Algorithms and Techniques for Scalable, Reliable Edge-to-Cloud Industrial Internet of Things

PhD Dissertation Defense
March 18, 2015

Kyoungho An
Institute for Software Integrated Systems (ISIS)
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, Tennessee

Technology Trends: Industrial Internet of Things

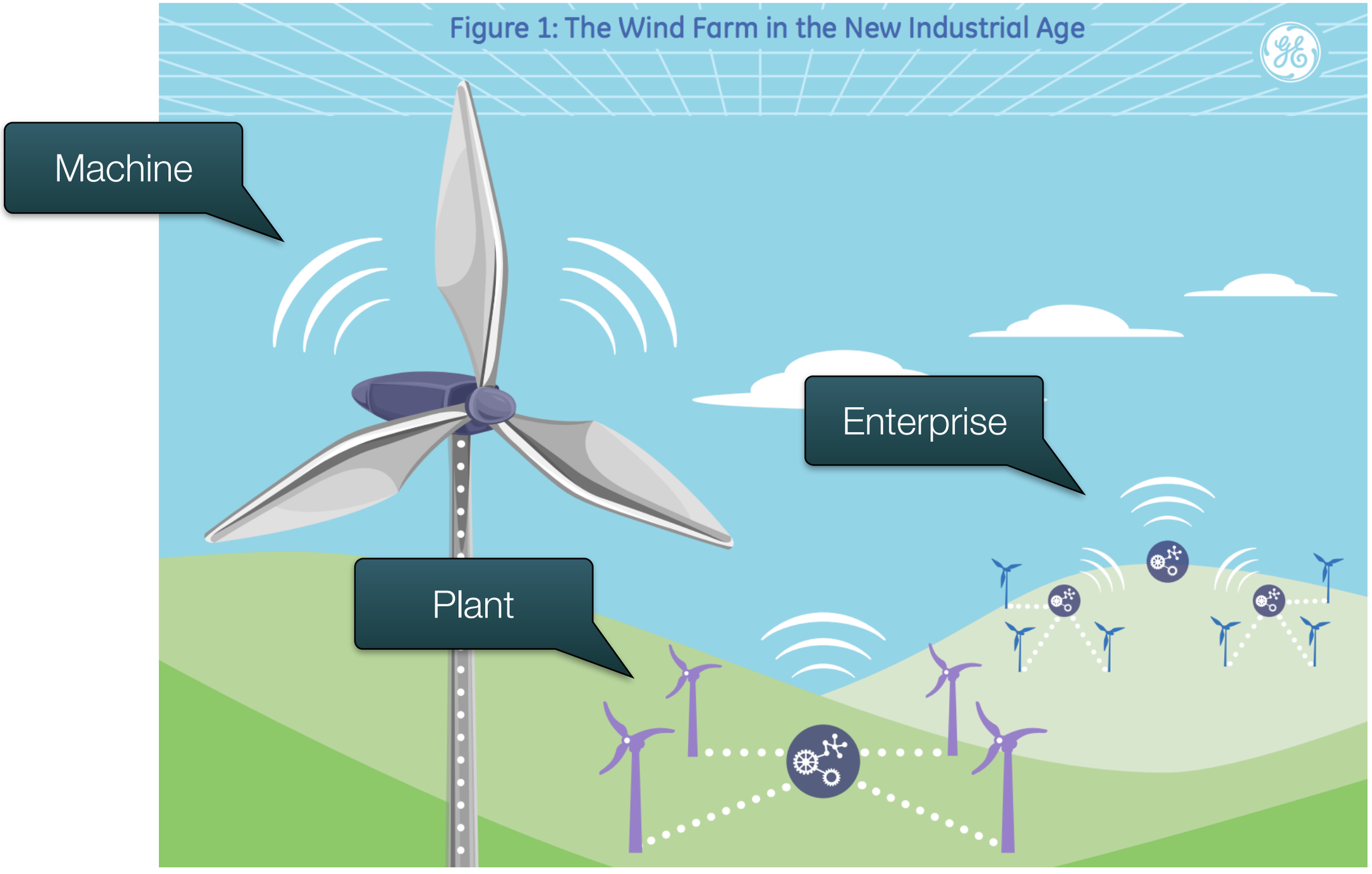
- Internet of Things (IoT) - Things hyper-connected over Internet realized by advances of networking, sensors, and embedded devices

- Collecting, sharing, analyzing data from connected things to provide intelligent and predictive services

- Industrial IoT - Industry oriented and mission-critical applications such as Healthcare, Transportation, Manufacturing, Energy

- Dissertation describes challenges and solutions about data sharing middleware and cloud infrastructures for Industrial IoT systems

3-Level Analysis of Industrial IoT



Reference from https://www.gesoftware.com/Industrial_Big_Data_Platform.pdf

3-Level Analysis of Industrial IoT

	Turbine (Machine)	Wind farm (Plant)	Power producer (Enterprise)
Analytics	Asset optimization	Operations optimization	Business optimization
Data Quantity	>100 tags	>6,000	>1,000,000 tags
Data Frequency	40 milliseconds	1 second	1 second - 10 minutes

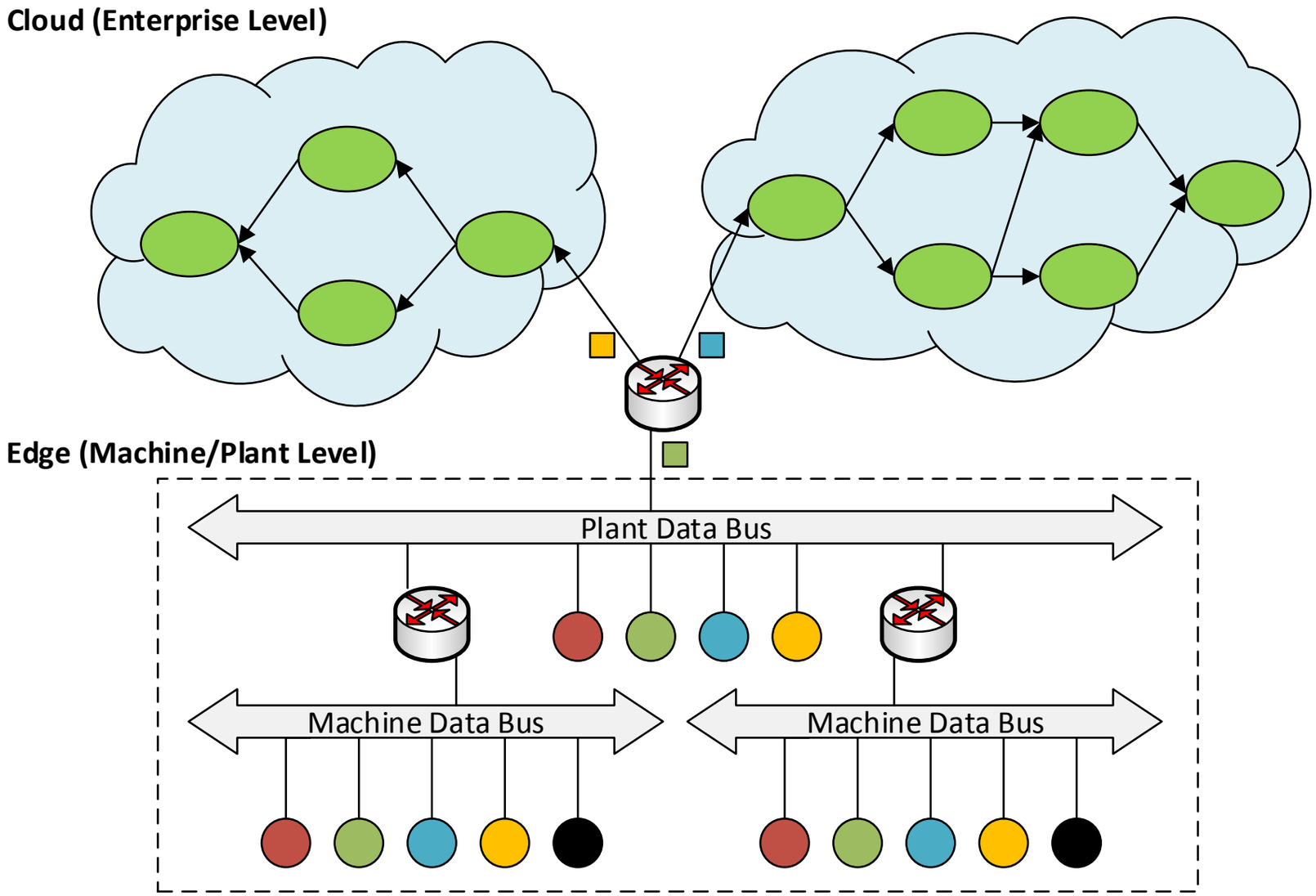
Reference from https://www.gesoftware.com/Industrial_Big_Data_Platform.pdf

- Each level requires different non-functional requirements
- All levels demand some degrees of timeliness requirements

Motivational Architecture

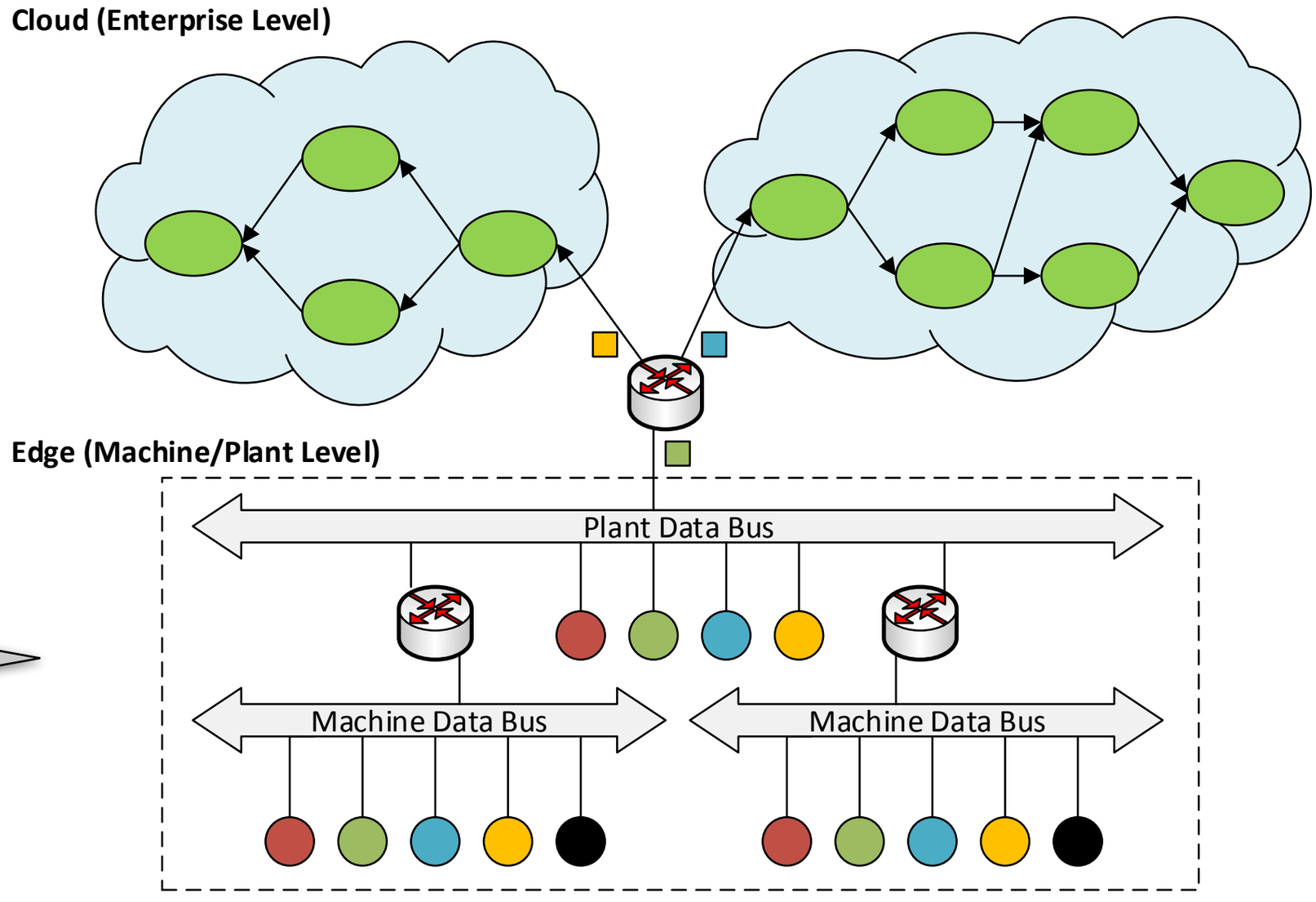
Data processing to optimize operations at edge

Data collection and sharing from edge to edge or to cloud

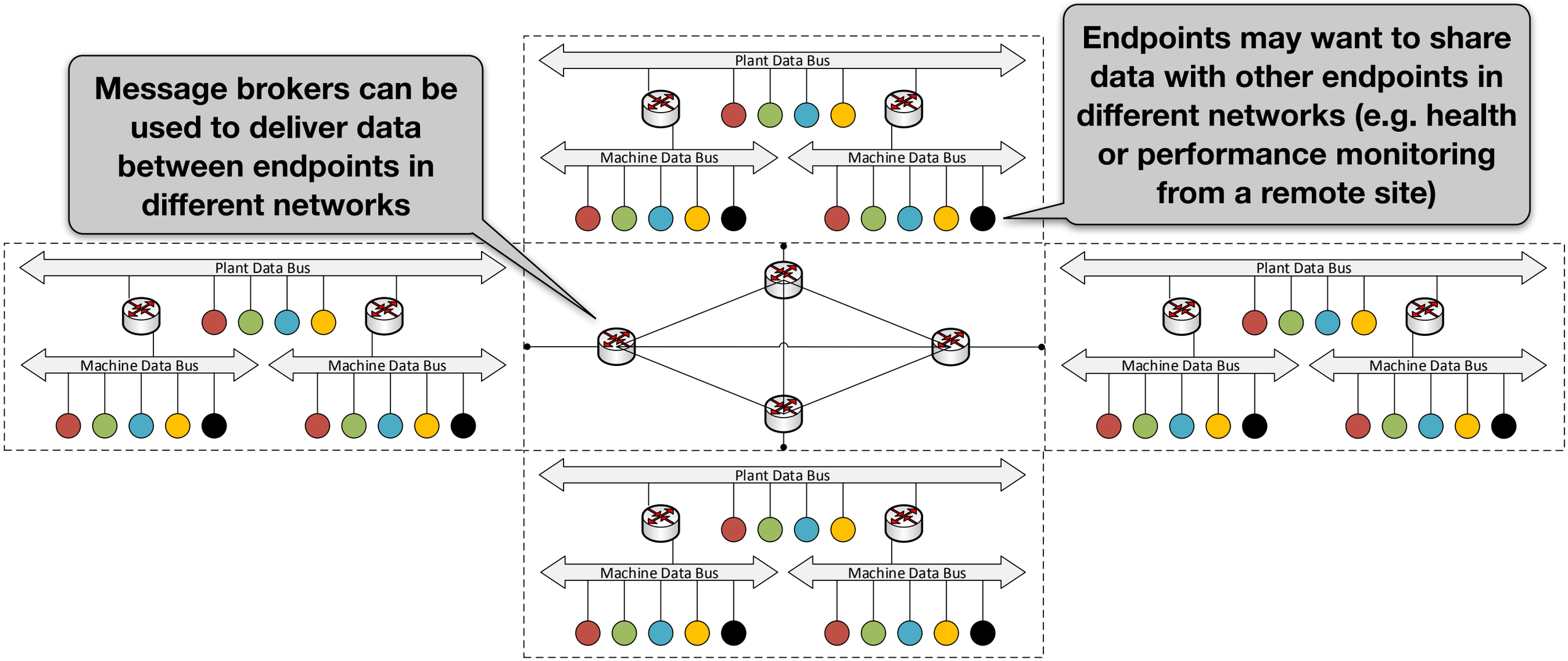


Challenge 1:
Scalability of Discovering Devices and Endpoints at Edge

- A turbine contains more than 50 sensors
- A wind farm contains more than 200 turbines
- At least 10,000 endpoints exist



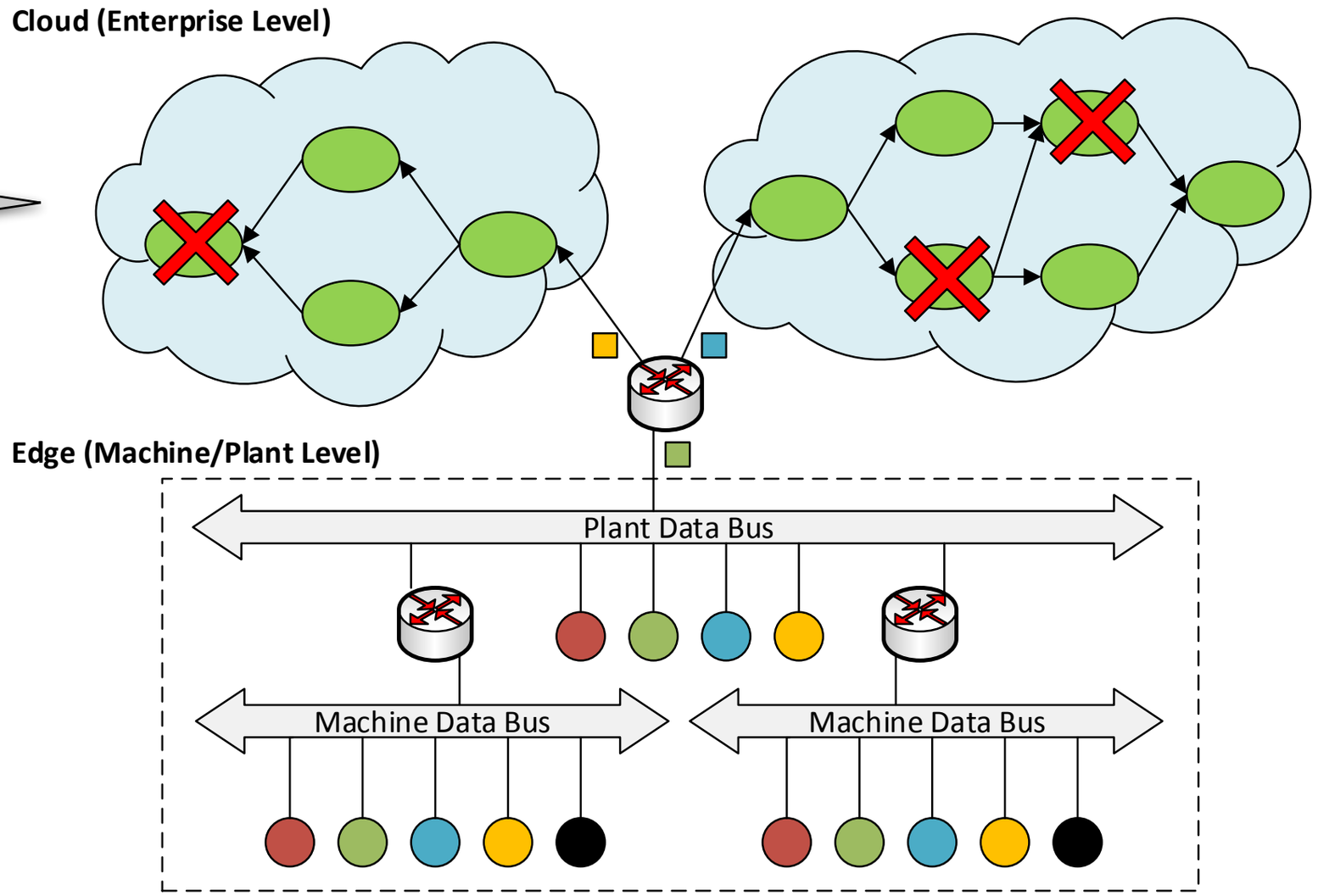
Challenge 2:
Overlay Networks for Data Distribution over WANs



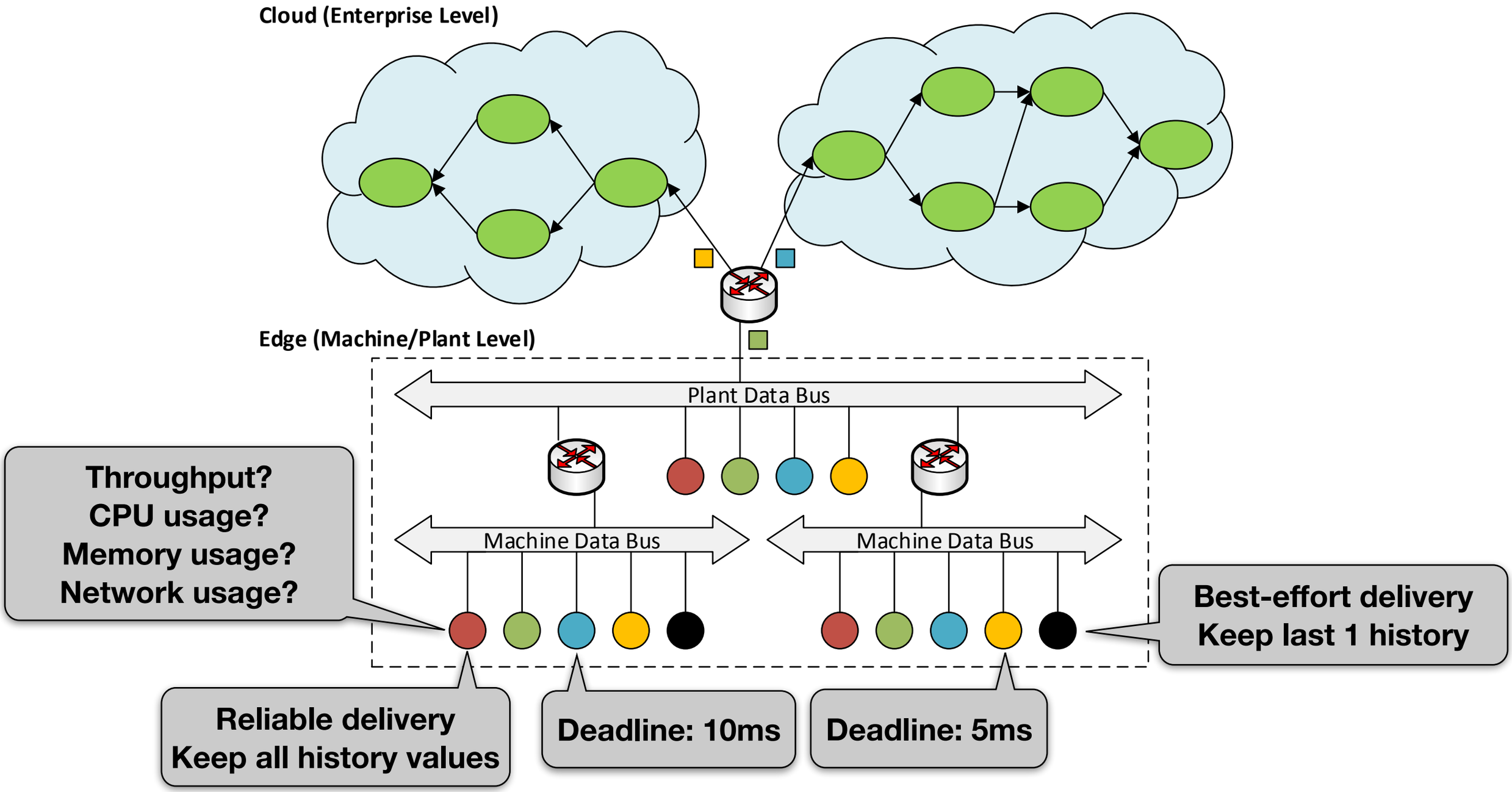
- How to automatically discover brokers over WANs? How to form an optimal overlay network in terms of scalability and low latency? How to guarantee consistency of dissemination paths for dynamic endpoints?

