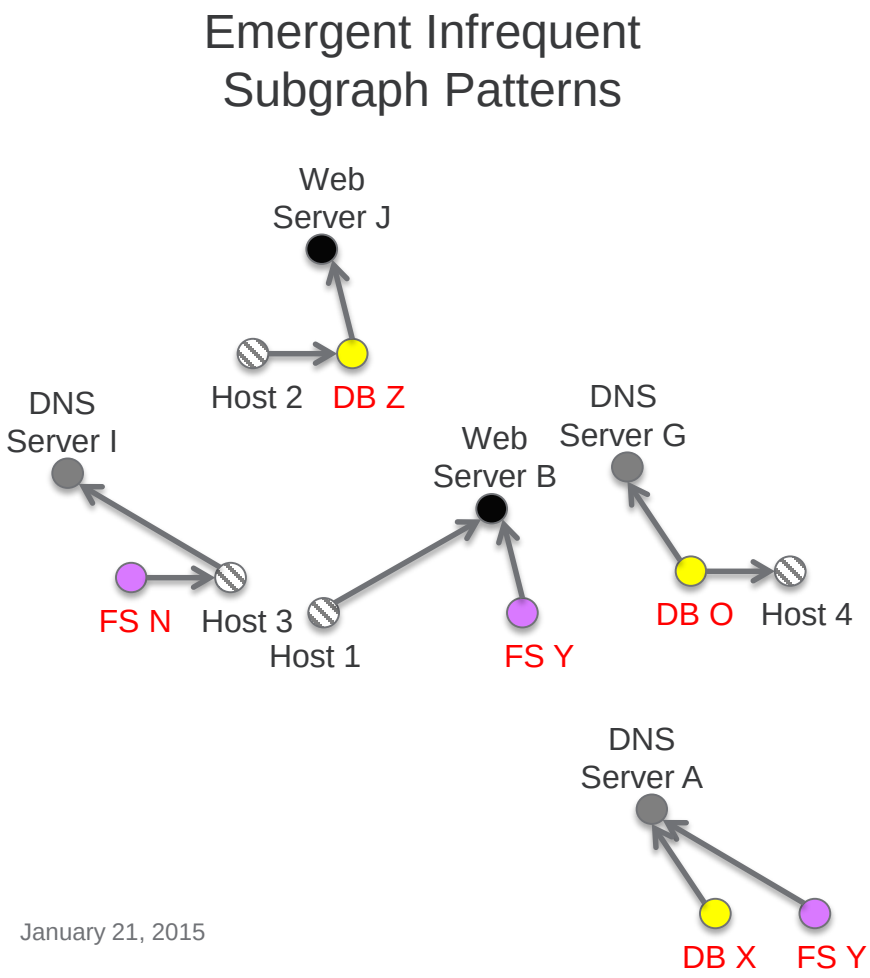


Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern

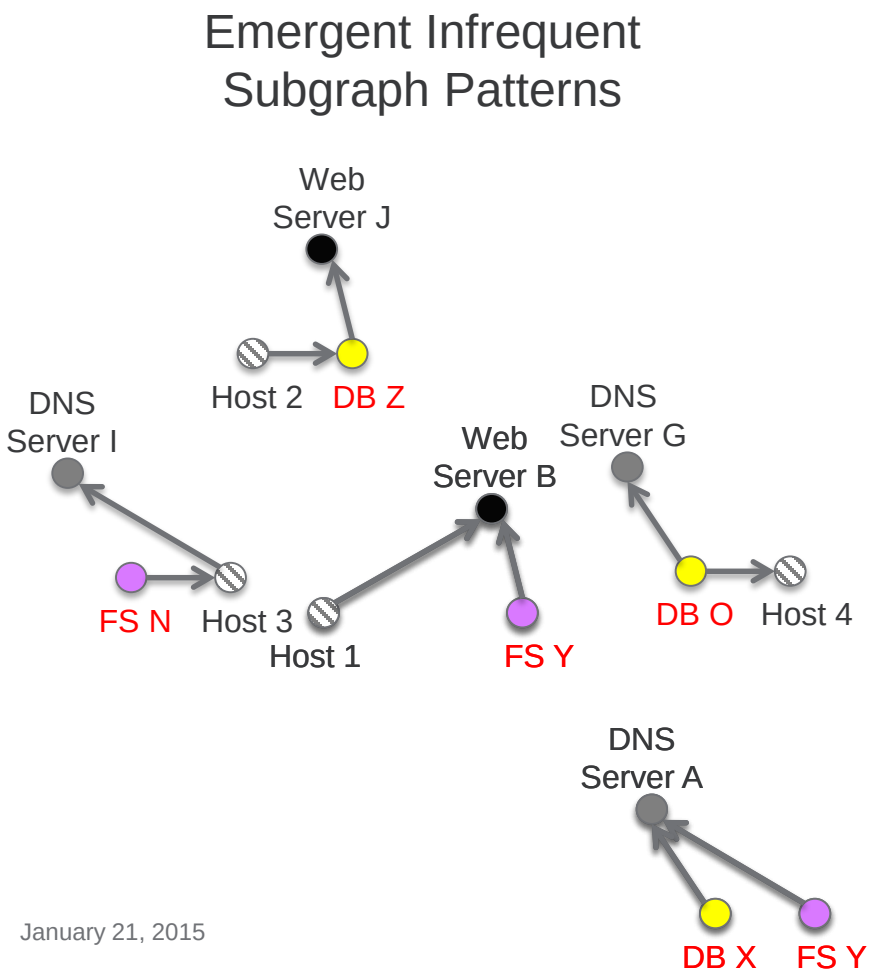
Emergent Infrequent Subgraph Patterns



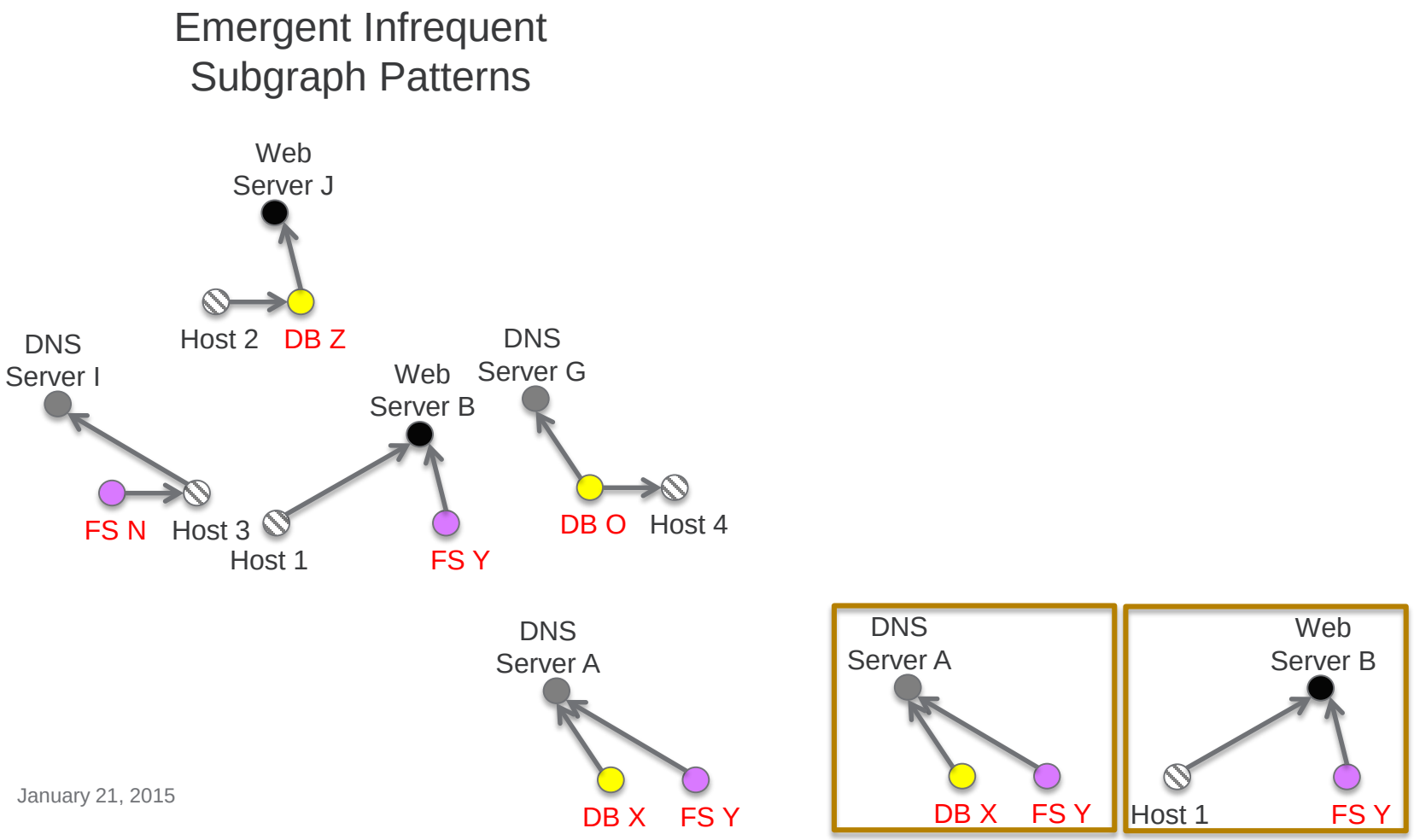
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