Challenge 3:
High Availability and Timeliness at Enterprise-level

- **Faults can happen by failures of physical machines or virtual machines (VM)s in the cloud**
- **Failover using periodic snapshots of VMs**
- **How to guarantee the same service level even after failover? Optimal placement of backup VMs?**



Challenge 4:
Testing Performance with Different QoS settings?



- Requires a technique to validate performance impact by different combinations of QoS settings

Dissertation Focus Areas

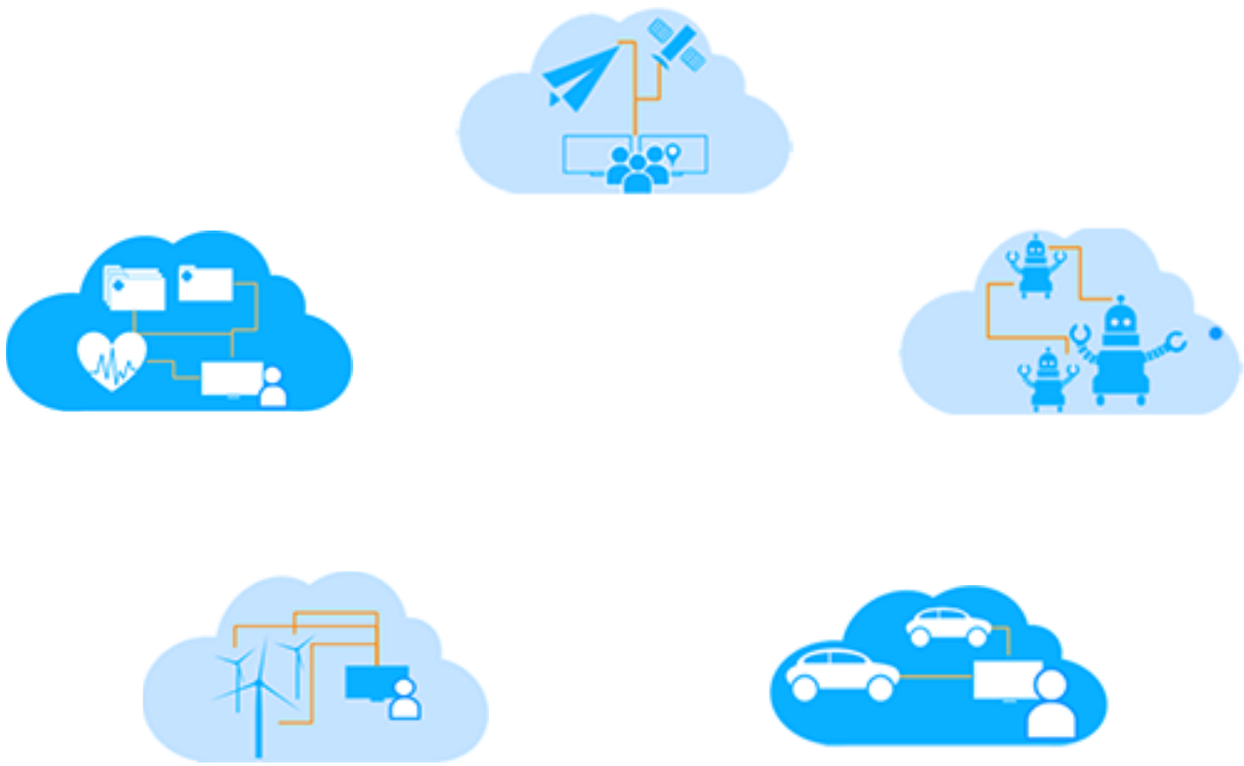
	Turbine (Machine)	Wind farm (Plant)	Power producer (Enterprise)
Focus Area 1		Scalable discovery protocols for pub/sub middleware	
Focus Area 2			Coordination service for pub/sub middleware in WANs
Focus Area 3			Fault-tolerant and real-time cloud infrastructures
Focus Area 4		Performance testing with different combinations of QoS settings	

Focus Area 1:
Scalable discovery protocols for pub/sub middleware

	Turbine (Machine)	Wind farm (Plant)	Power producer (Enterprise)
Focus Area 1		Scalable discovery protocols for pub/sub middleware	
Focus Area 2		Coordination service for pub/sub middleware in WANs	
Focus Area 3			Fault-tolerant and real-time cloud infrastructures
Focus Area 4		Performance testing with different combinations of QoS settings	

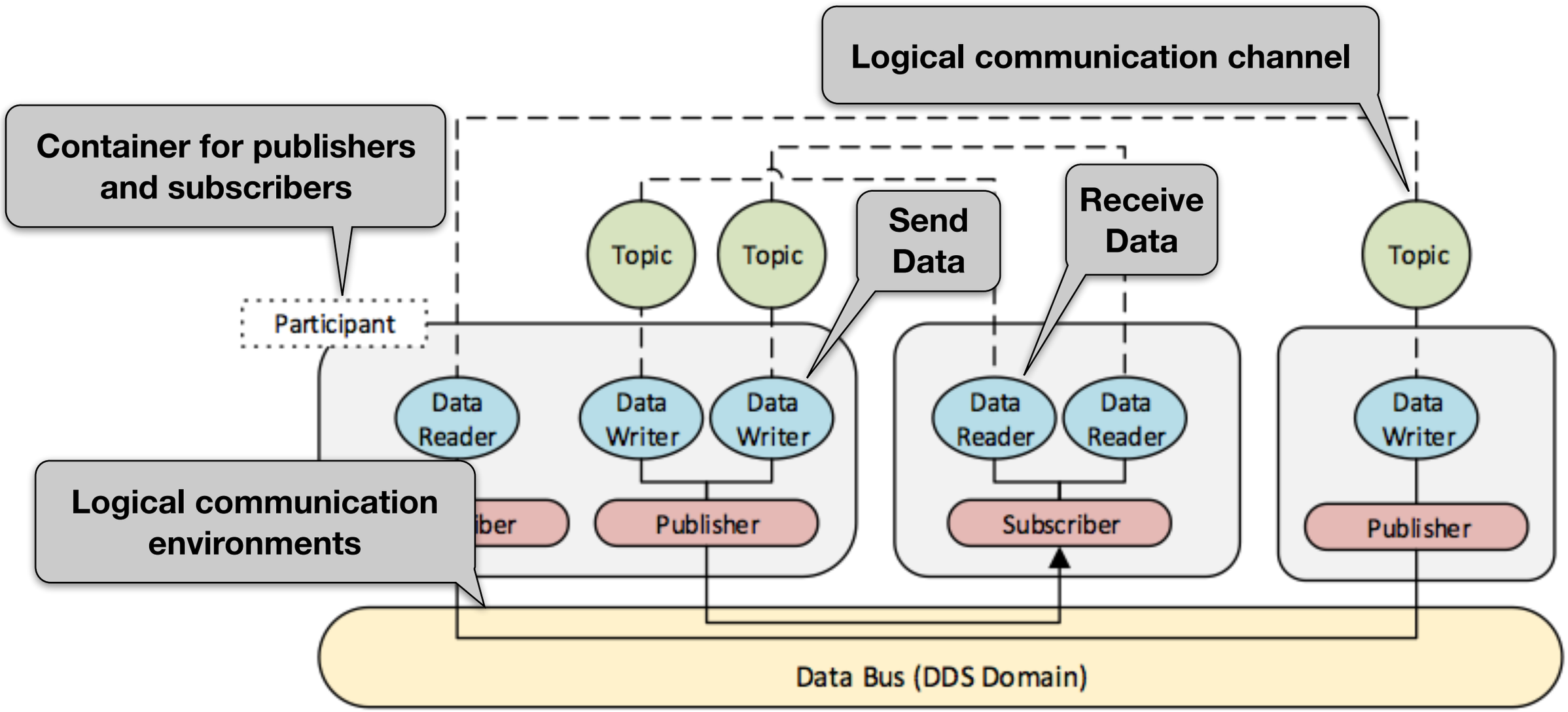
Context

- Data-centric pub/sub middleware can be used to share data between devices in Industrial IoT systems
  - Event-based communications
  - Data-centric addressing
  - Decoupling communication endpoints
  - Many-to-many communications
  - QoS and content-based filtering supports

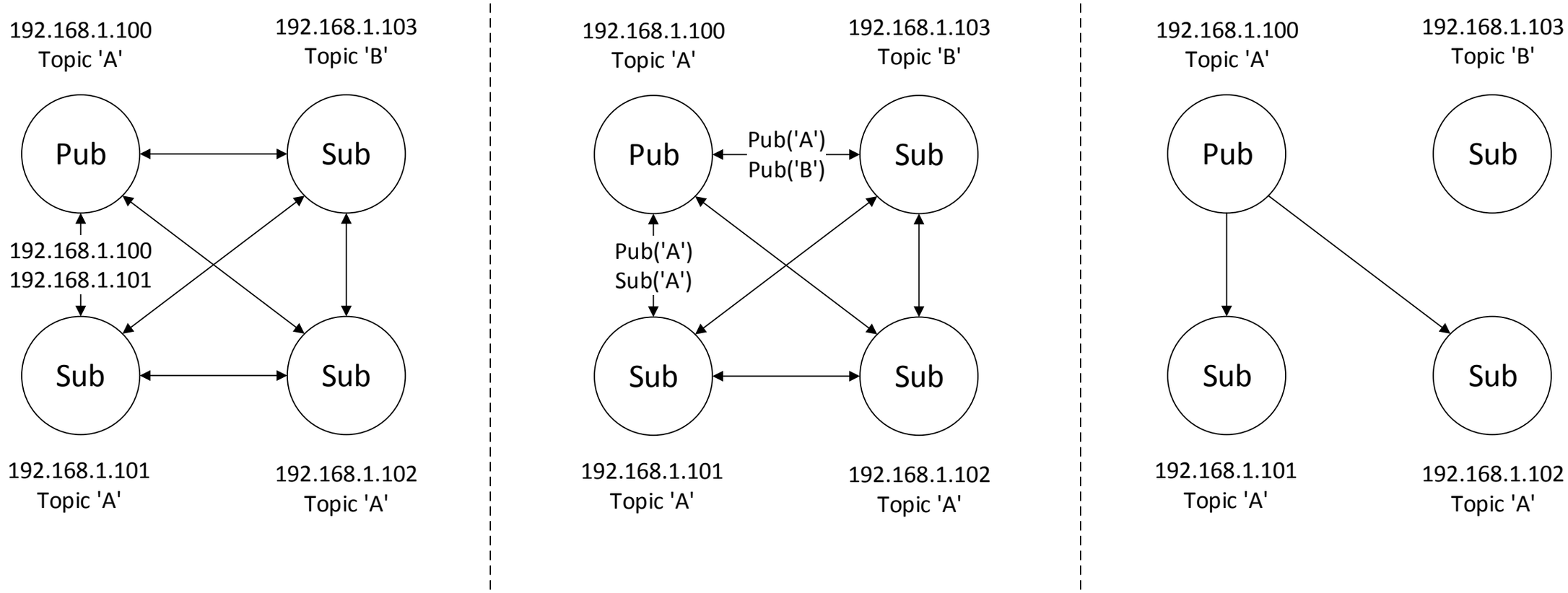


Data Distribution Service (DDS)

- Data Distribution Service (DDS) is an OMG standard specification for data-centric publish/subscribe middleware



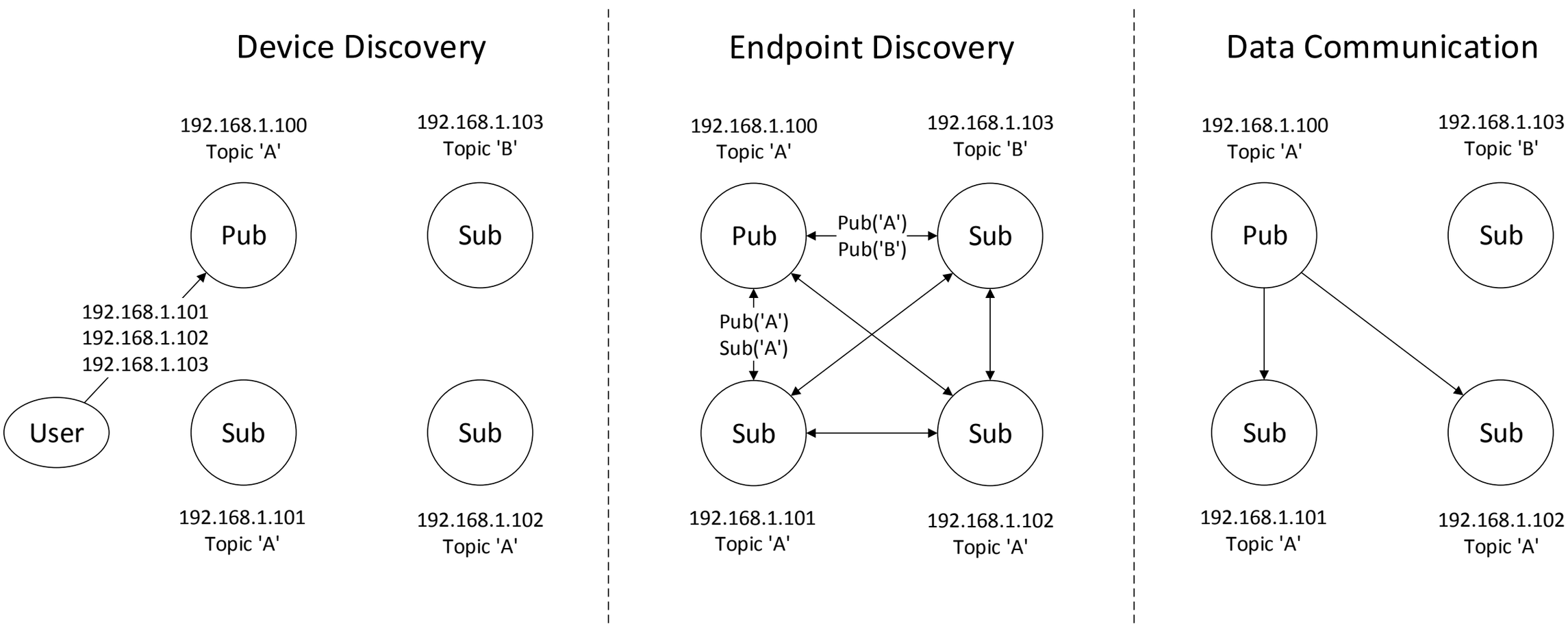
Pub/Sub Discovery Mechanisms: Peer-to-Peer Discovery with Multicast



- **Automatic discovery without external infrastructures (zero infrastructure)**

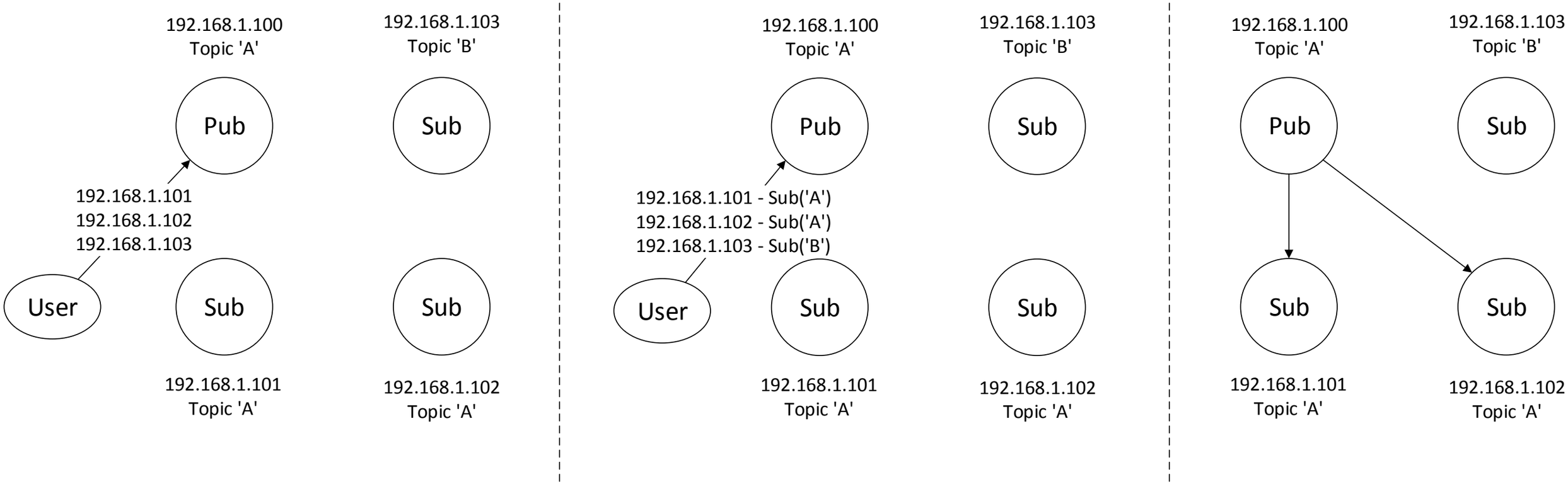
- **DDS Simple Discovery Protocol (SDP), AllJoyn Data-Driven API, and ZeroMQ zbeacon**

Pub/Sub Discovery Mechanisms: Peer-to-Peer Discovery with Unicast



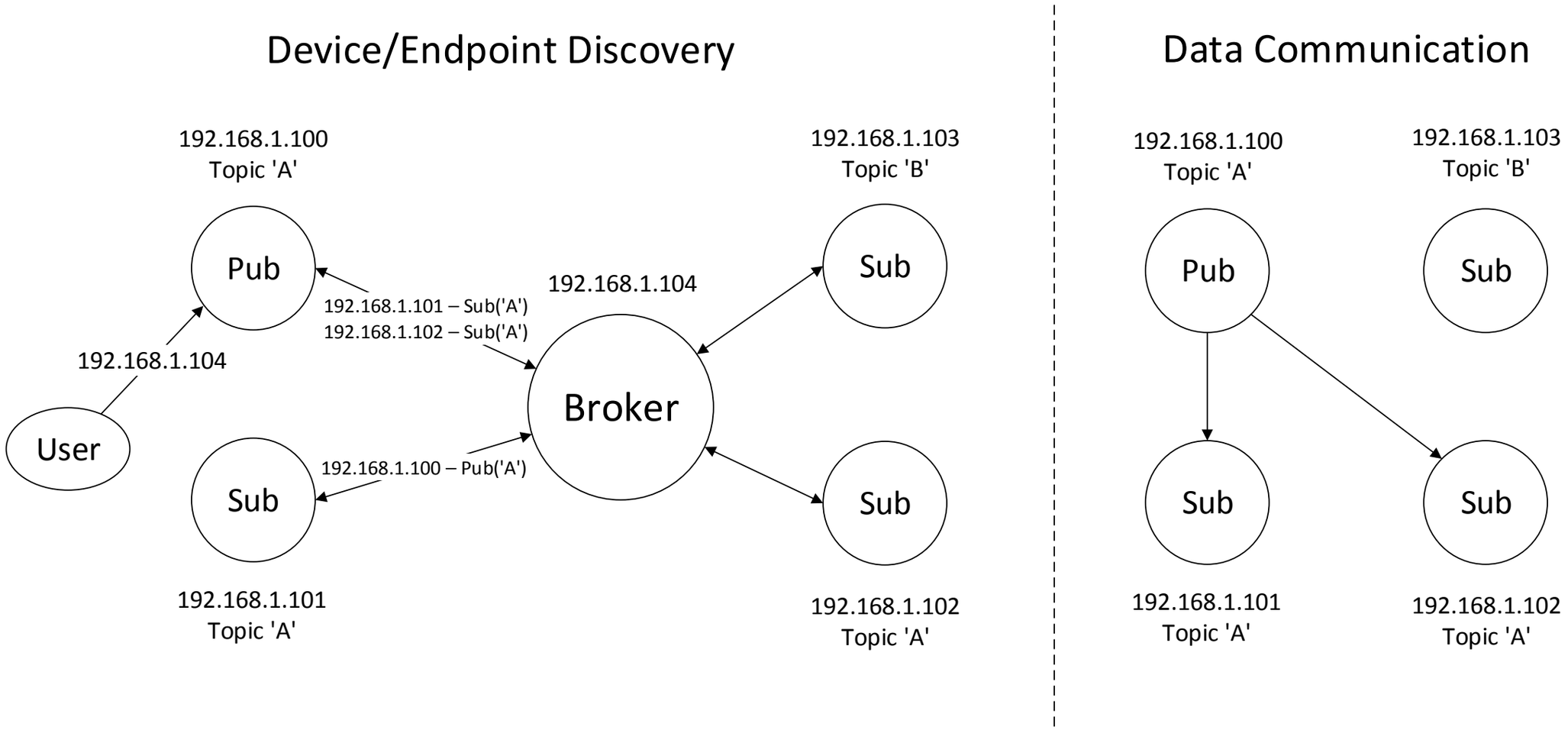
- **Manual configurations required at the device discovery**

Pub/Sub Discovery Mechanisms: Static Discovery



- Useful for resource constrained environments, but requires manual configuration efforts
- ZeroMQ and DDS Low-Bandwidth Discovery

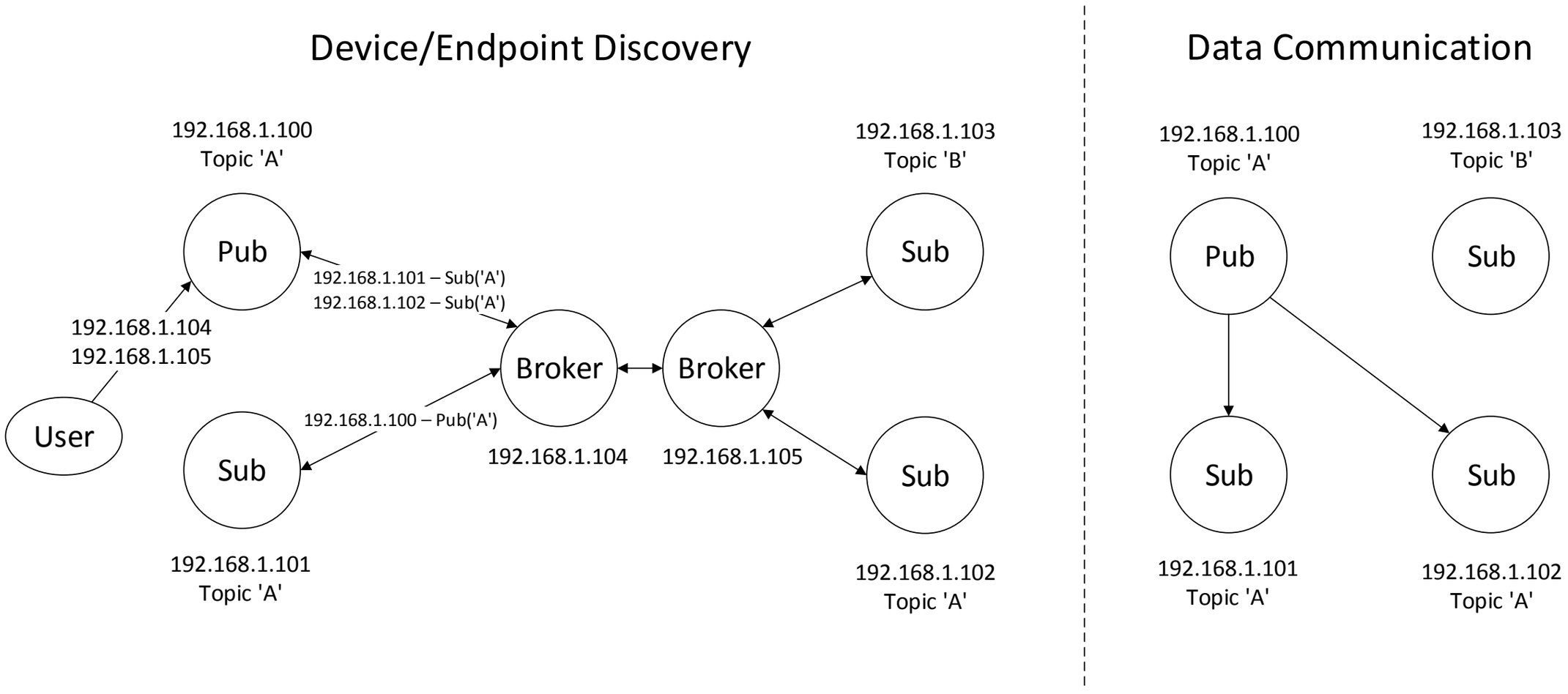
Pub/Sub Discovery Mechanisms: Centralized Discovery



- No redundant discovery information and requires minimal configuration efforts, but single-point of failure and bottleneck

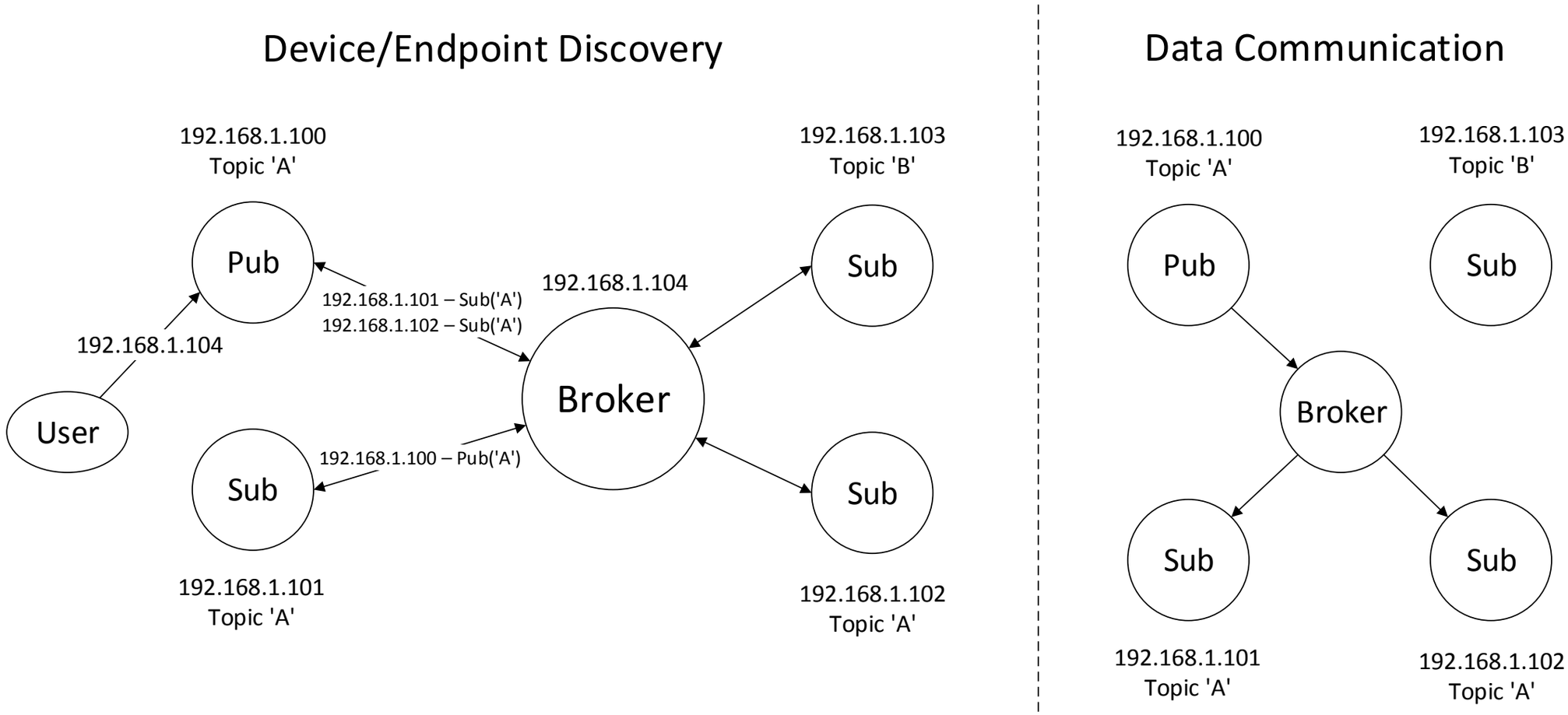
- DDS Enterprise Discovery Protocol (EDP)

Pub/Sub Discovery Mechanisms: Federated Discovery



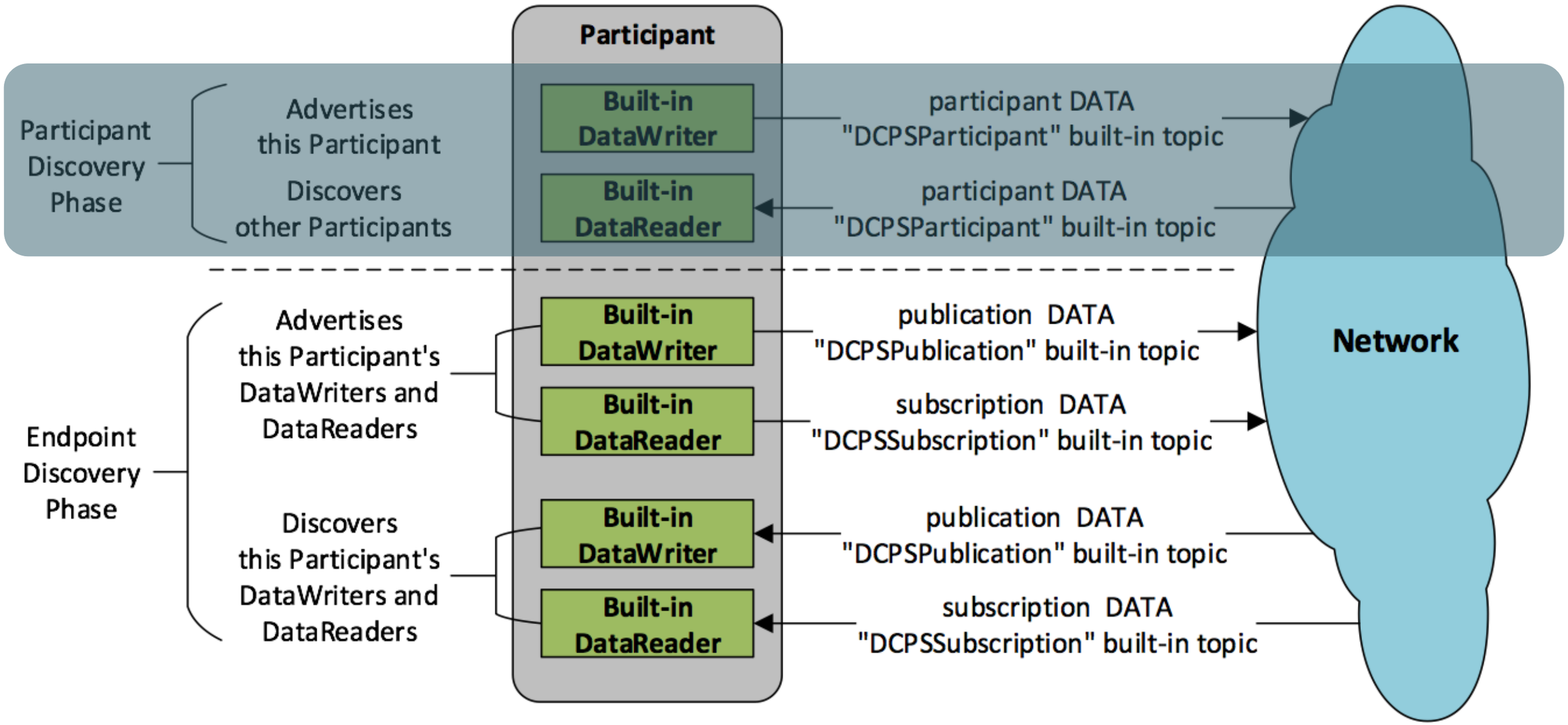
- **Avoid single point of failure and bottleneck**
- **Complicated implementations to synchronize states between brokers**

Pub/Sub Discovery Mechanisms: Brokered Discovery and Communications



- Uses a broker or federated brokers for discovery as well as communications
- Same issues as the centralized discovery and incurs higher latency compared to the peer-to-peer way
- XMPP, AMQP, and MQTT

DDS Simple Discovery Protocol (SDP) Entities



Challenges

- SDP scales poorly as the number of peers and their endpoints increases in a domain

- Why?

  - Each participant exchanges discovery messages with all participants in the same domain regardless of topics or endpoint types

- For a large scale system, substantial network, memory, and computing resources are consumed just for the discovery process

- This overhead degrades discovery completion time and hence overall scalability

Solution Approach:
Content-based Filtering Discovery Protocol (CFDP)

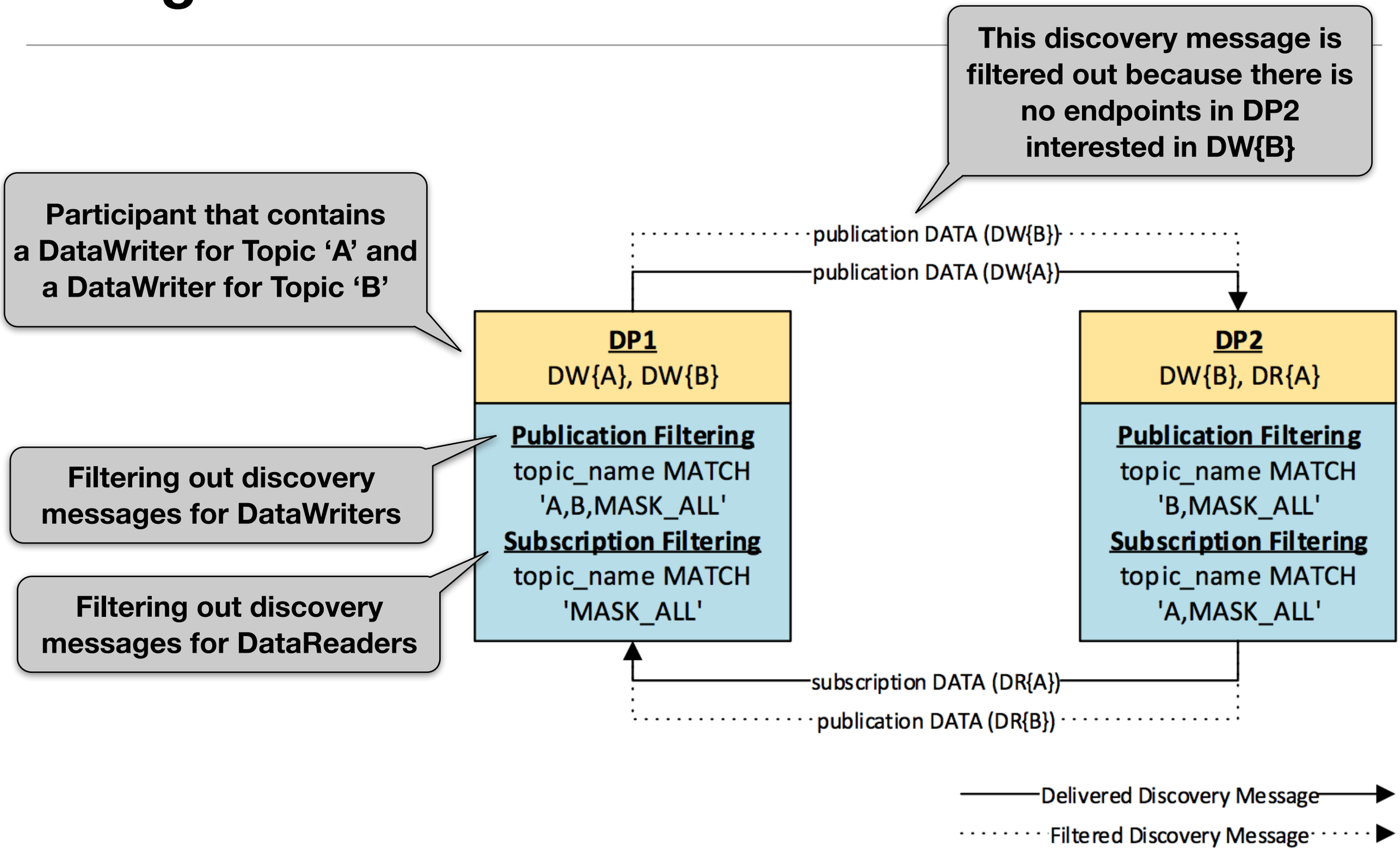
- CFDP filters out discovery messages based on topic names and endpoint types (e.g. DW or DR)

- Utilizes the participant discovery phase of SDP called SPDP

- Differs from SDP at the endpoint discovery phase

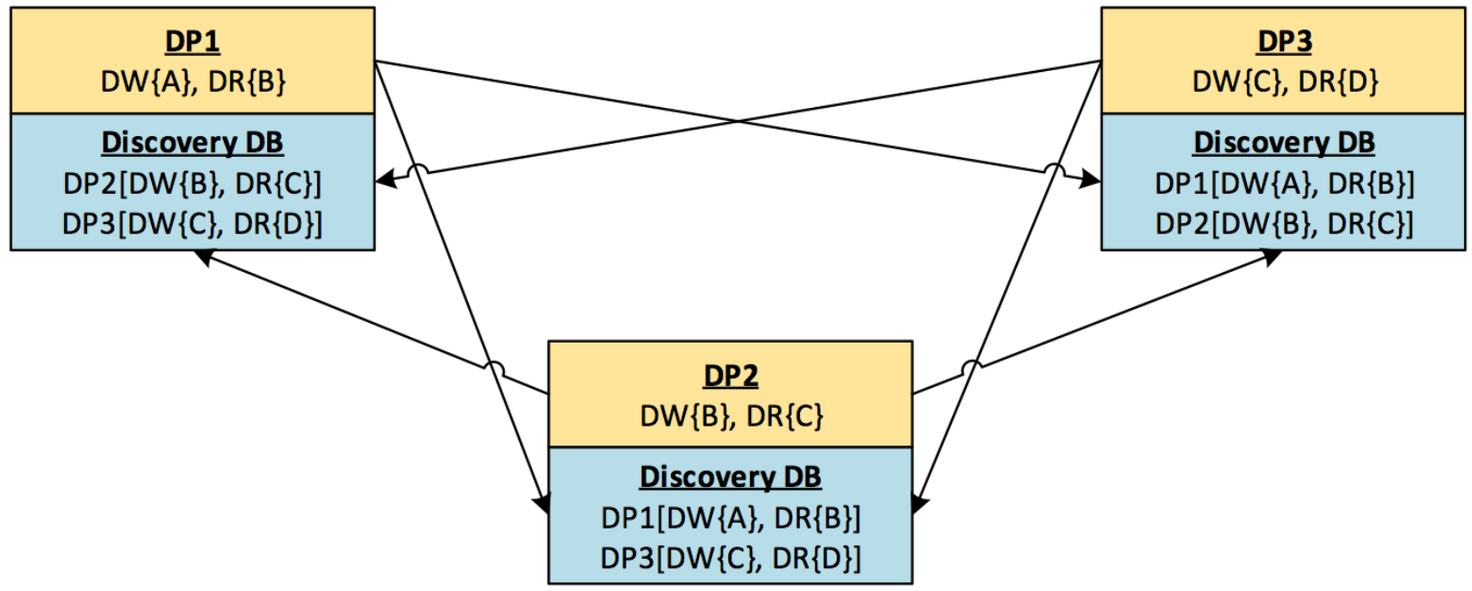
- Creates built-in entities with Content Filtered Topics (CFTs) that filter out discovery messages on topic names

Design of CFDP

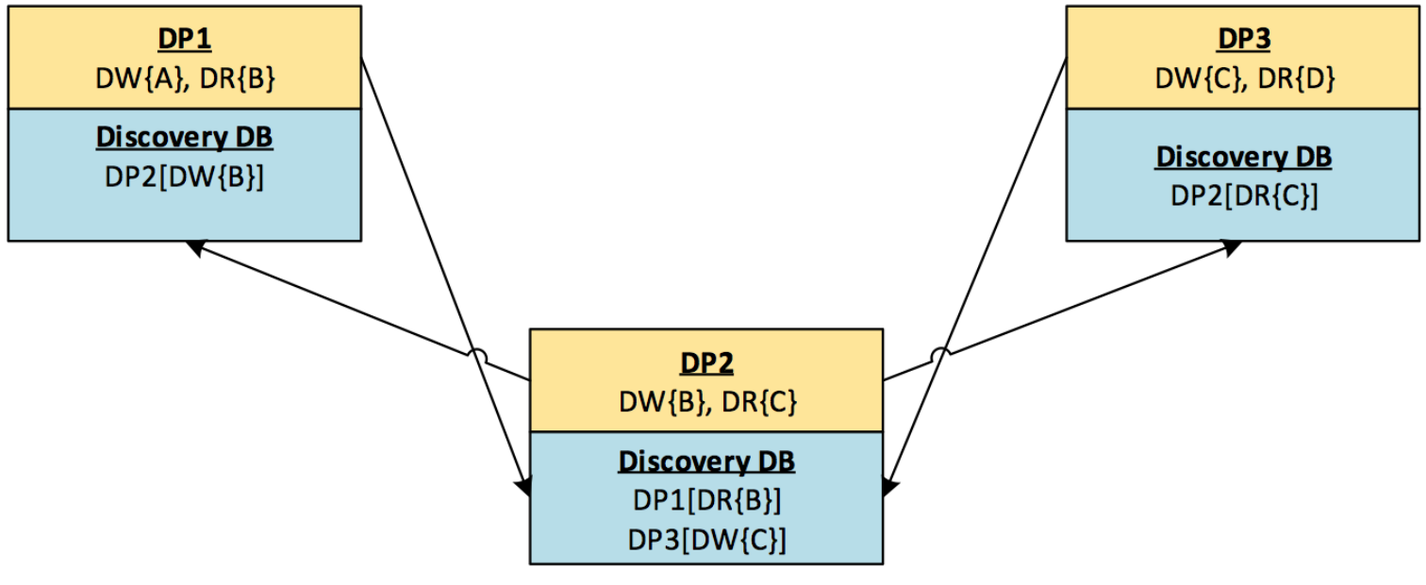


SDP and CFDP Comparison

SDP



CFDP



Lessons Learned

- CFDP is more efficient and scalable than SDP
- CFDP's current lack of support for multicast can impede scalability
- Instance-based filtering can help to make CFDP scalable in a large-scale system with a small set of topics

Kyoungho An, Sumant Tambe, Paul Pazandak, Gerardo Pardo-Castellote, Aniruddha Gokhale, and Douglas Schmidt, ***"Content-based Filtering Discovery Protocol (CFDP): Scalable and Efficient OMG DDS Discovery Protocol"***, 8th ACM International Conference on Distributed Event-Based Systems (DEBS 2014), Mumbai, India, May 26-29, 2014.

Focus Area 2:
Coordination service for pub/sub middleware in WANs

	Turbine (Machine)	Wind farm (Plant)	Power producer (Enterprise)
Focus Area 1		Scalable discovery protocols for pub/sub middleware	
Focus Area 2		**Coordination service for pub/sub middleware in WANs**	
Focus Area 3			Fault-tolerant and real-time cloud infrastructures
Focus Area 4		Performance testing with different combinations of QoS settings	

Context

- **Pub/Sub middleware for Industrial IoT systems requires non-functional properties to guarantee reliability and timeliness of data sharing between devices**

  - **Availability: Fault-tolerance via redundant publishers or routing paths**

  - **Durability: Data consistency for late joiners**

  - **Reliability: Ensuring data delivery**

  - **Timeliness: Prioritizing based on latency requirements, Deadline-based notifications**

Context

- **Pub/Sub middleware for Industrial IoT systems requires dynamic discovery of brokers and endpoints in different networks**
  - **Reduce manual efforts on system configurations**
  - **Decouple systems in time and space**
- **OMG DDS supports configurable QoS policies, and dynamic discovery in LANs**

Challenges

- DDS for WAN-based Industrial IoT systems is limited because of multicast and peer-to-peer communications (e.g. multicast not supported, NAT and firewall)

- DDS broker solutions exist to resolve the issues
  - DDS Proxy developed by A. Hakiri et al.
  - DDS Routing Service by Real-Time Innovations (RTI)

- A middleware solution to discover and coordinate DDS brokers for WAN-based large-scale applications does not exist

- It is challenging to provide scalability and expected latency as well as consistency of dynamic data dissemination paths on overlay networks

Related Work

Related Research
M. Li, F. Ye, M. Kim, H. Chen, and H. Lei. A scalable and elastic publish/subscribe service. In Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International, pages 1254–1265. IEEE, 2011.
M. Kim, K. Karenos, F. Ye, J. Reason, H. Lei, and K. S[illegible]agin. "Efficacy of techniques for responsiveness in a wide-area publish/subscribe syster[illegible] International Middleware Conference Industrial track. A[illegible]
N. Carvalho, F. Araujo, and L. Rodrigues, "Scalable qos[illegible] systems," in Network Computing and Applications, Four[illegible] on. IEEE, 2005, pp. 101–108.
A. Hakiri, P. Berthou, A. Gokhale, D. C. Schmidt, and G. Thierry. Supporting end-to-end scalability and real-time event dissemination in the omg data distribution service over wide area networks. Elsevier Journal of Systems Software (JSS), 2013.
K. Zhang and H.-A. Jacobsen, "Sdn-like: The next generation of pub/sub," arXiv preprint arXiv: 1308.0056, 2013.