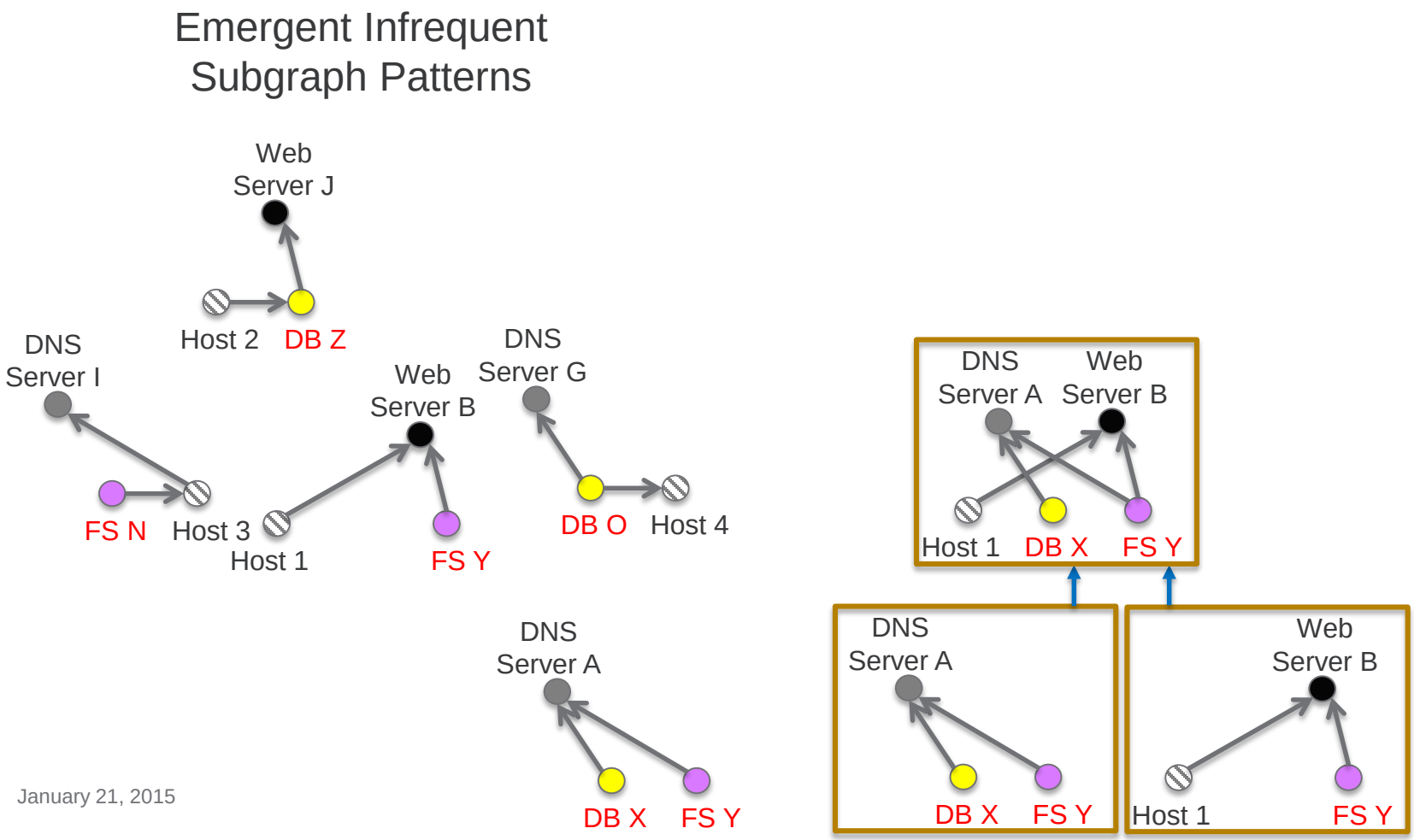
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



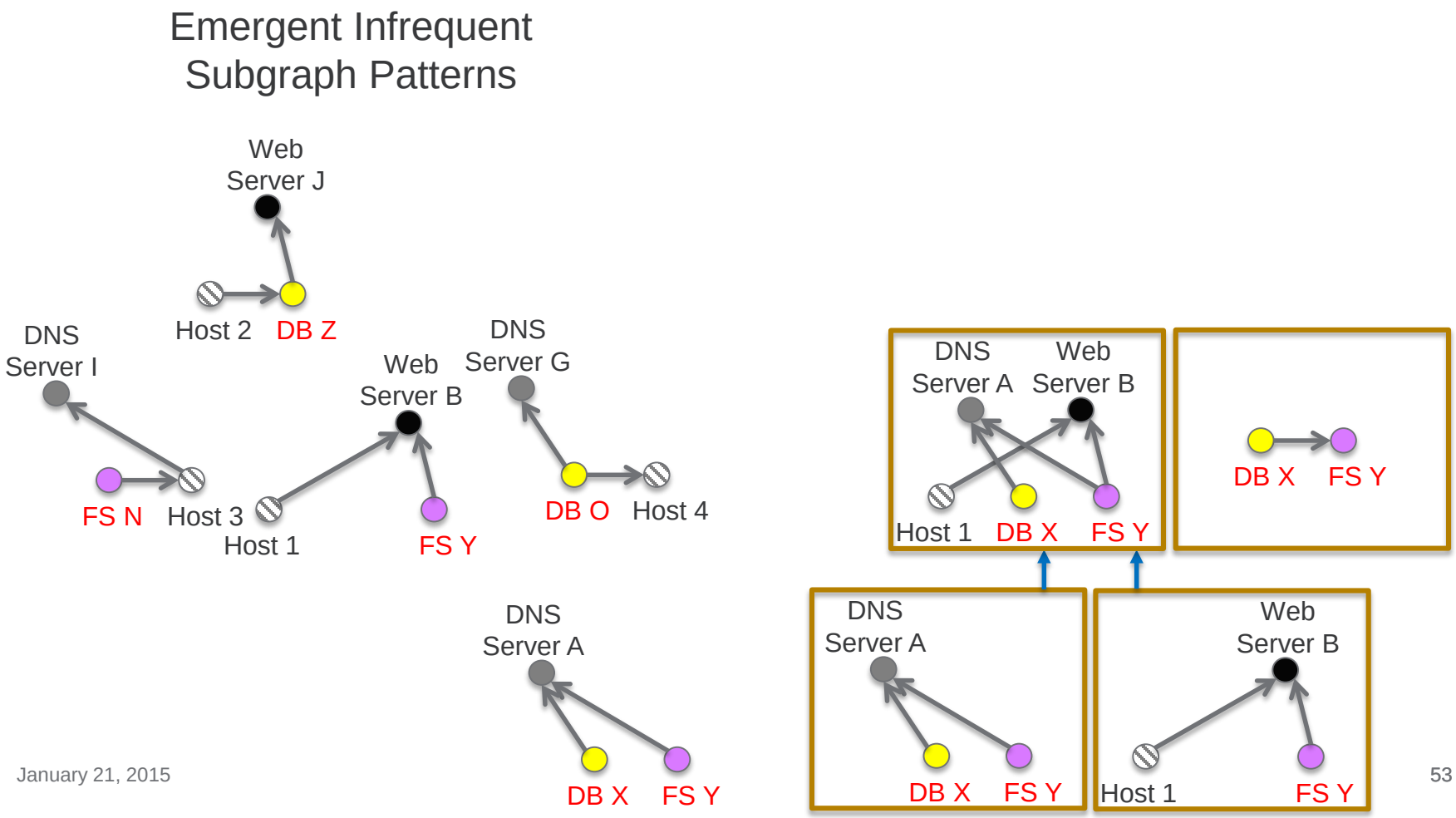
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