Good for cloud-based scalable content-based pub/sub service, but does not support QoS policies for mission critical applications and only applicable in systems located in the cloud without considering location constraints

Related Work

Related Research
M. Li, F. Ye, M. Kim, H. Chen, and H. Lei. A scalable and elastic publish/subscribe service. In Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International, pages 1254–1265. IEEE, 2011.
M. Kim, K. Karenos, F. Ye, J. Reason, H. Lei, and K. Shagin, “Efficacy of techniques for responsiveness in a wide-area publish/subscribe system,” in Proceedings of the 11th International Middleware Conference Industrial track. ACM, 2010, pp. 40–45.
N. Carvalho, F. Araujo, and L. Rodrigues, “Scalable qos-based event routing in publish-subscribe systems,” in Network Computing and Applications, Fourth IEEE International Symposium on. IEEE, 2005, pp. 101–108.
A. Hakiri, P. Berthou, A. Gokhale, D. C. Schmidt, and G. Thierry. Supporting end-to-end scalability and real-time event dissemination in the omg data distribution service over wide area networks. Elsevier Journal of Systems Software (JSS), 2013.
K. Zhang and H.-A. Jacobsen, “Sdn-like: The next generation of pub/sub,” arXiv preprint arXiv: 1308.0056, 2013.

Good for improving QoS in terms of latency for broker-based pub/sub systems

Related Work

Related Research
M. Li, F. Ye, M. Kim, H. Chen, and H. Lei. A scalable and elastic publish/subscribe service. In Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International, pages 1254–1265. IEEE, 2011.
M. Kim, K. Karenos, F. Ye, J. Reason, H. Lei, and K. Shagin, “Efficacy of techniques for responsiveness in a wide-area publish/subscribe system,” in Proceedings of the 11th International Middleware Conference Industrial track. ACM, 2010, pp. 40–45.
N. Carvalho, F. Araujo, and L. Rodrigues, “Scalable qos-based event routing in publish-subscribe systems,” in Network Computing and Applications, Fourth IEEE International Symposium on. IEEE, 2005, pp. 101–108.
A. Hakiri, P. Berthou, A. Gokhale, D. C. Schmidt, and G. Thierry, Supporting end-to-end scalability and real-time event dissemination in the omg data distrib Elsevier Journal of Systems Software (JSS), 2013.
K. Zhang and H.-A. Jacobsen, “Sdn-like: The next generation of pub/sub,” arXiv preprint arXiv: 1308.0056, 2013.

Good for providing required QoS in terms of latency by reserving network bandwidth for DHT-based pub/sub systems

Related Work

Related Research
M. Li, F. Ye, M. Kim, H. Chen, and H. Lei. A scalable and elastic publish/subscribe service. In Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International, pages 1254–1265. IEEE, 2011.
M. Kim, K. Karenos, F. Ye, J. Reason, H. Lei, and K. Shagin, “Efficacy of techniques for responsiveness in a wide-area publish/subscribe syst… International Middleware Conference Industrial track.
N. Carvalho, F. Araujo, and L. Rodrigues, “Scalable c… systems,” in Network Computing and Applications, F… on. IEEE, 2005, pp. 101–108.
A. Hakiri, P. Berthou, A. Gokhale, D. C. Schmidt, and G. Thierry. Supporting end-to-end scalability and real-time event dissemination in the omg data distribution service over wide area networks. Elsevier Journal of Systems Software (JSS), 2013.
K. Zhang and H.-A. Jacobsen, “Sdn-like: The next generation of pub/sub,” arXiv preprint arXiv: 1308.0056, 2013.

Good for connecting DDS endpoints located in different networks, but requires manual configurations for brokers

Related Work

Related Research
M. Li, F. Ye, M. Kim, H. Chen, and H. Lei. A scalable and elastic publish/subscribe service. In Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International, pages 1254–1265. IEEE, 2011.
M. Kim, K. Karenos, F. Ye, J. Reason, H. Lei, and K. Shagin, “Efficacy of techniques for responsiveness in a wide-area publish/subscribe system,” in Proceedings of the 11th International Middleware Conference Industrial track. ACM, 2010, pp. 40–45.
N. Carvalho, F. Araujo, and L. Rodrigues, “Scalable qos-based event routing in publish-subscribe systems,” in Network Computing and Applications, Fourth IEEE International Symposium on. IEEE, 2005, pp. 101–108.
A. Hakiri, P. Berthou, A. Gokhale, D. C. Schmidt, and G. and real-time event dissemination in the omg data distri Elsevier Journal of Systems Software (JSS), 2013.
K. Zhang and H.-A. Jacobsen, “Sdn-like: The next generation of pub/sub,” arXiv preprint arXiv: 1308.0056, 2013.

Separating control and data plane in the next generation of pub/sub motivated by the SDN architecture

Solution Approach

- PubSubCoord: Cloud-enabled discovery and coordination service for WAN-based DDS applications

  - Automatic discovery mechanism: Exploiting event-based DDS discovery and ZooKeeper notification service

  - Scalability: Hierarchical architecture, clustering brokers by topics, and harnessing scalable cloud resources

  - Low-latency: Minimizing the number of delivery hops

  - Load balancing and fault-tolerance for clustered brokers in the cloud

ZooKeeper for PubSubCoord

- A service for coordinating distributed processes
- Discover brokers (IP addresses and ports)
- Coordinate data routing paths consistently

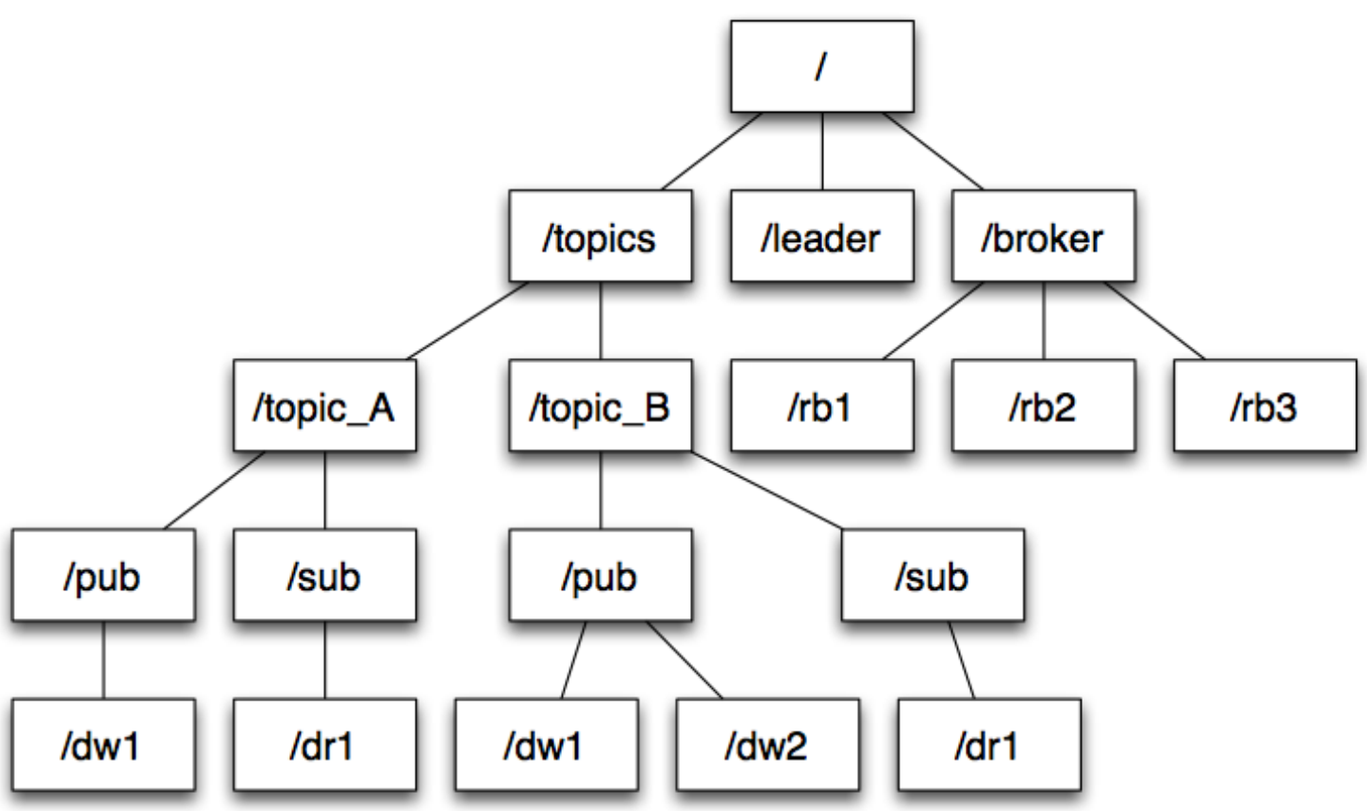
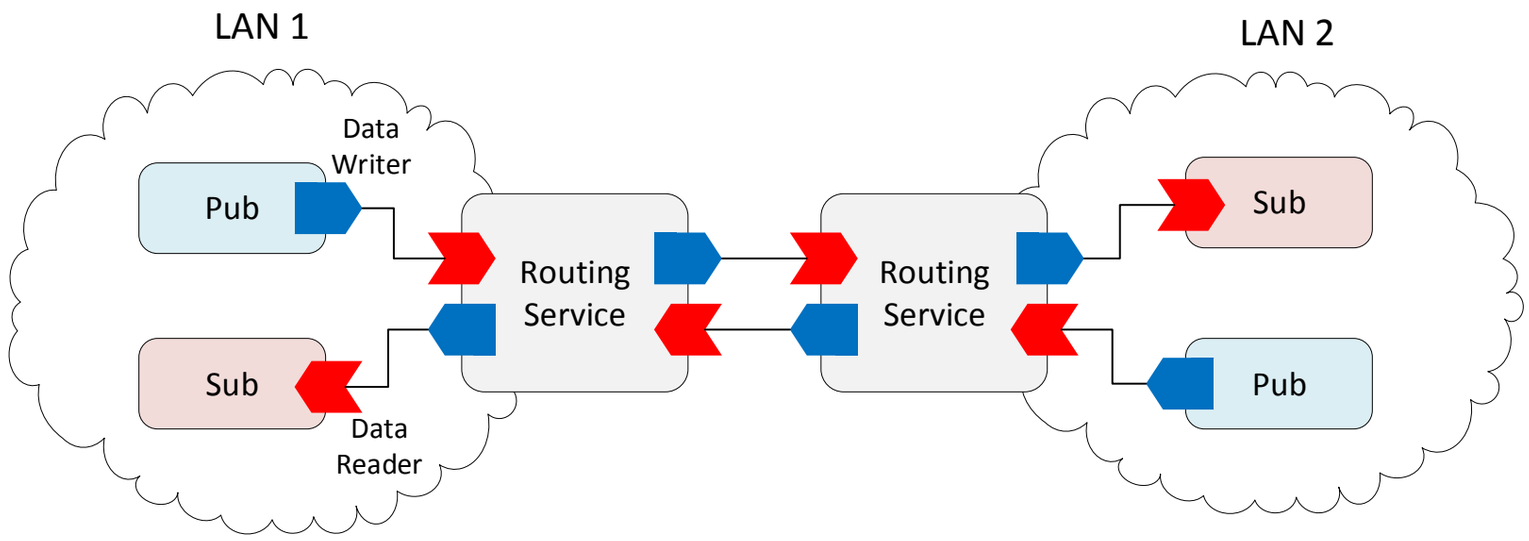


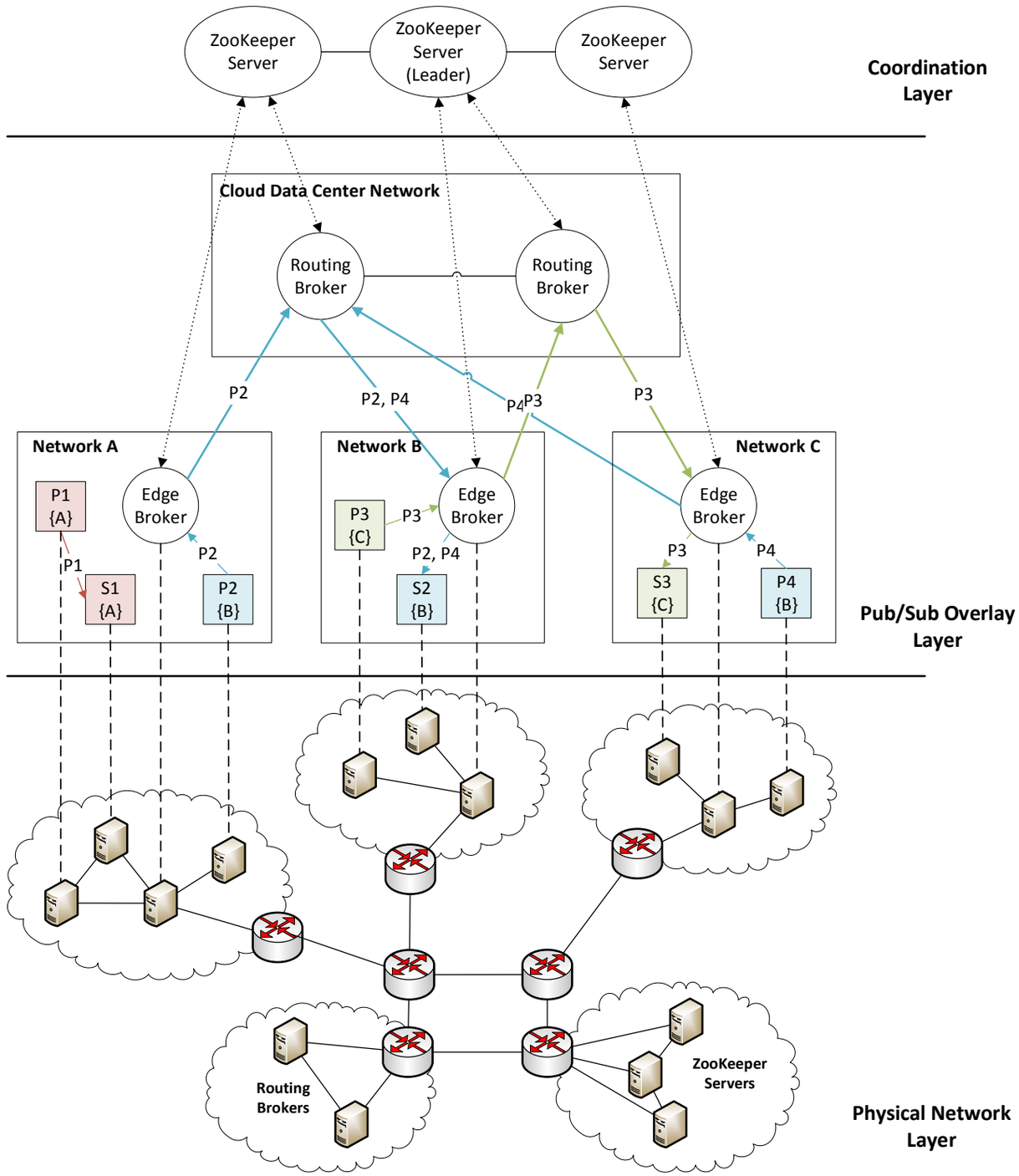
Figure1. PubSubCoord znode Data Tree Structure

DDS Routing Service for PubSubCoord

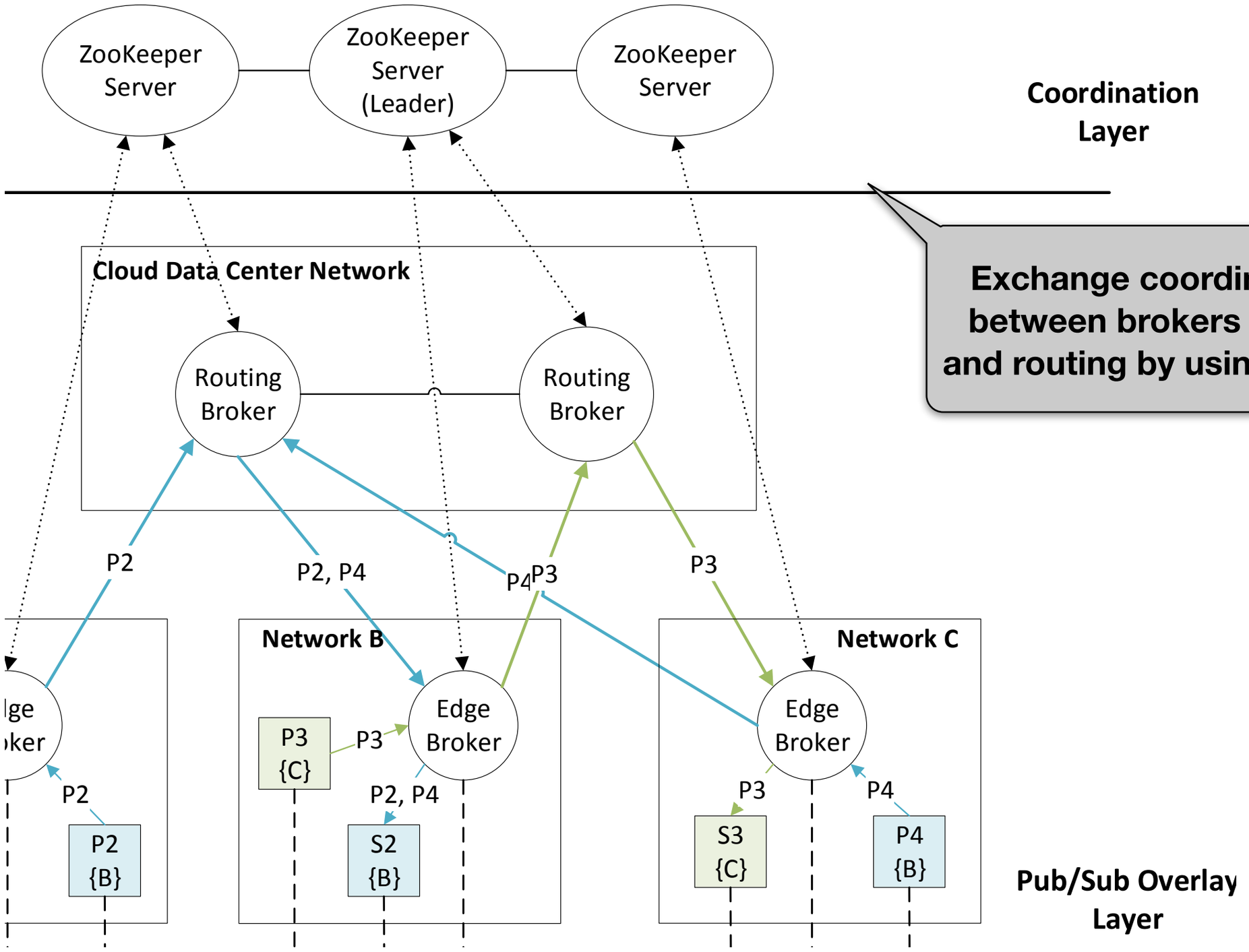
- DDS Routing Service
  - A service for integrating DDS applications deployed in isolated networks
  - Enables DDS applications to send and receive data across domains in LANs as well as WANs
- PubSubCoord uses Routing Service for…
  - Establish DDS data dissemination paths based on routing decisions by coordination logic