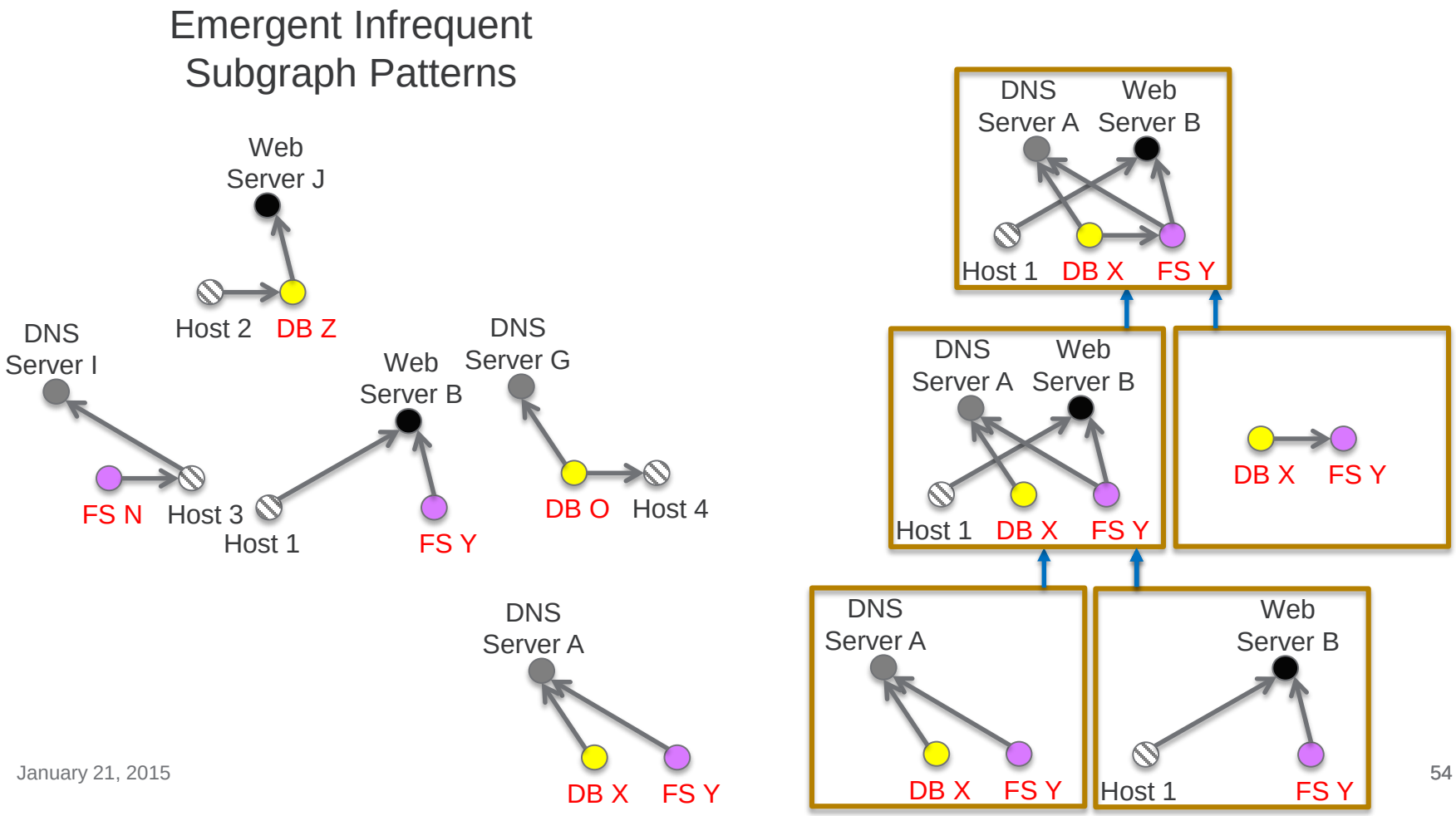
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