PubSubCoord Architecture

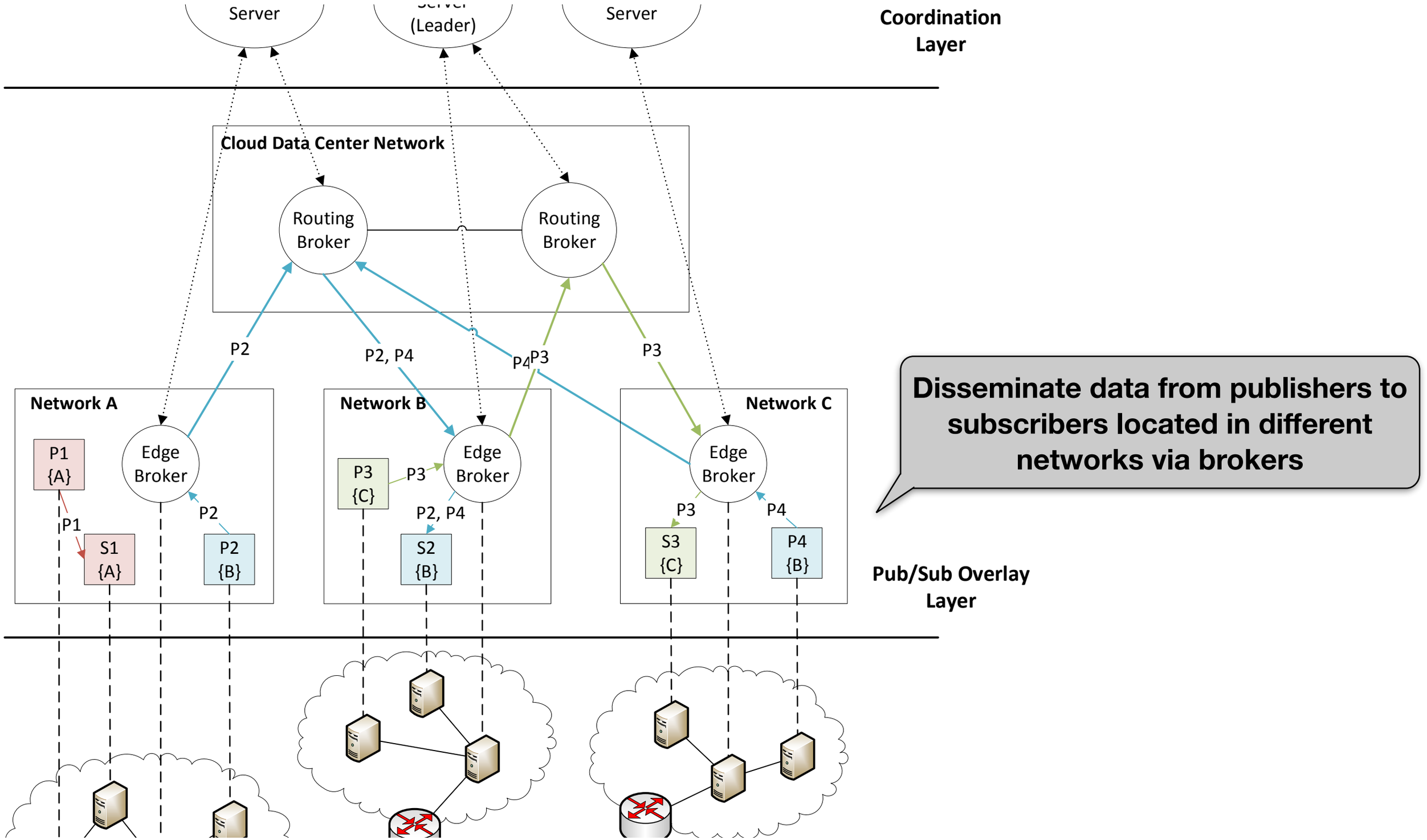


PubSubCoord Architecture



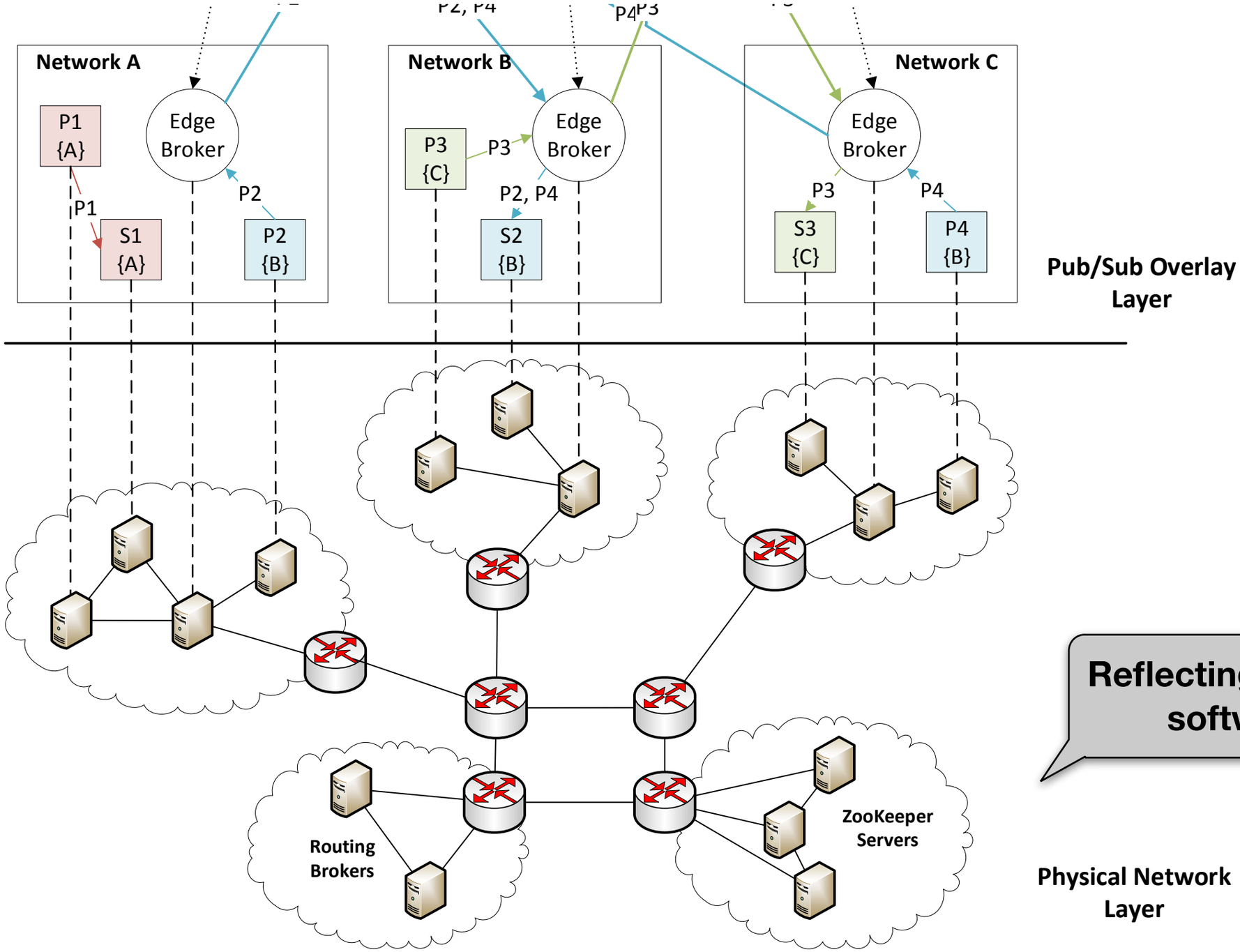
Exchange coordination information between brokers such as discovery and routing by using federated servers

PubSubCoord Architecture



Disseminate data from publishers to subscribers located in different networks via brokers

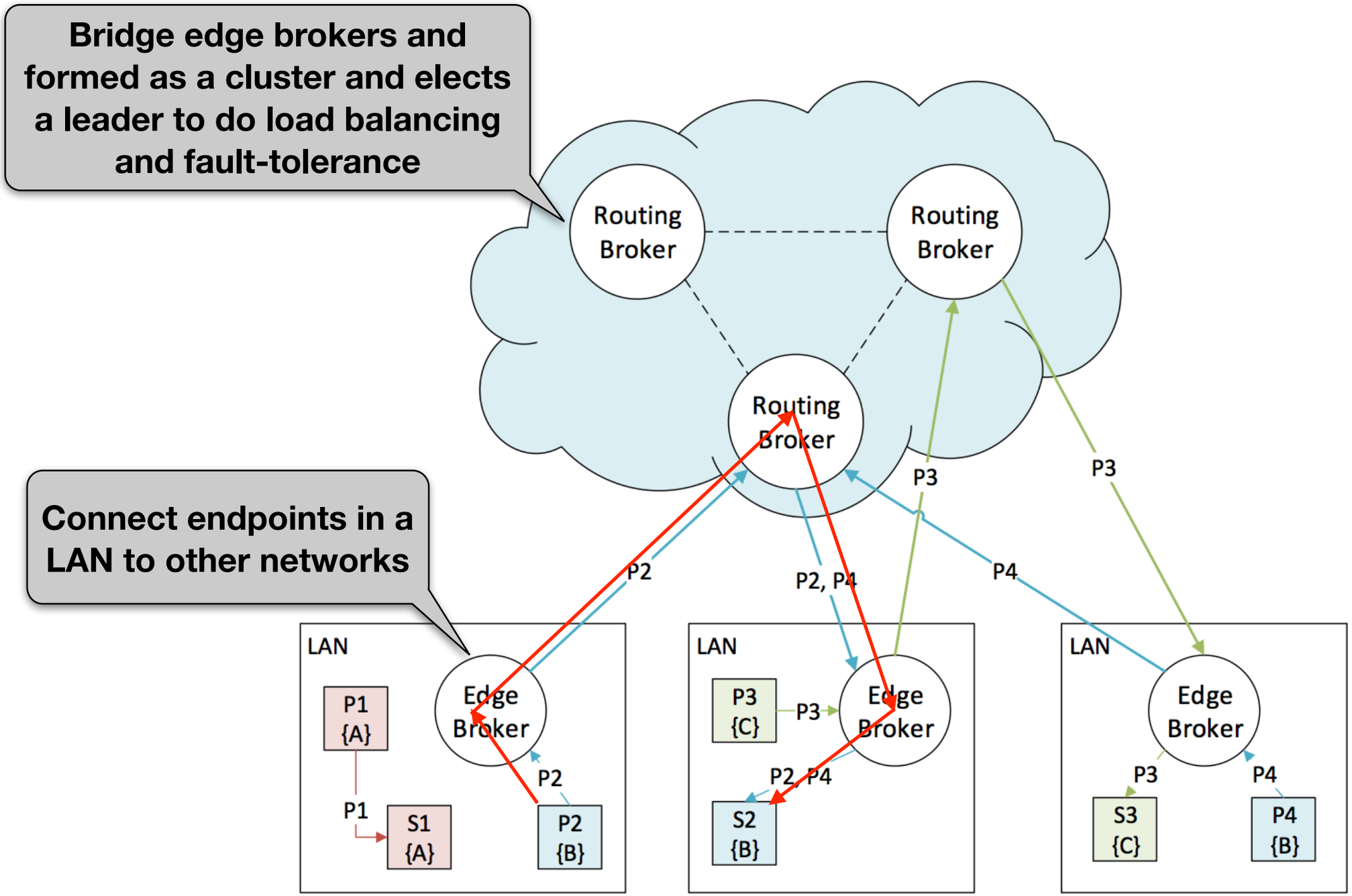
PubSubCoord Architecture



Reflecting physical locations of software components

Pub/Sub Overlay Layer

Bridge edge brokers and formed as a cluster and elects a leader to do load balancing and fault-tolerance



Connect endpoints in a LAN to other networks

PubSubCoord Architecture

- 2-level Broker Hierarchy
  - Edge brokers and Routing Brokers
  - Reduces the need for maintaining states for edge brokers
- Failed edge brokers do not affect others
- Clusters brokers according to matching topics
- Routing brokers may be overloaded, but can be scaled by cloud infrastructures

PubSubCoord Architecture

- Load balancing and Fault-tolerance
  - Cluster of routing brokers
  - Leader election among routing brokers
    - Leader balances loads of workers according to the number of assigned topics to workers
    - Leader reassign topics assigned by failed brokers to other brokers

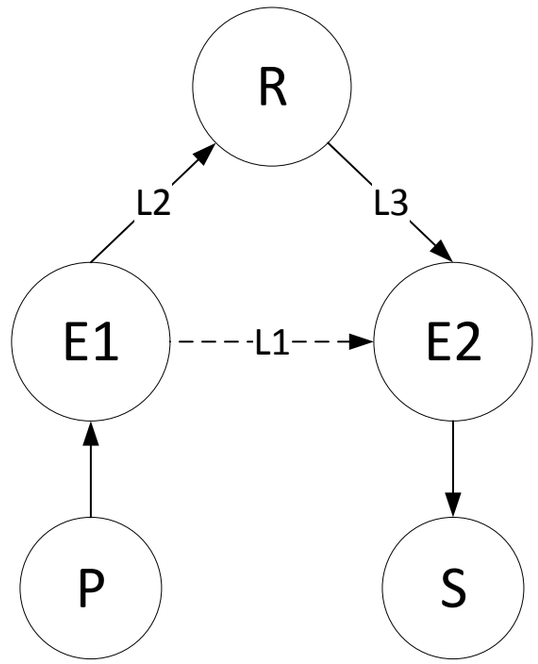
PubSubCoord Architecture

- **Deadline-aware Overlay**

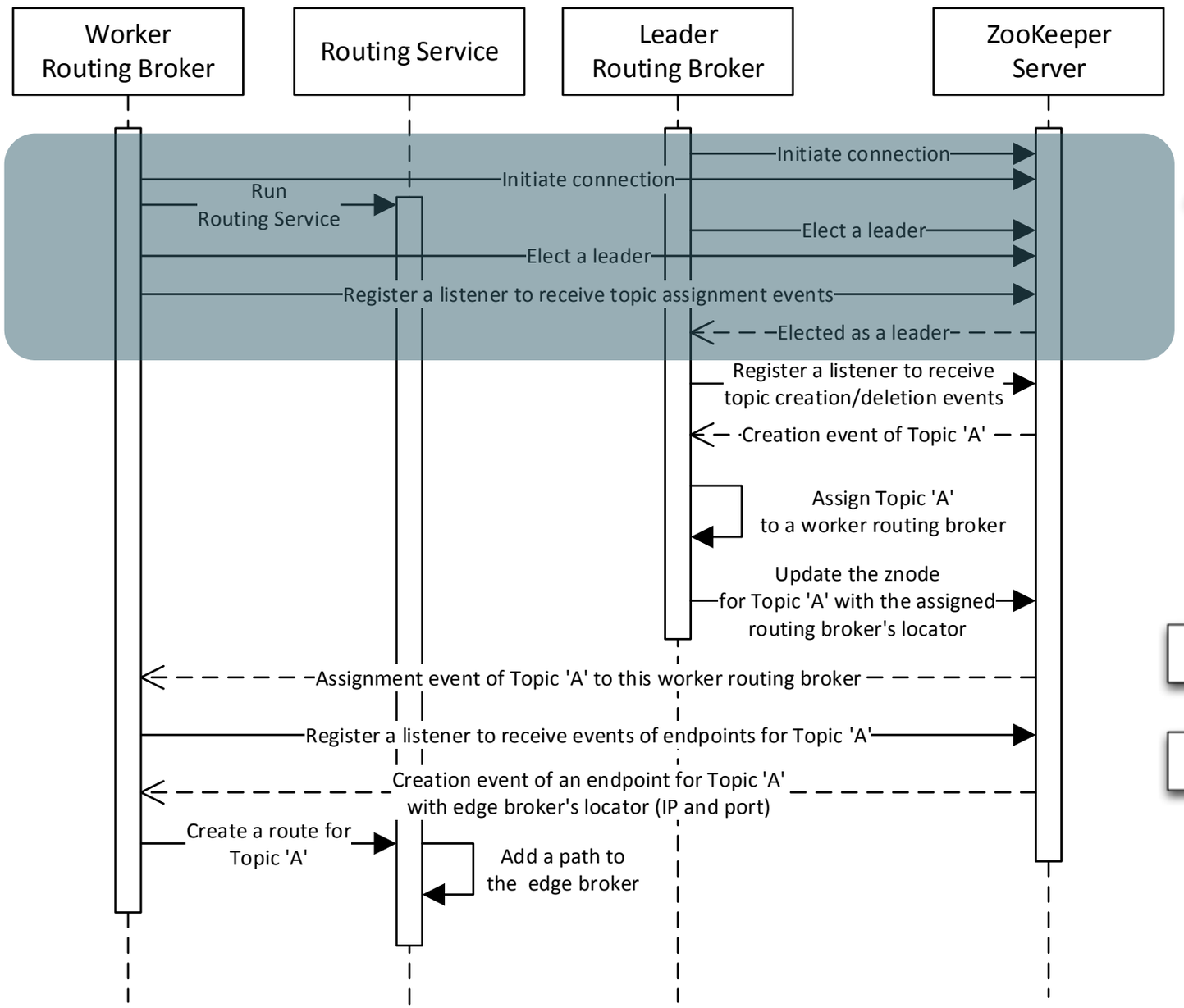
  - **Improve reliability and latency by providing an additional one hop path on the overlay that directly connects edge brokers**

  - **Leveraged by pub/sub streams that require stringent assurance and deadline-driven data delivery**

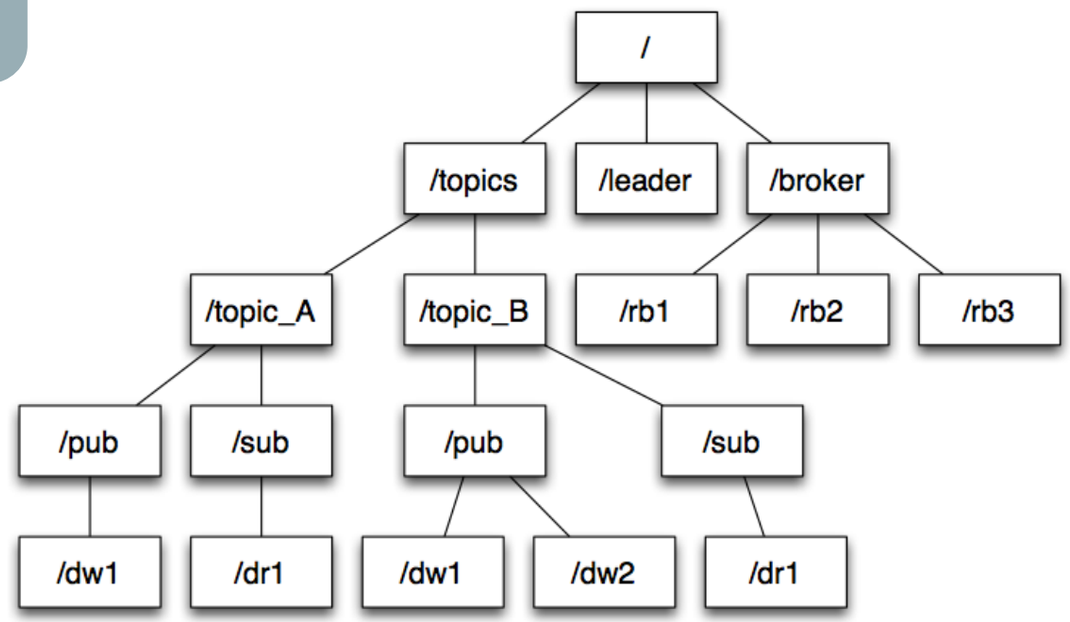
  - **Used DDS deadline QoS that expresses the maximum duration of a sample to be updated**



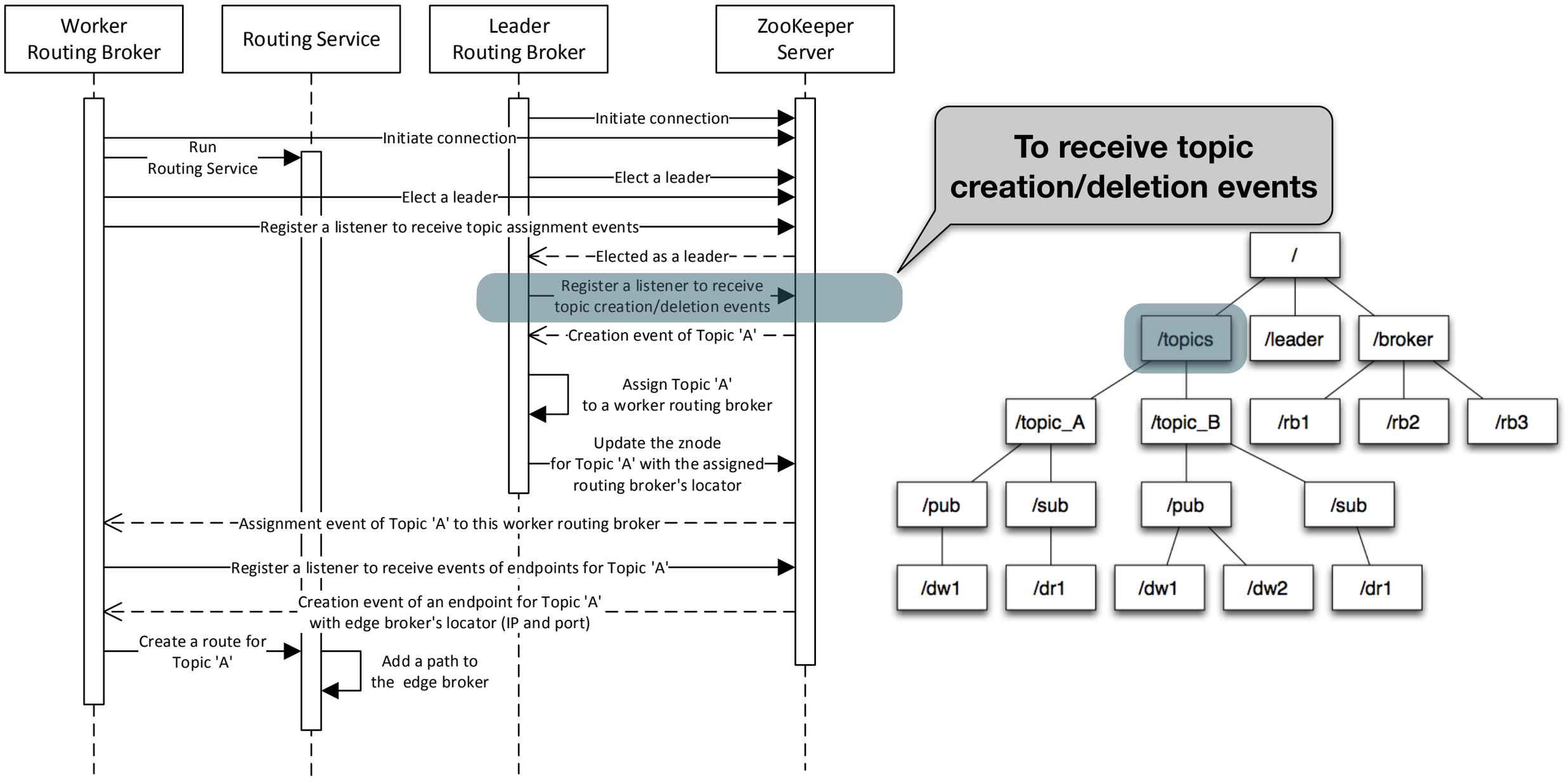
Routing Broker Sequence Diagram



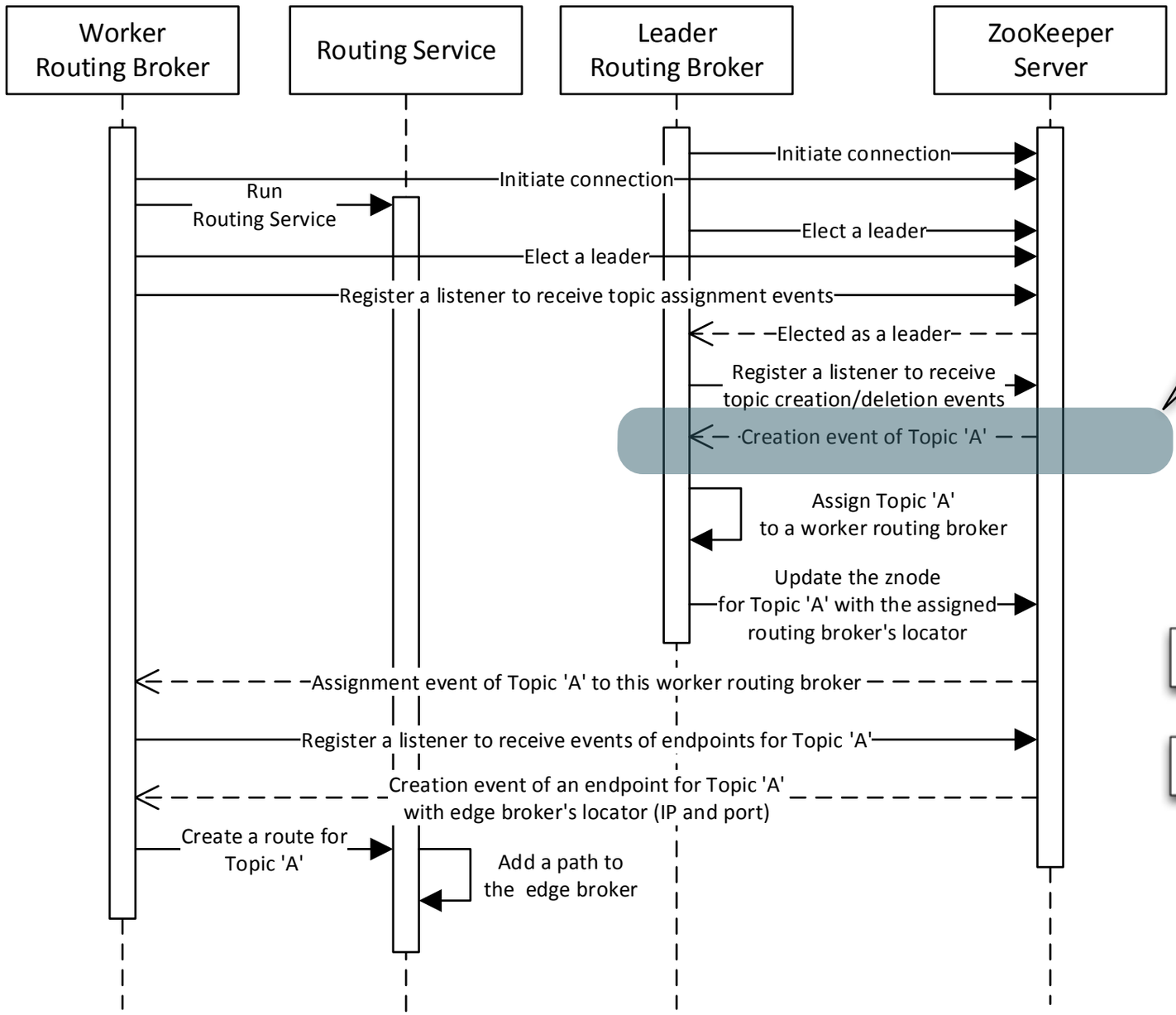
Bootstrapping - Connects to ZooKeeper servers, Run a Routing Service, and Elects a leader