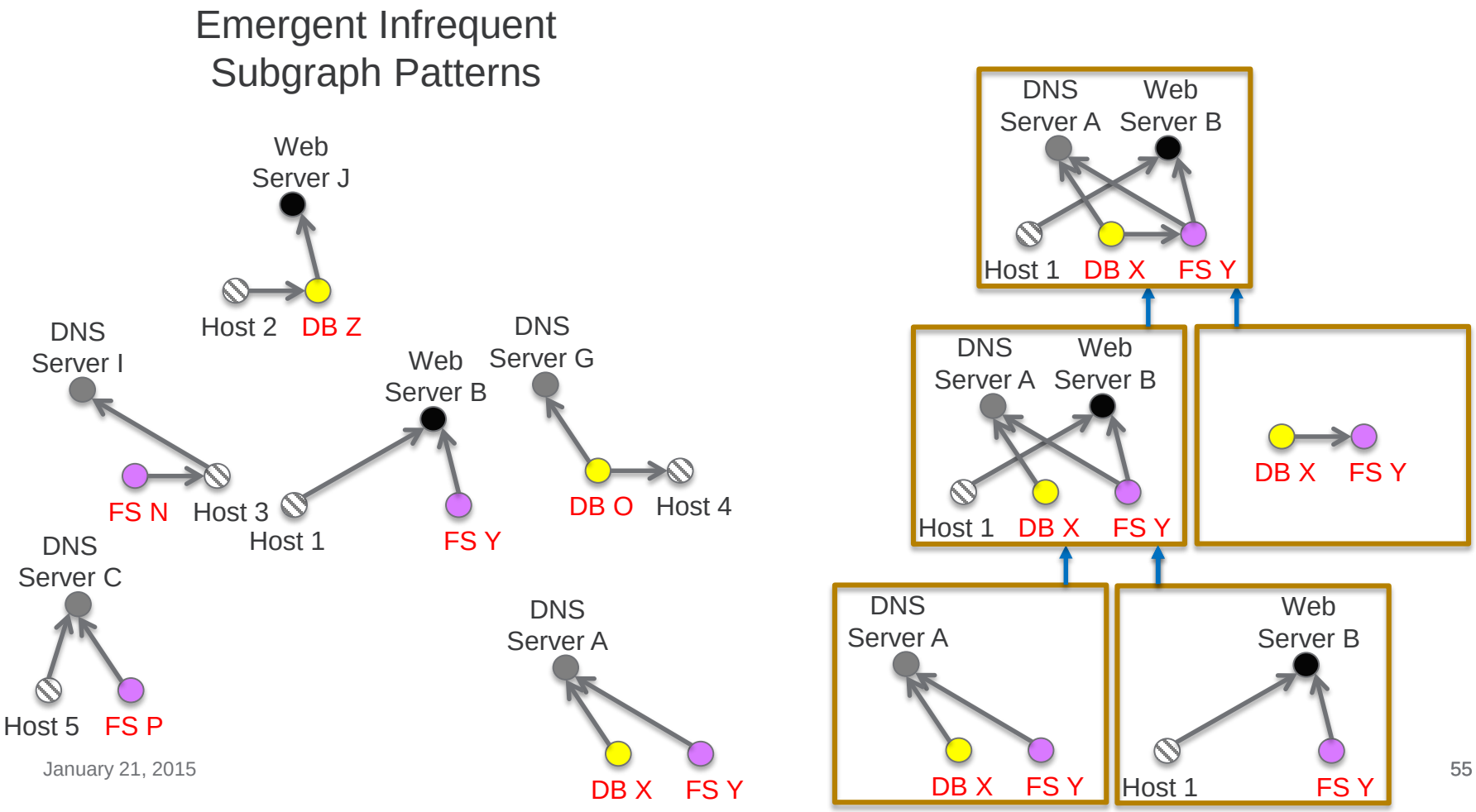
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



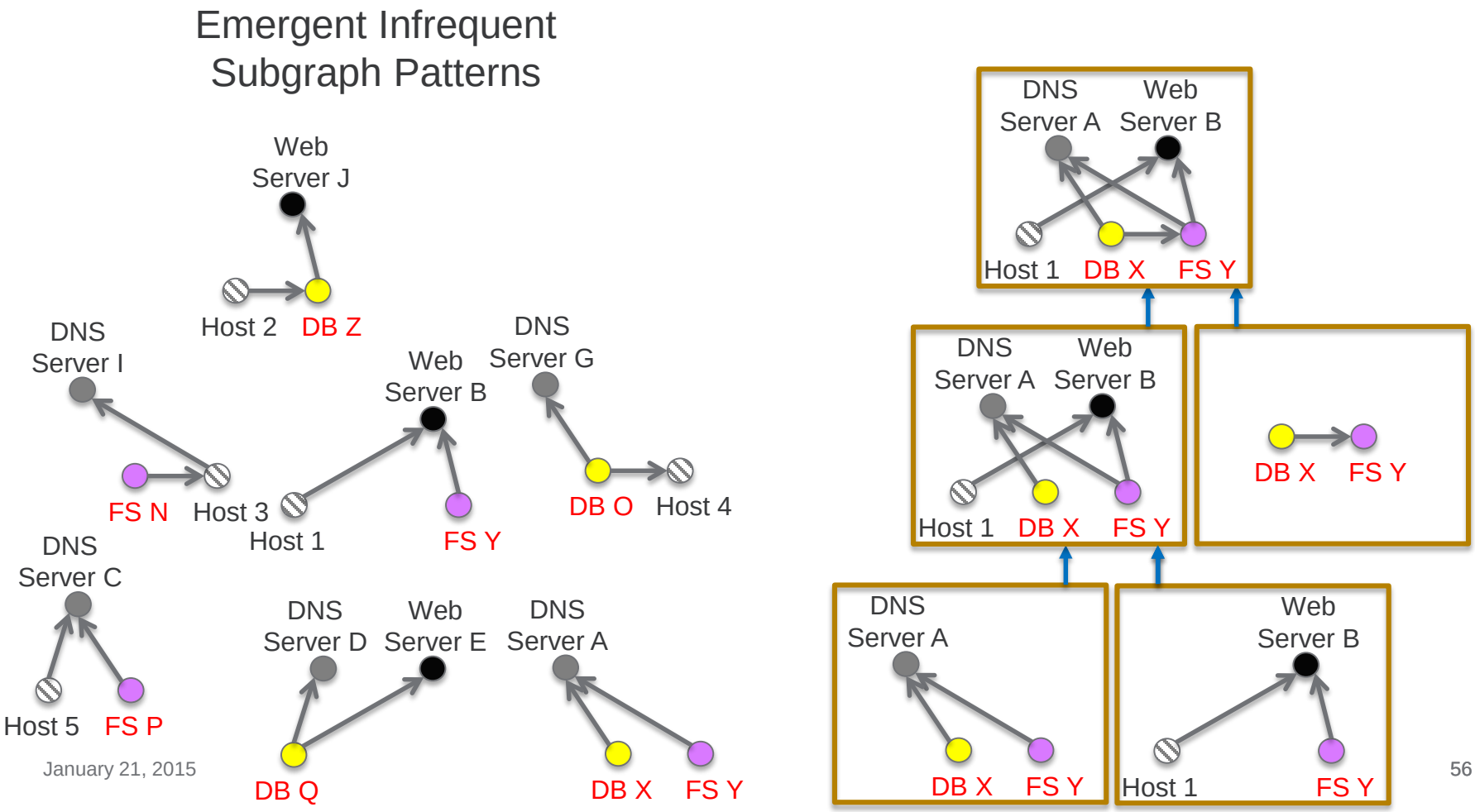
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



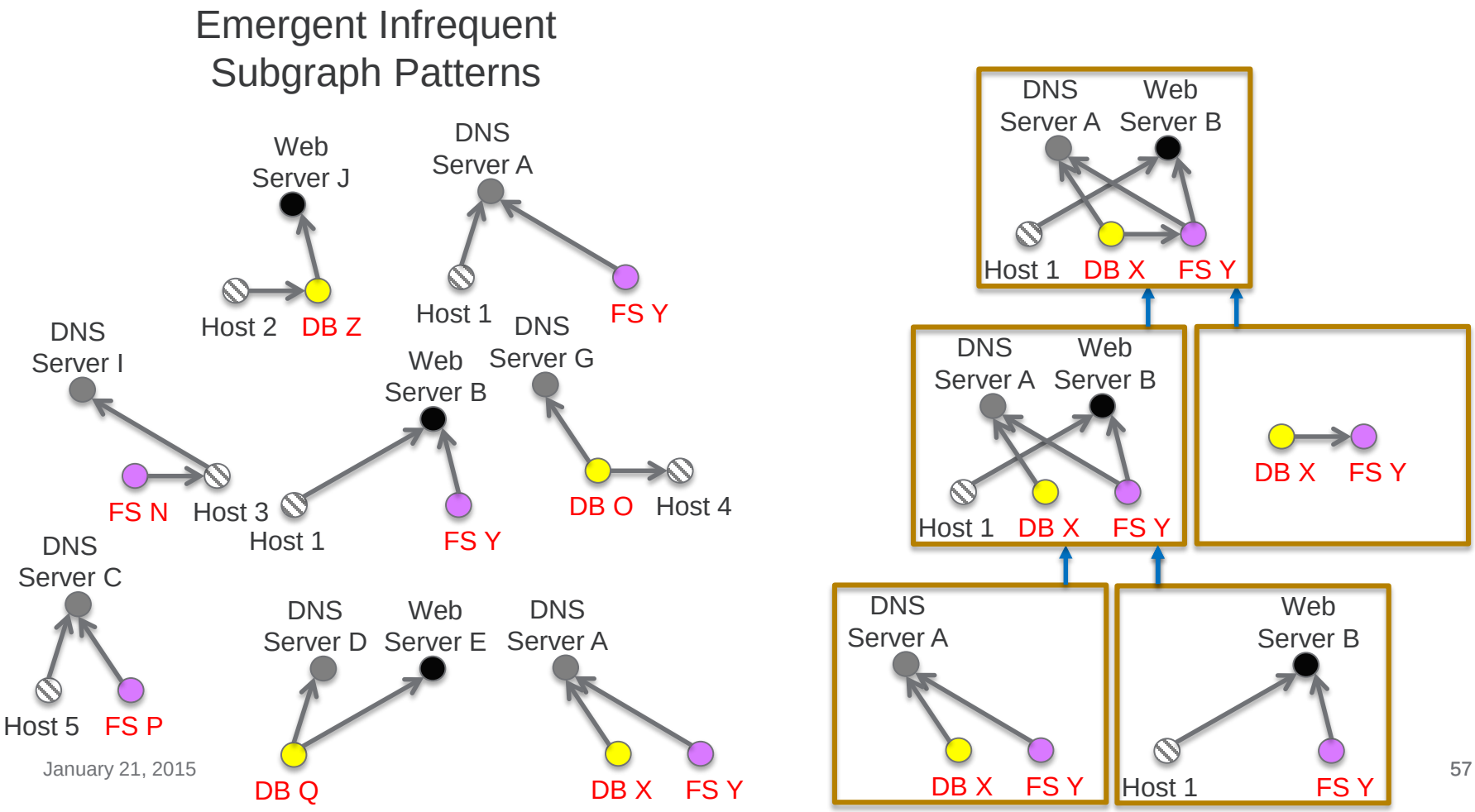
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