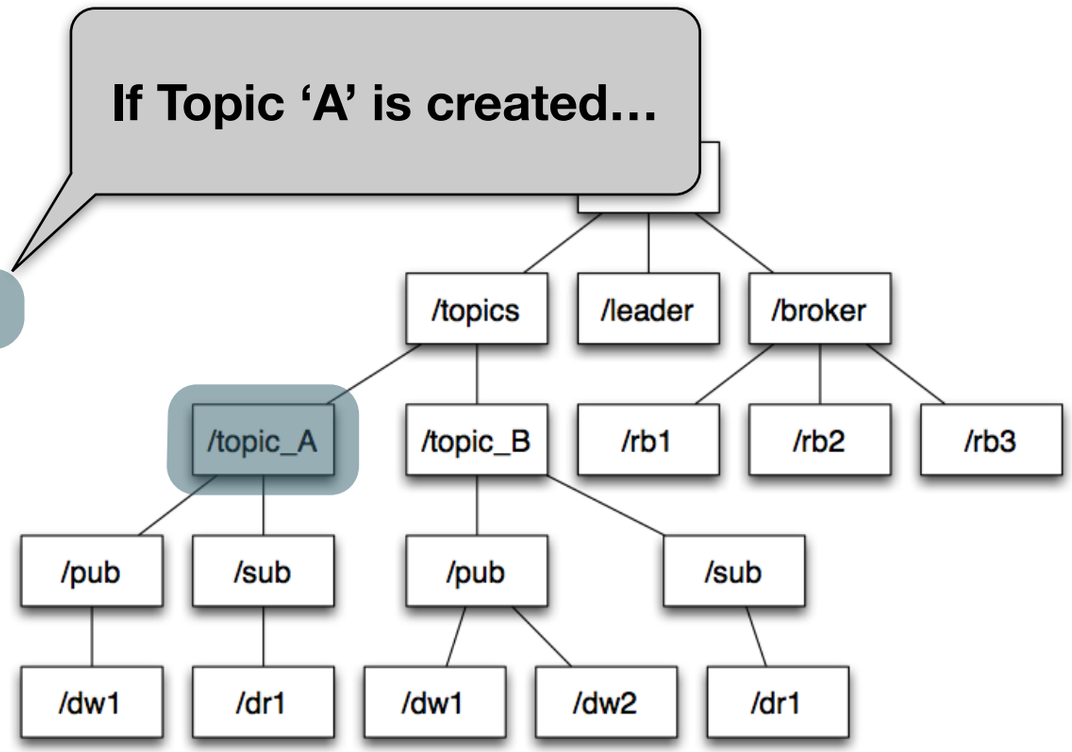


Routing Broker Sequence Diagram

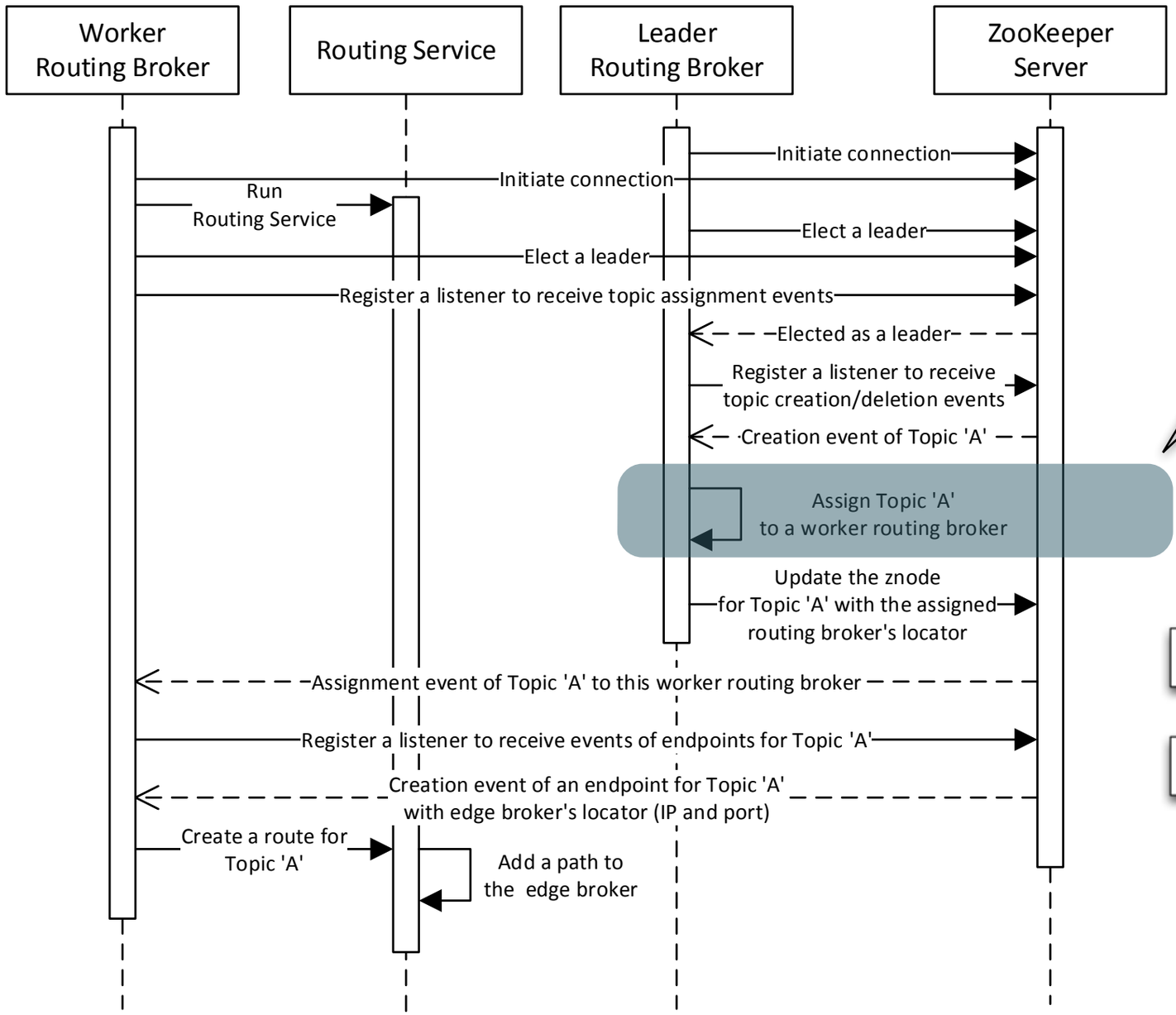


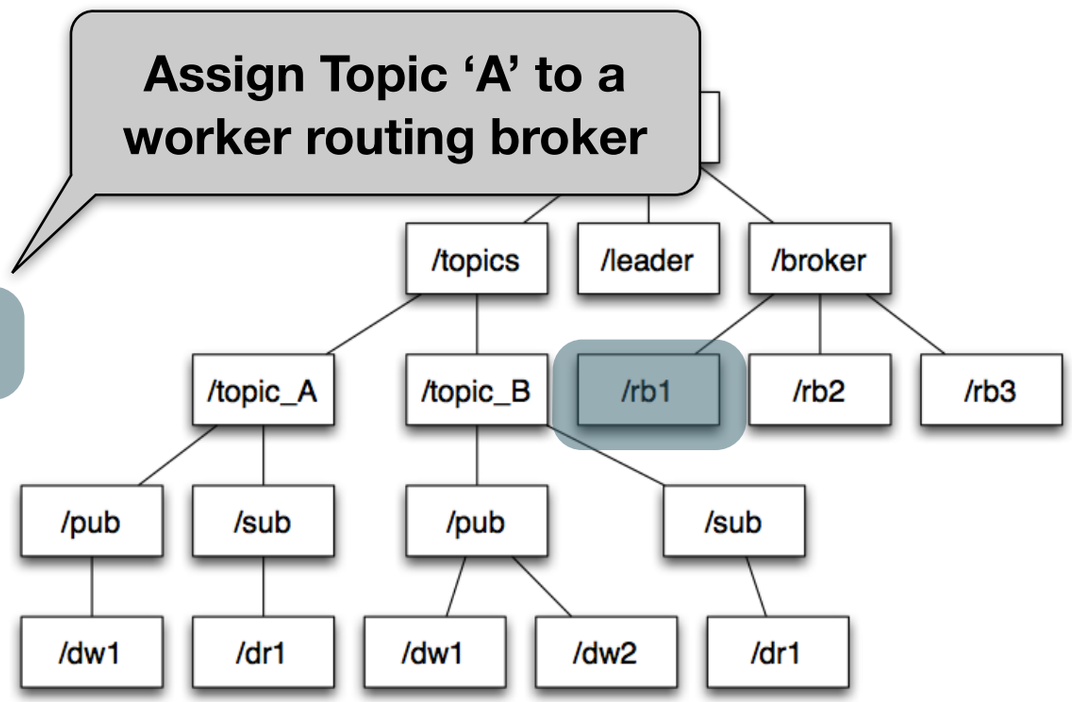
Routing Broker Sequence Diagram



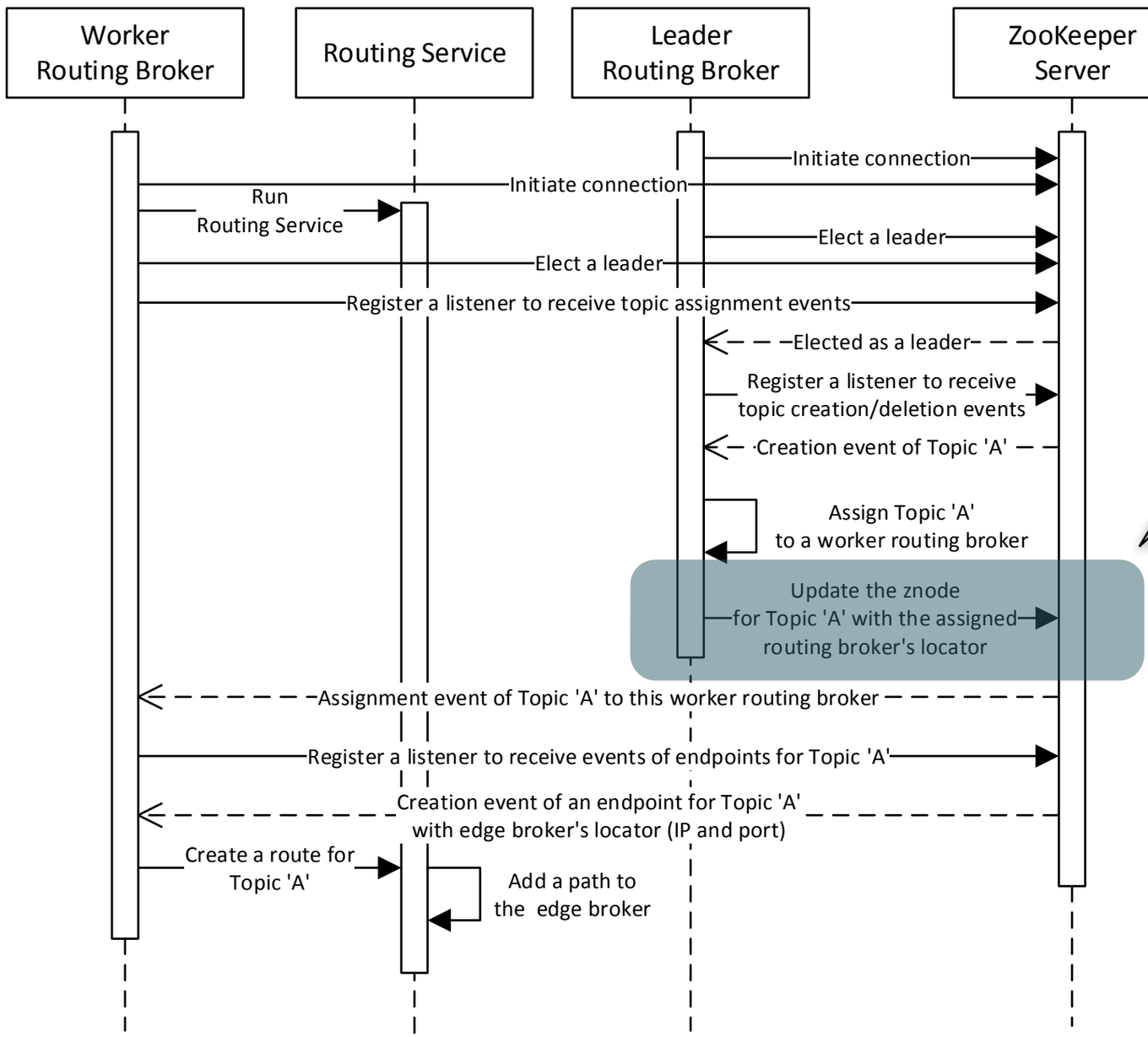


Routing Broker Sequence Diagram

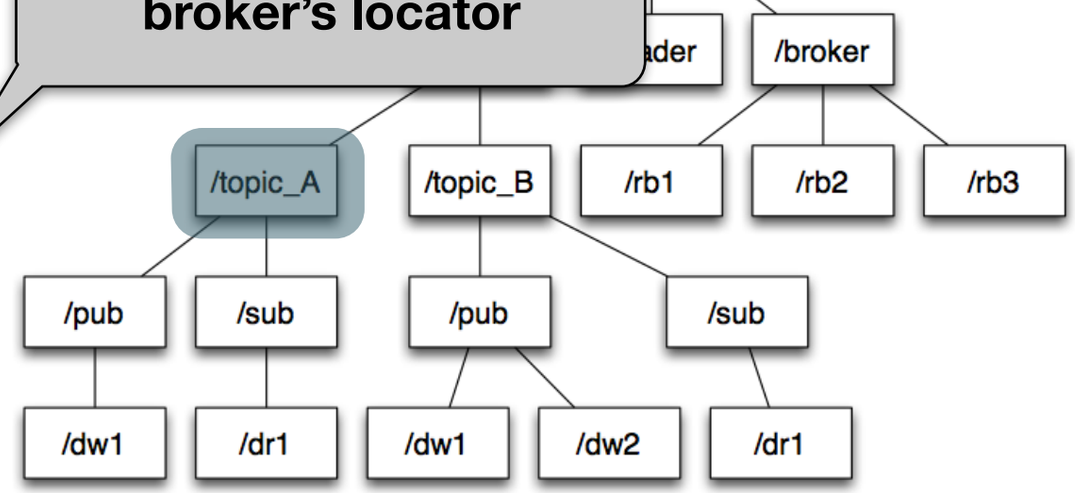




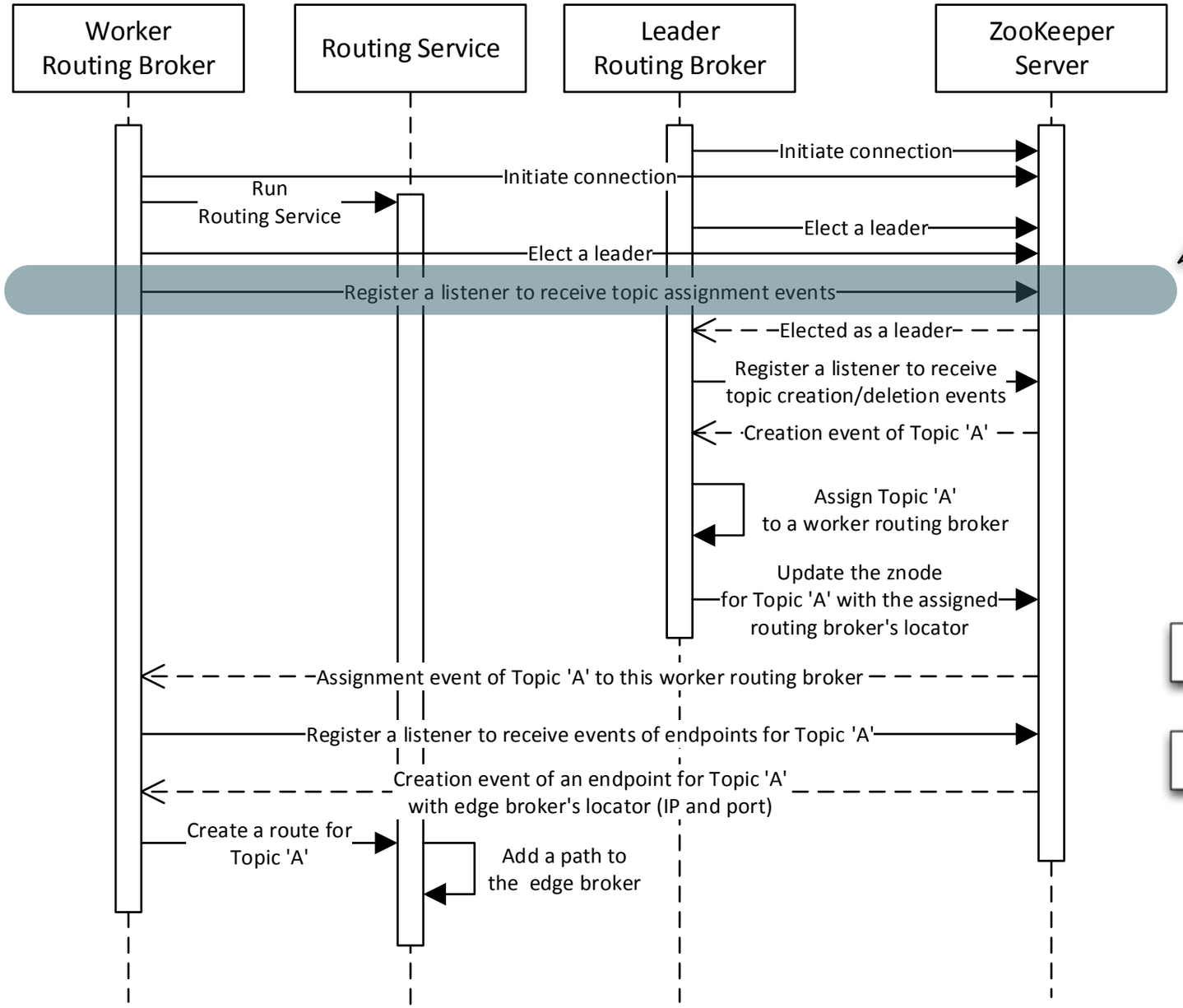
Routing Broker Sequence Diagram



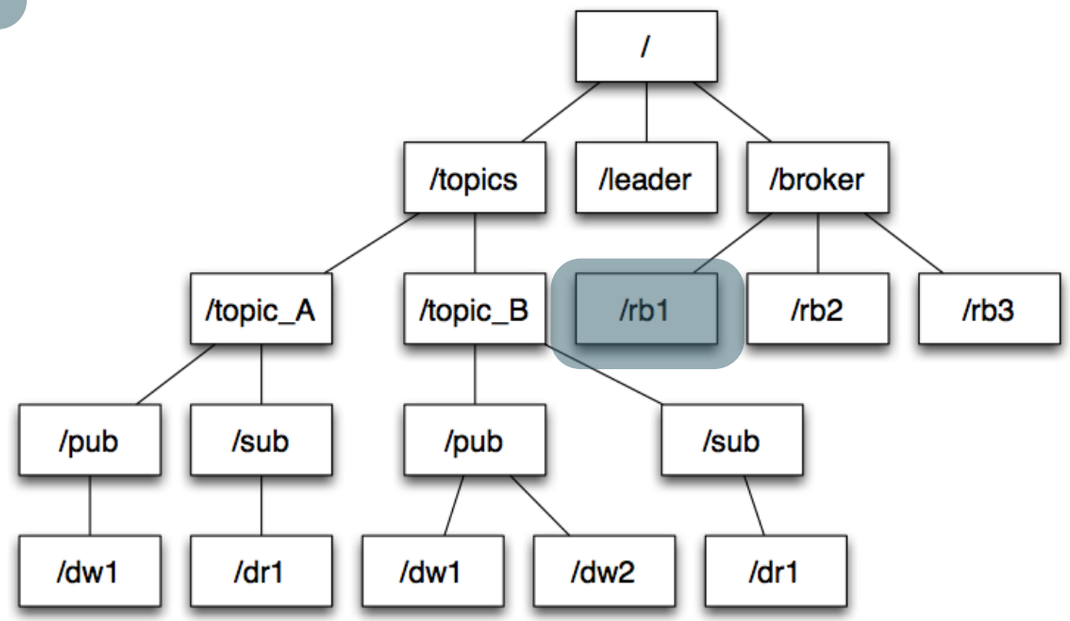
Will be used for edge brokers to receive the assigned routing broker's locator