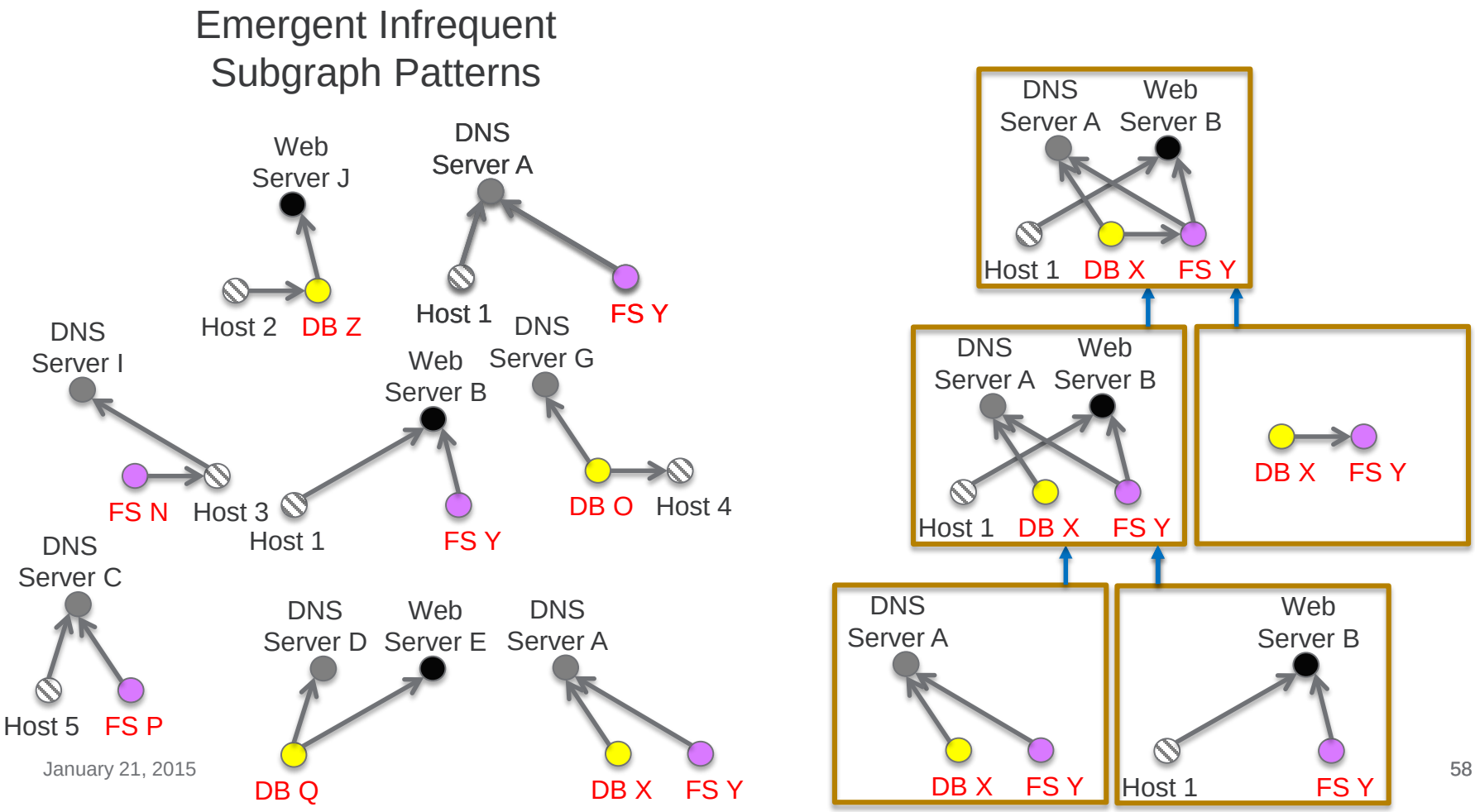
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



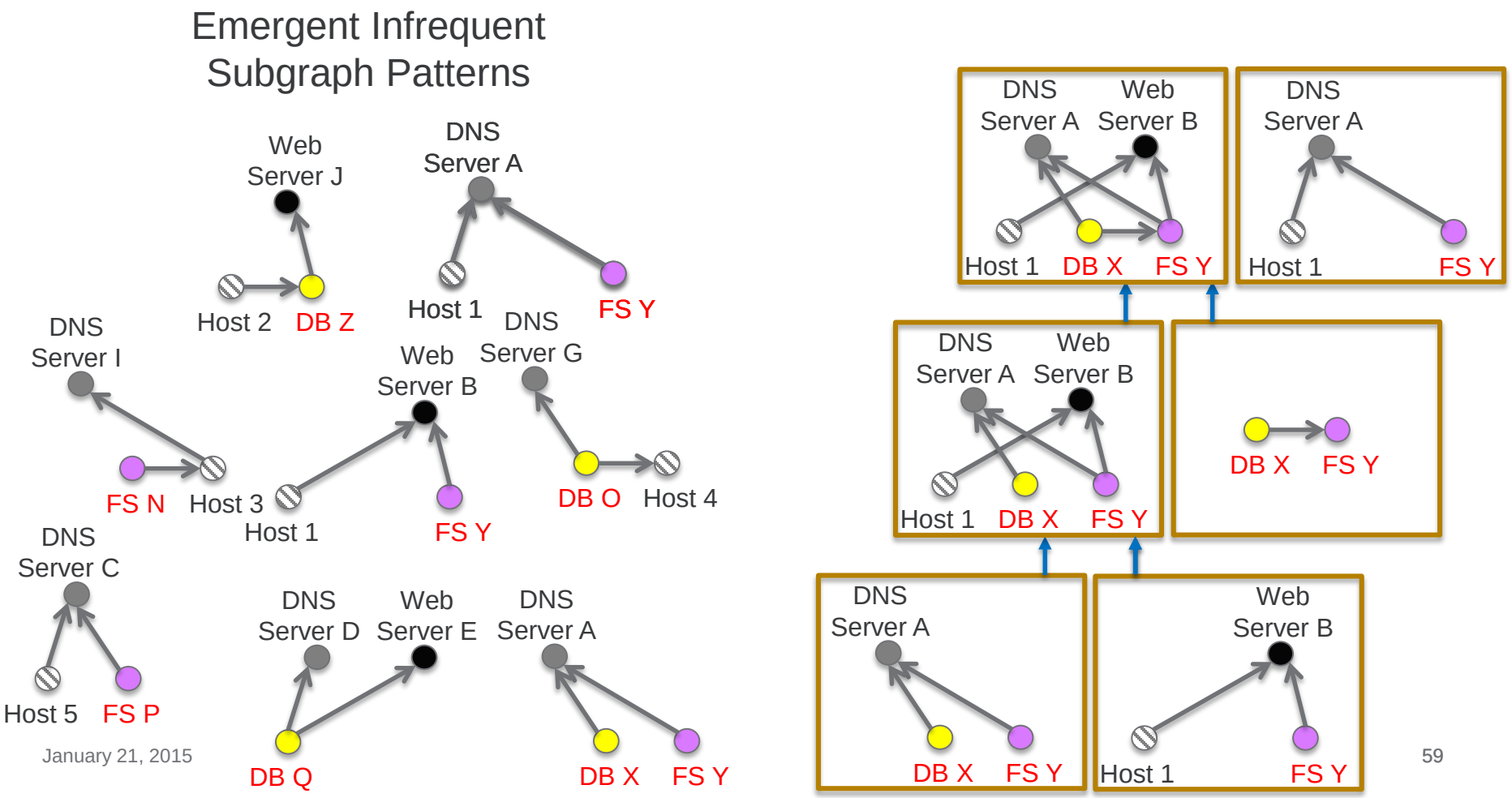
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



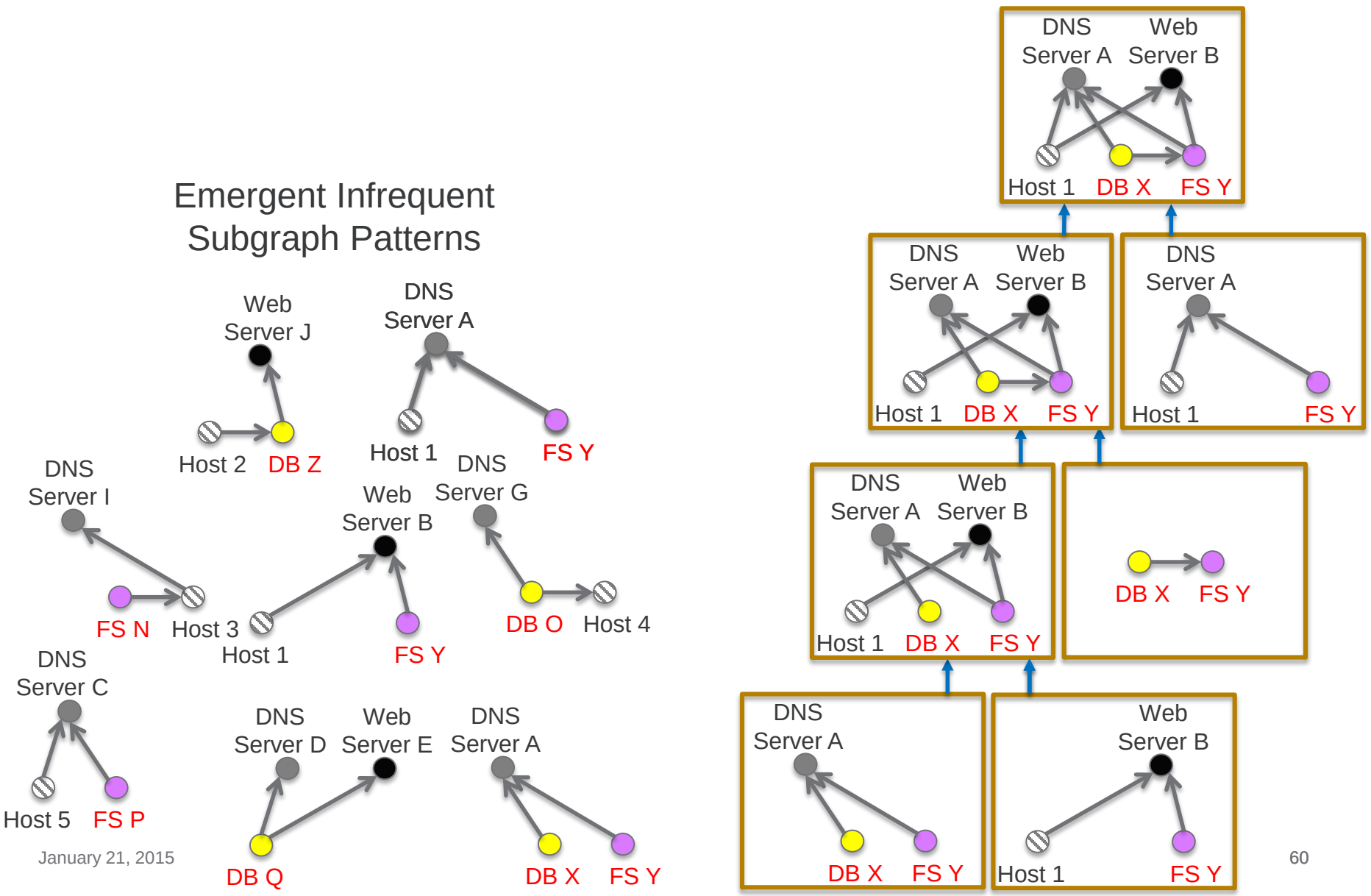
Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



Automatic Frequency-Based Join Tree Generation with Unknown Graph Pattern



Summary

- Developing scalable emerging subgraph pattern algorithm that can detect and identify precursor events and patterns as they emerge in complex networks
- Utilizing an efficient and novel subgraph join tree approach which tracks and monitors partial matches of a query graph against a large-scale dynamic network
- Applying emerging subgraph pattern algorithm to the detection of computer network threats and intrusions
- Packaging emerging subgraph pattern capabilities into an interactive network analysis framework called StreamWorks
- Extending StreamWorks to support emerging subgraph patterns across multiple dynamic data sources
- Developing approach for dynamic subgraph join tree generation to support the detection of zero-day exploits