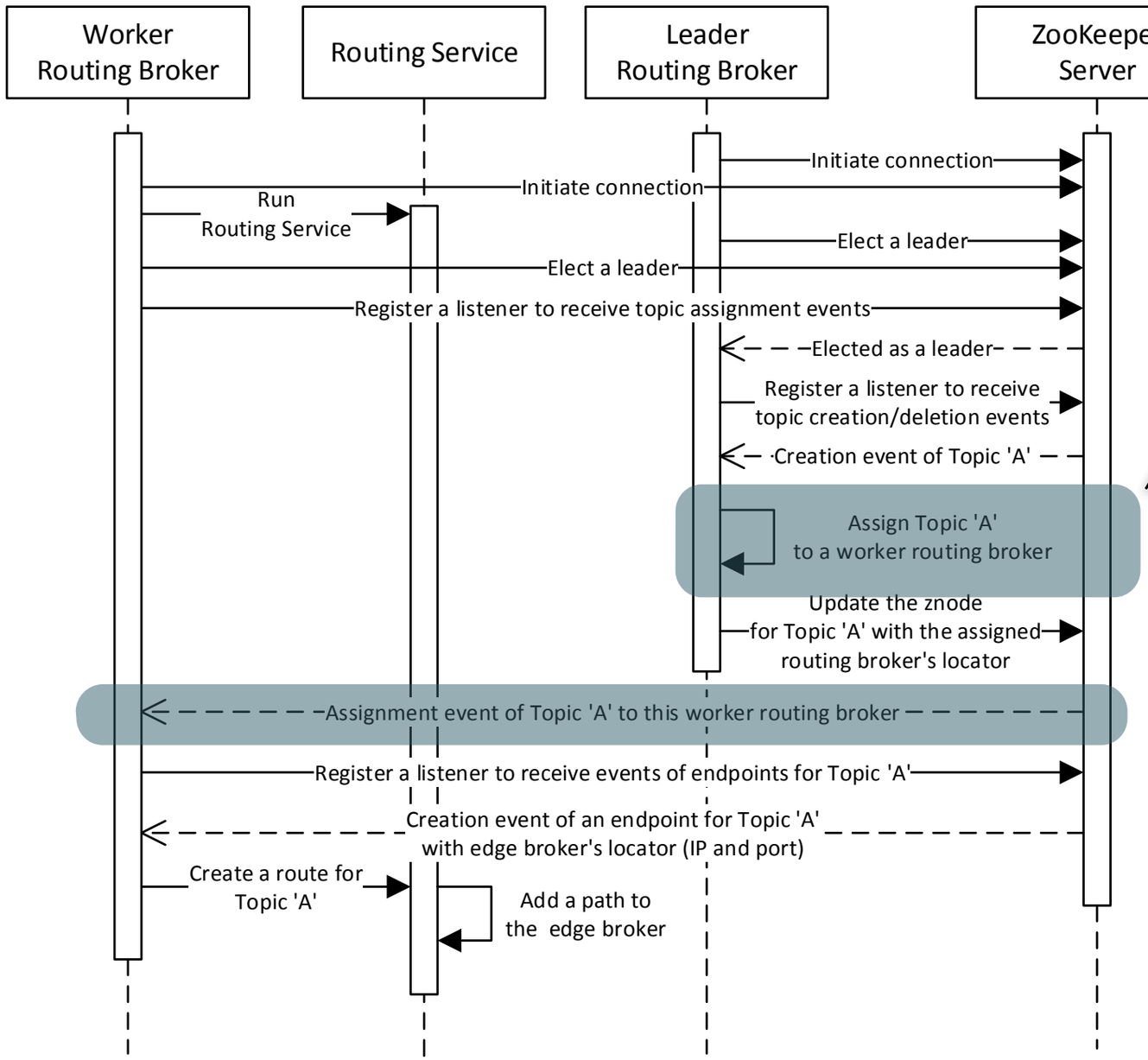
Routing Broker Sequence Diagram

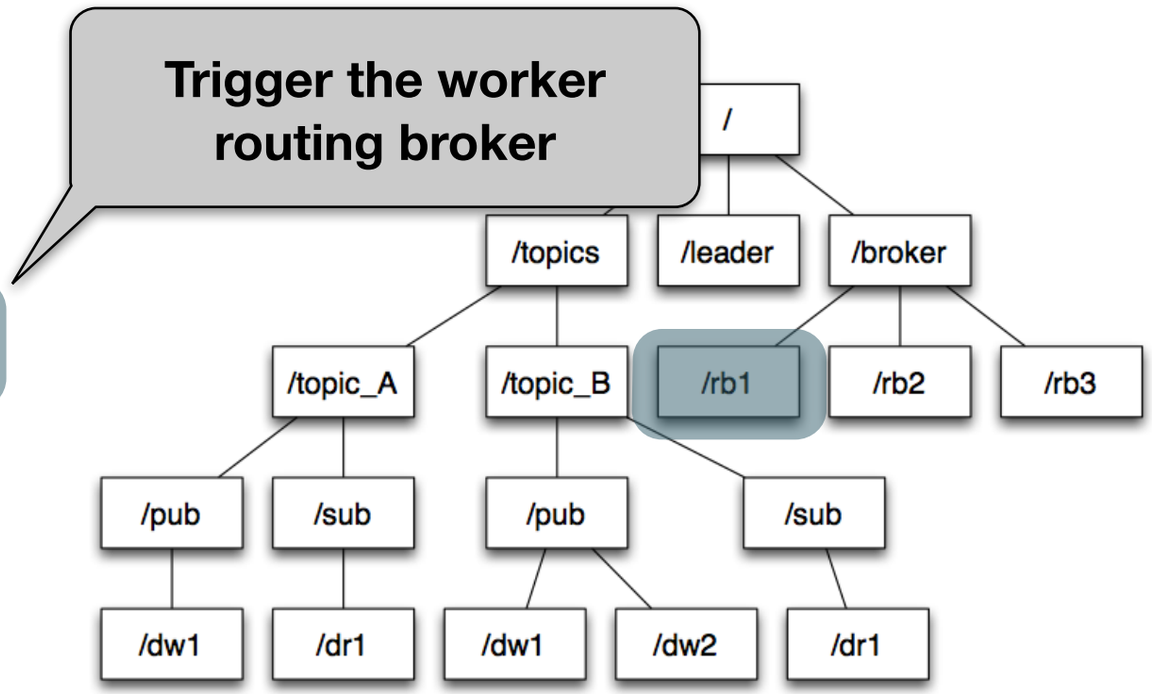


To receive topic assignment events

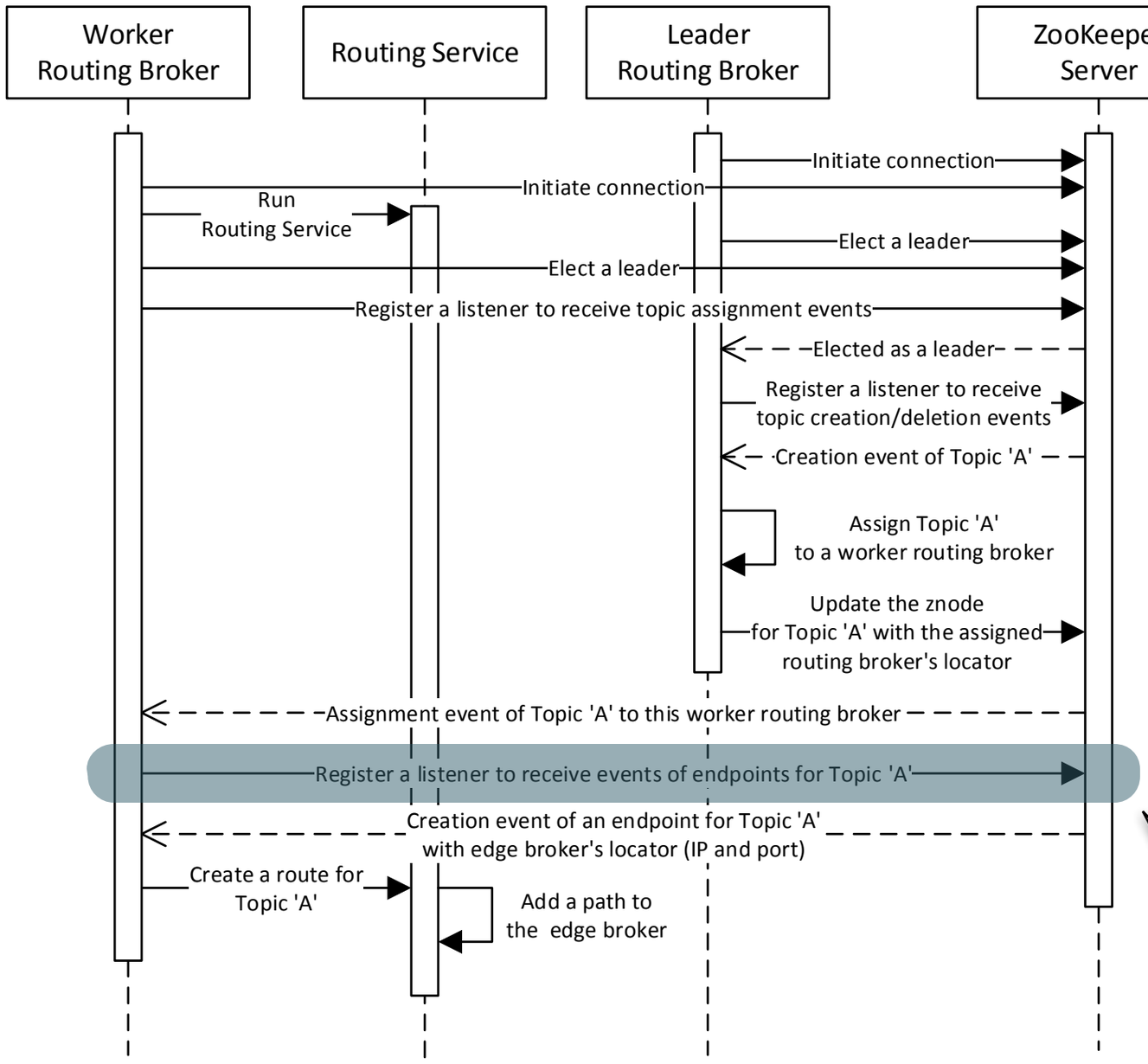


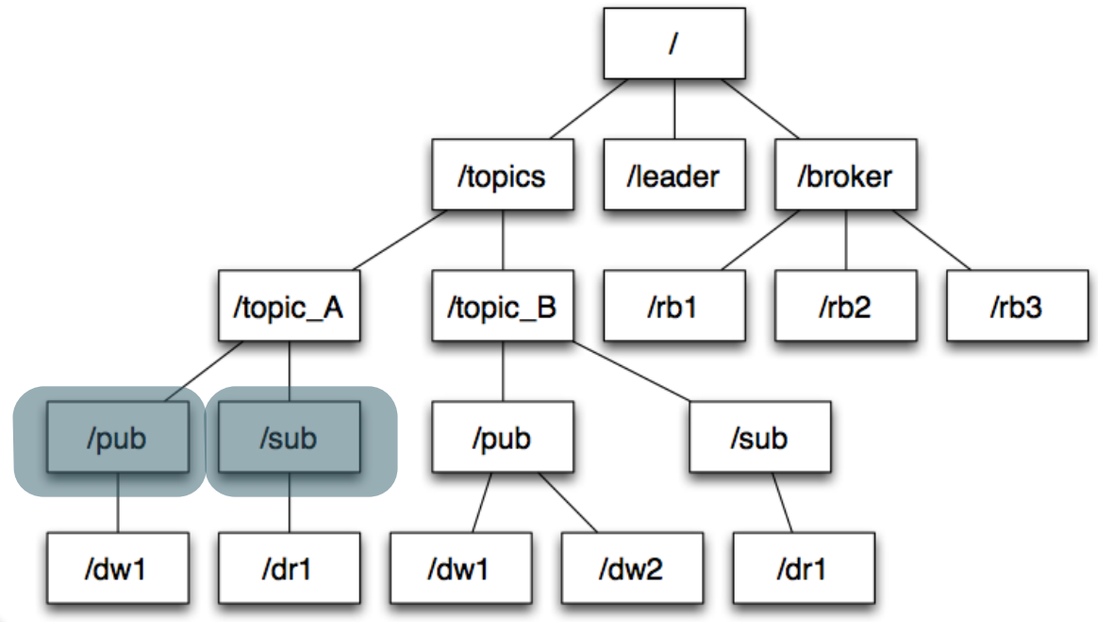
Routing Broker Sequence Diagram





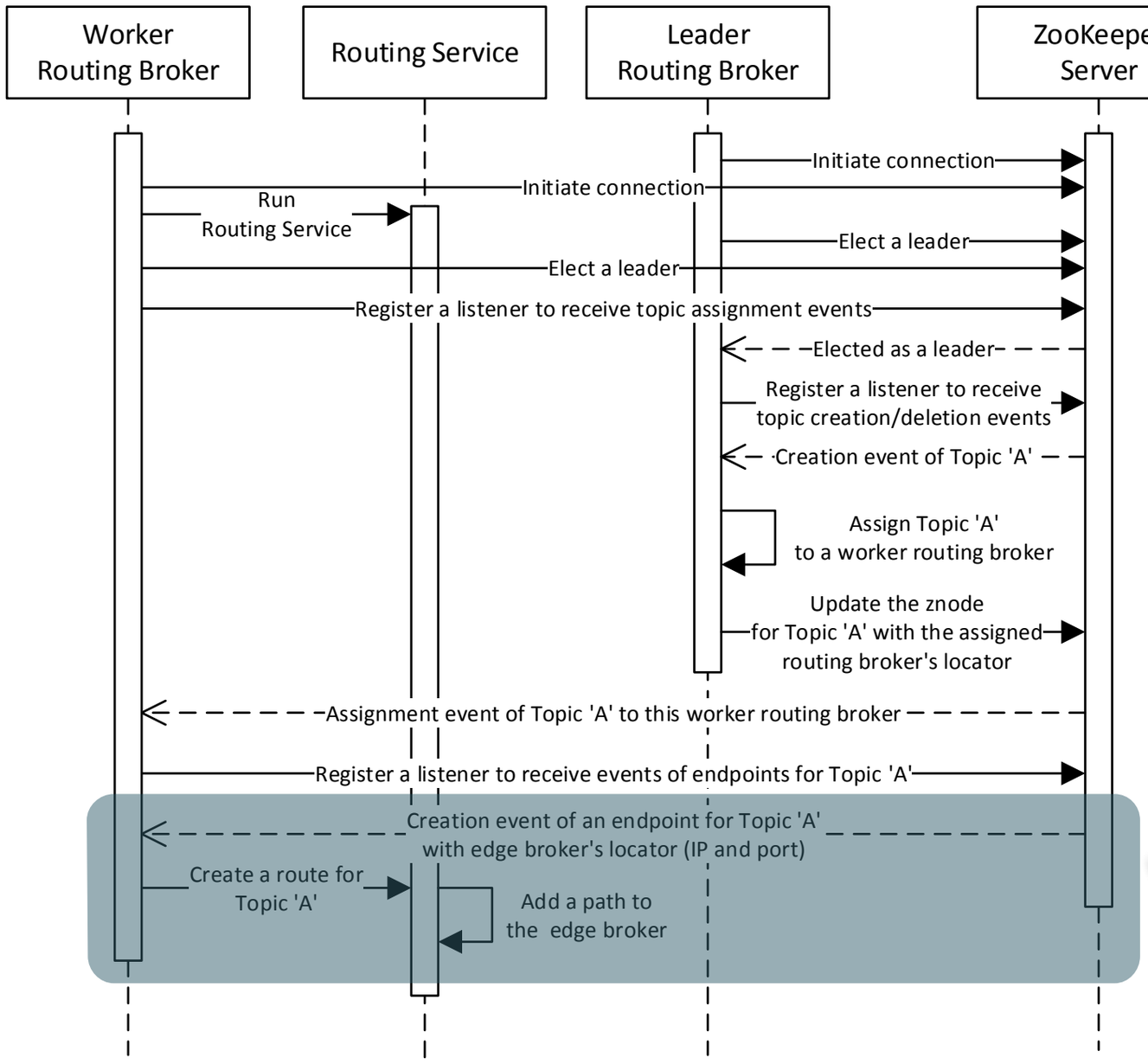
Routing Broker Sequence Diagram

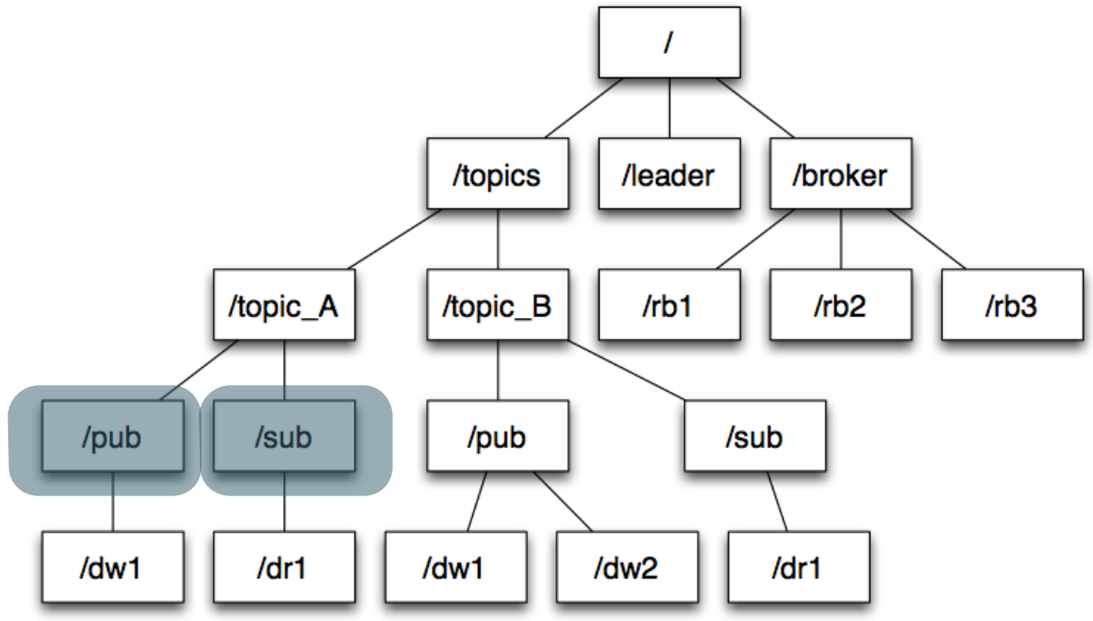




To detect creation events of pub/sub endpoints interested in Topic 'A'

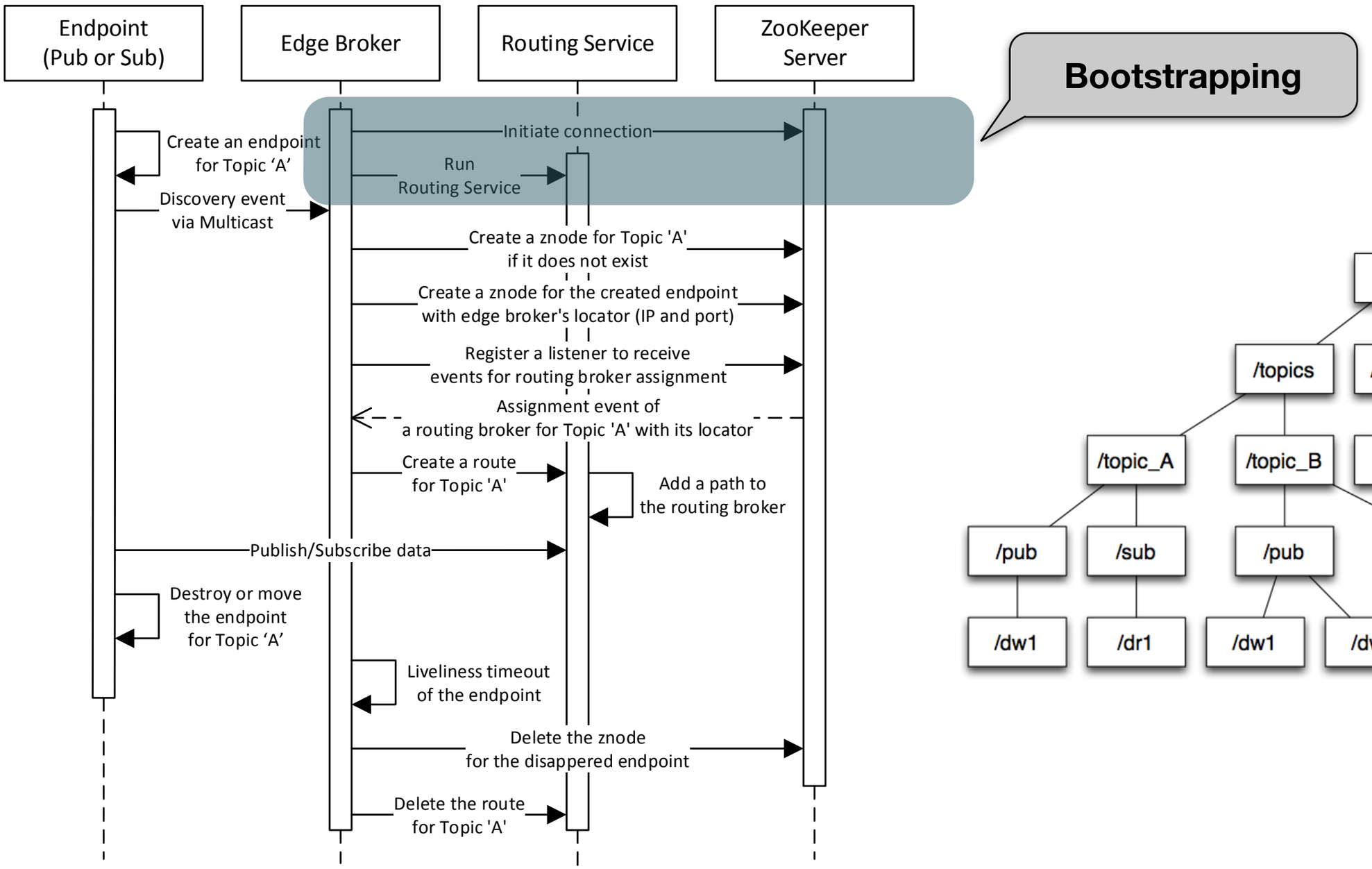
Routing Broker Sequence Diagram



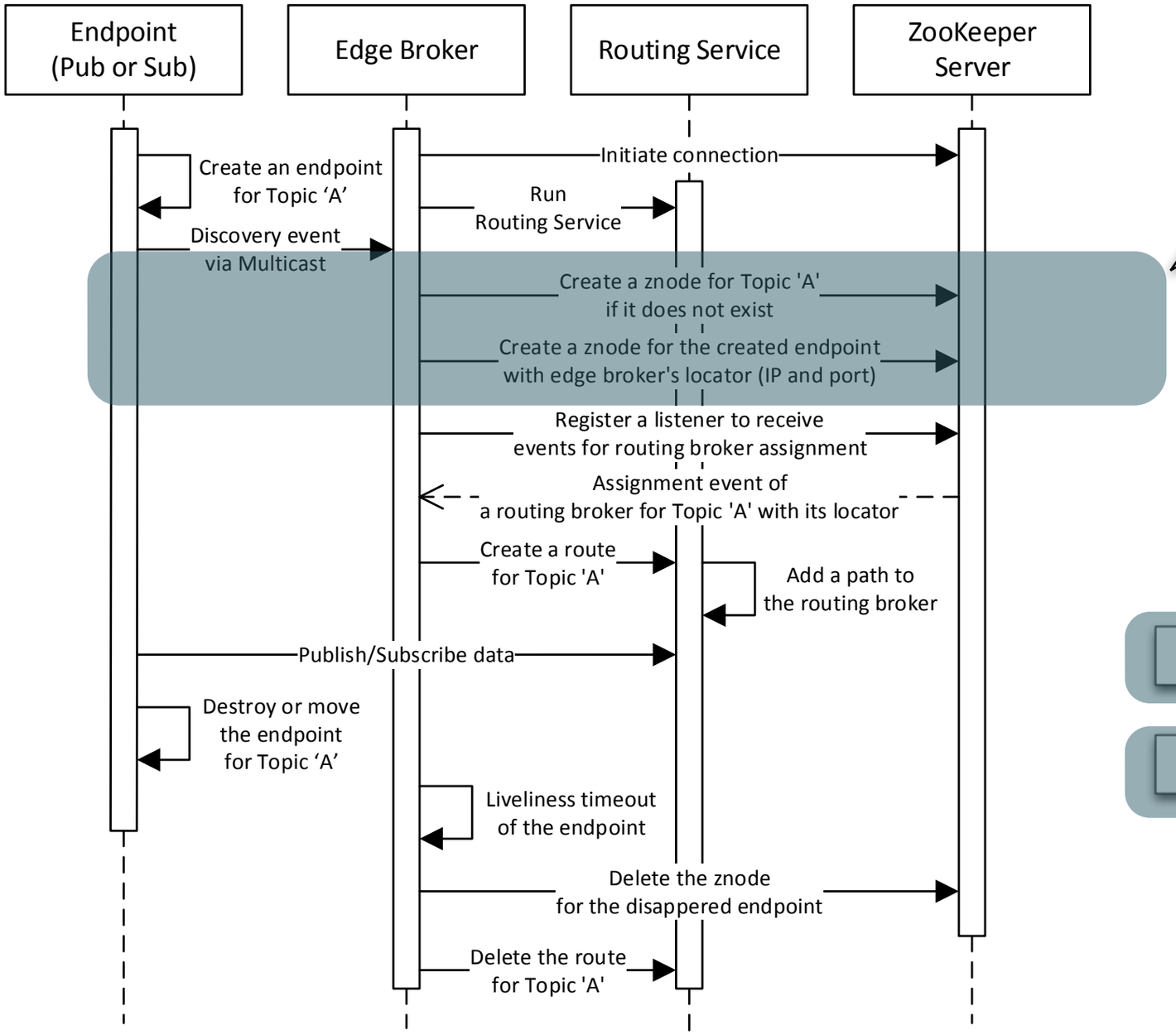


If pub/sub endpoints with Topic 'A' is created, routing paths to edge brokers are made

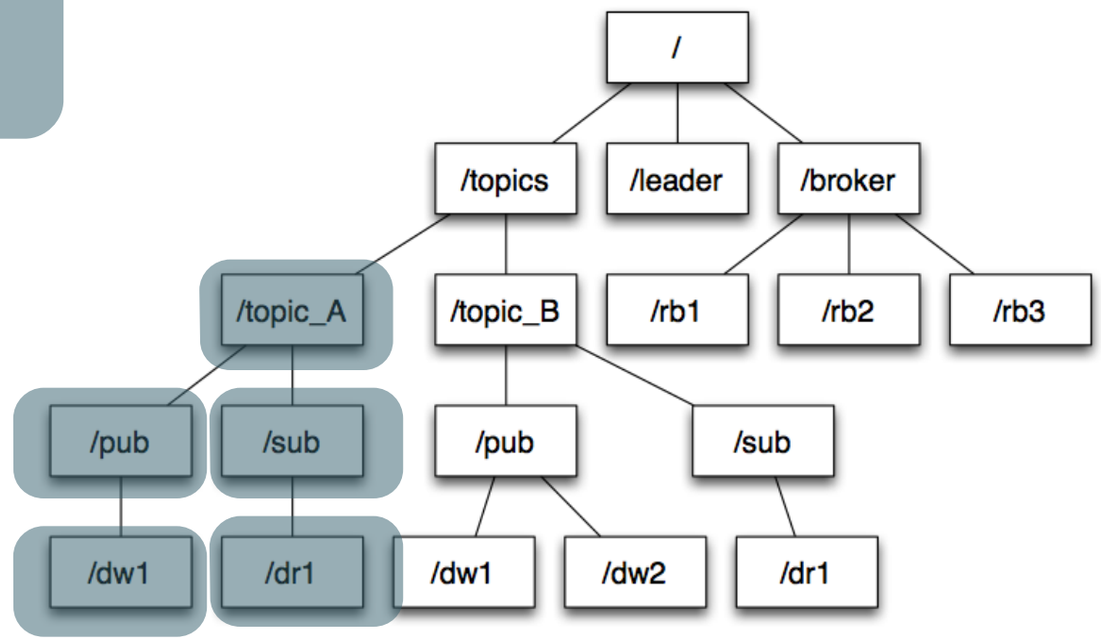
Edge Broker Sequence Diagram



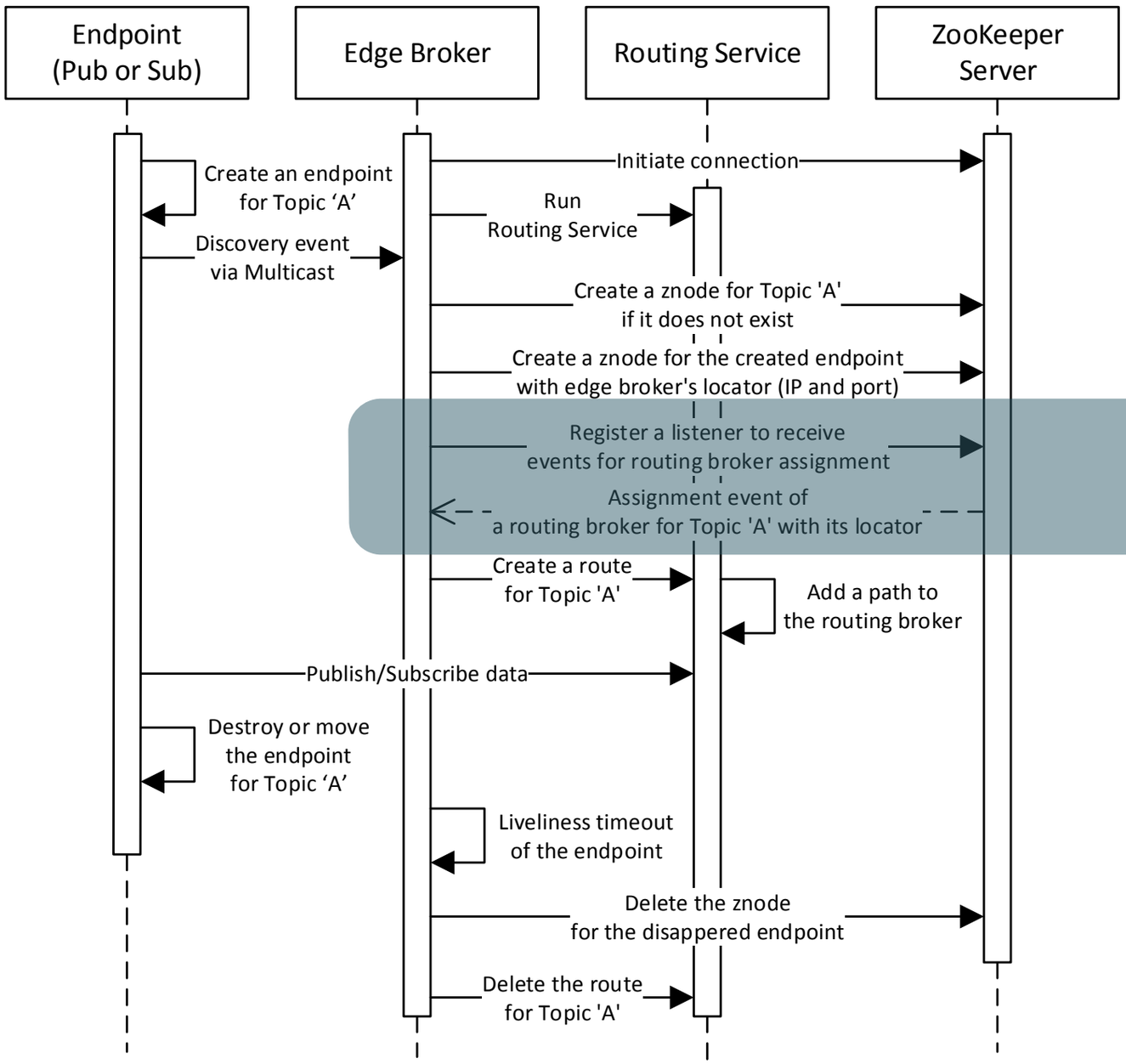
Edge Broker Sequence Diagram

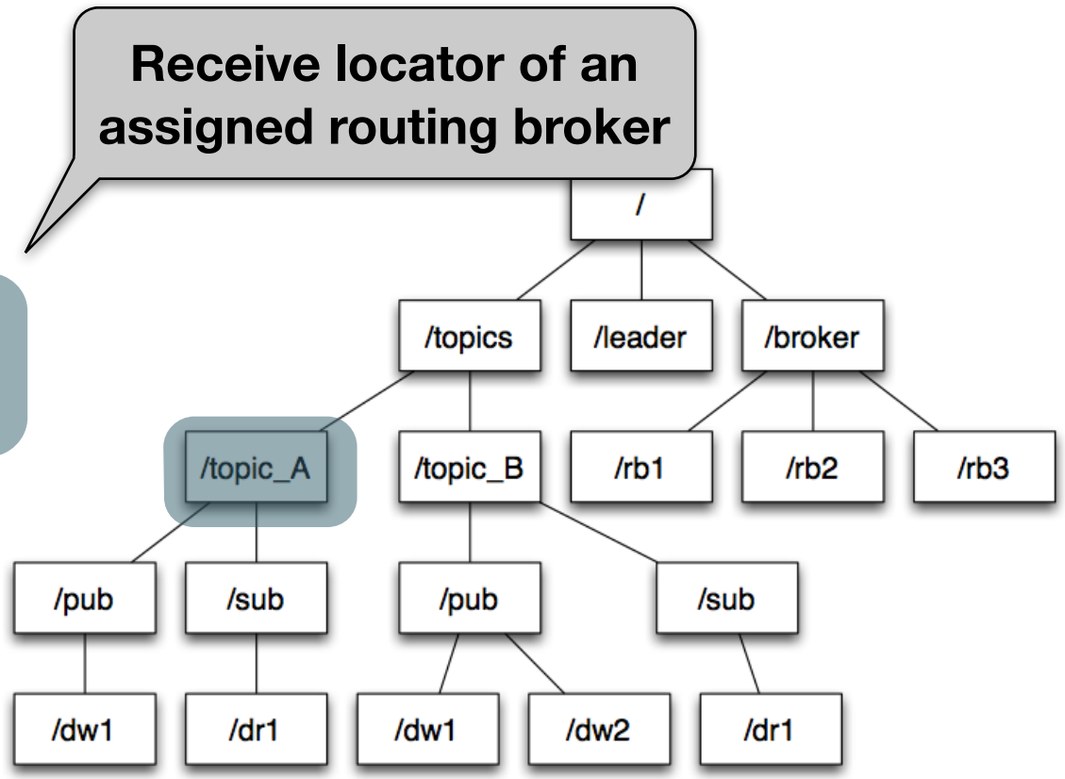


When pub/sub endpoints with Topic ‘A’ are created, create znodes for Topic ‘A’ and endpoints

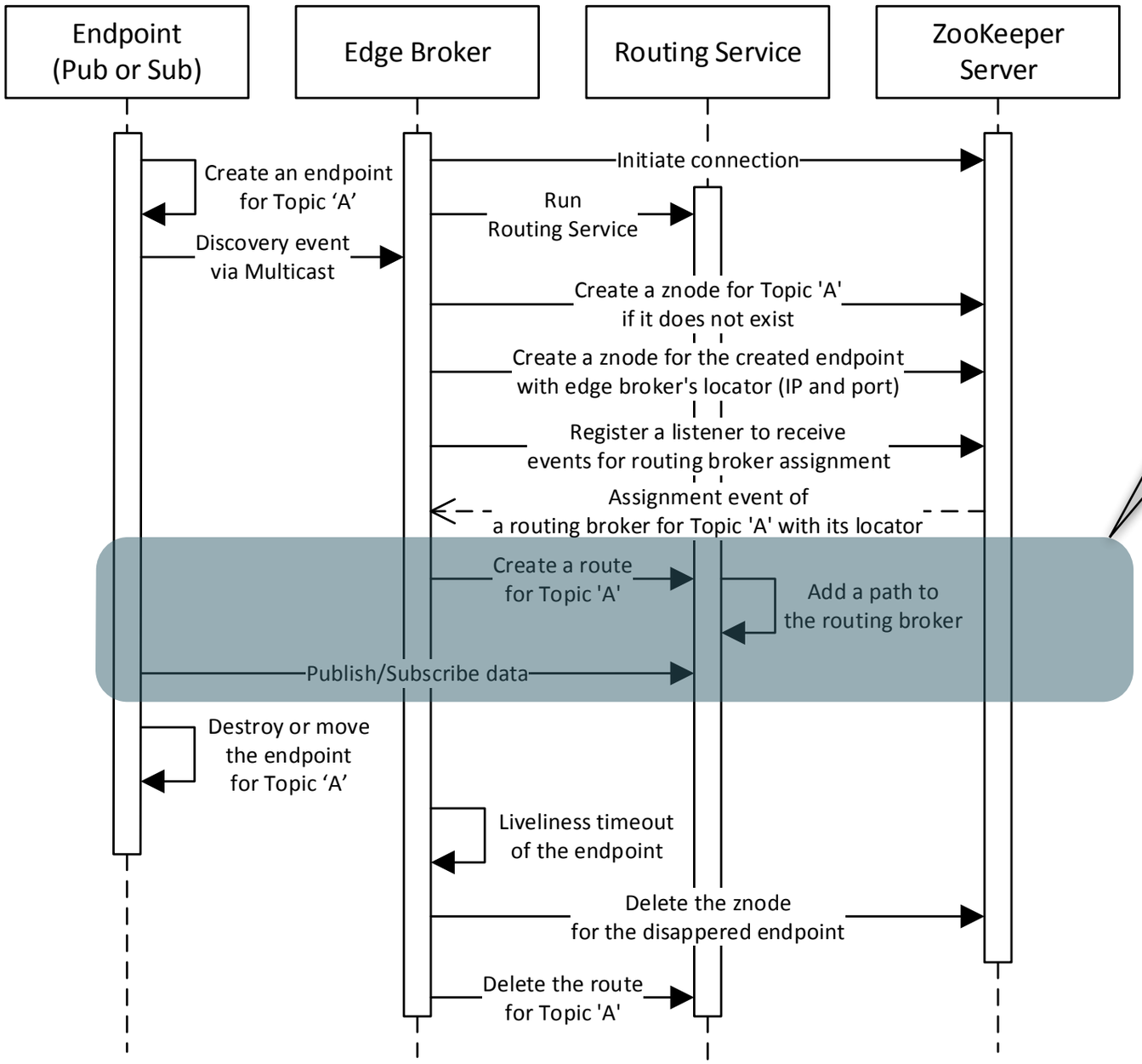


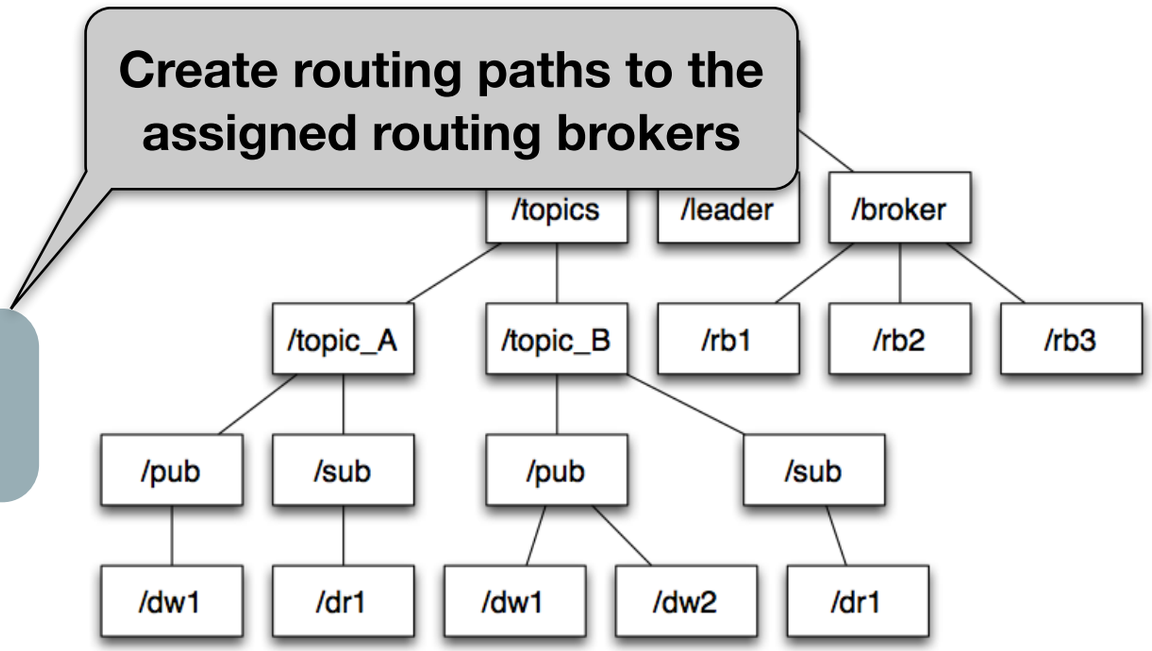
Edge Broker Sequence Diagram



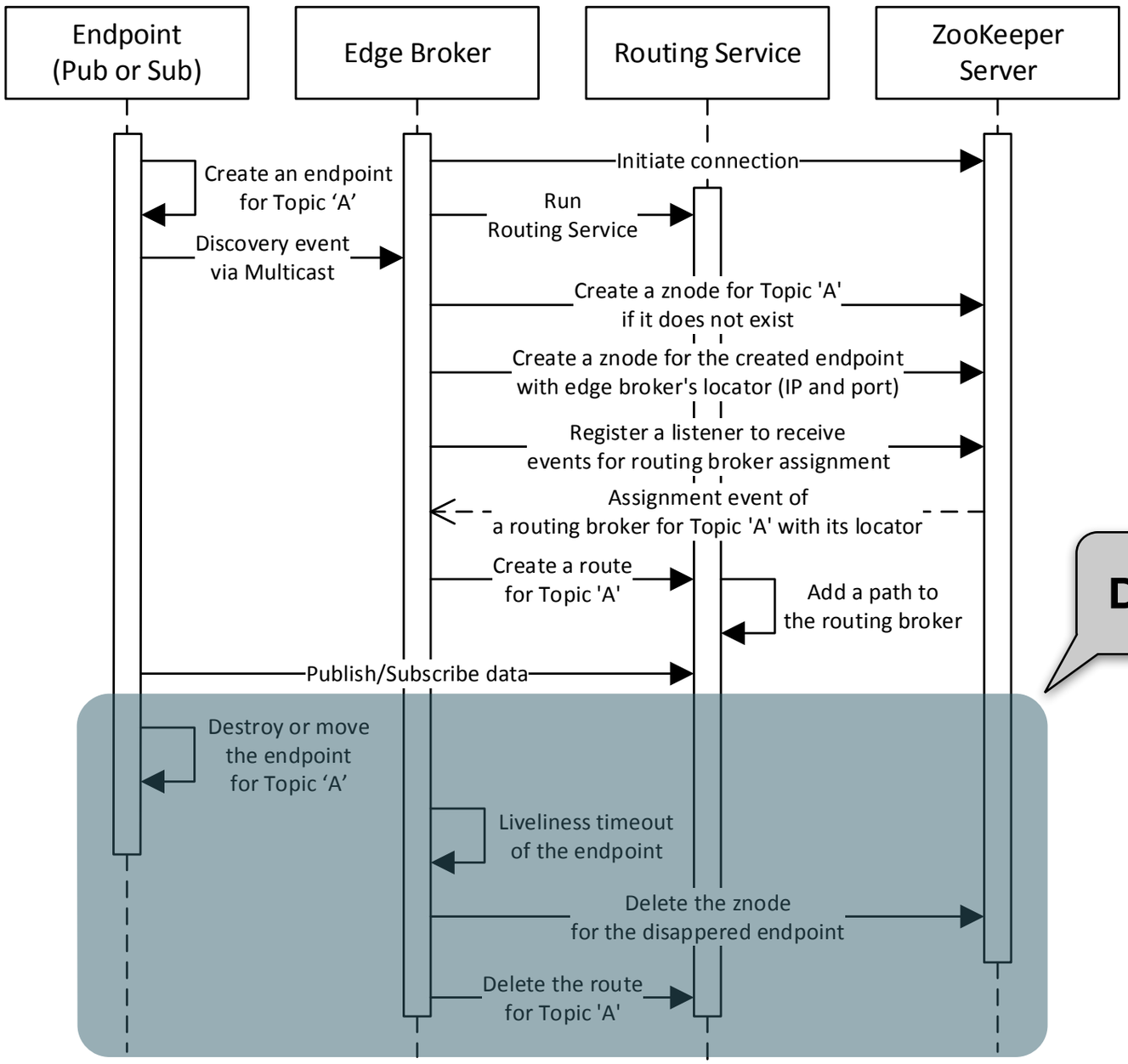


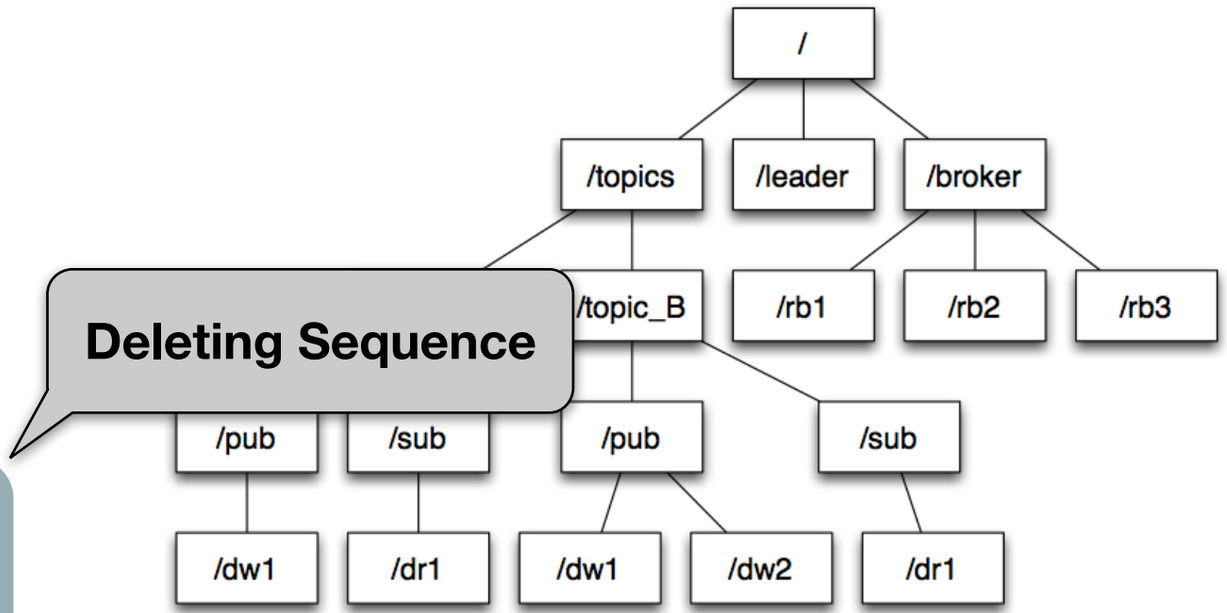
Edge Broker Sequence Diagram





Edge Broker Sequence Diagram





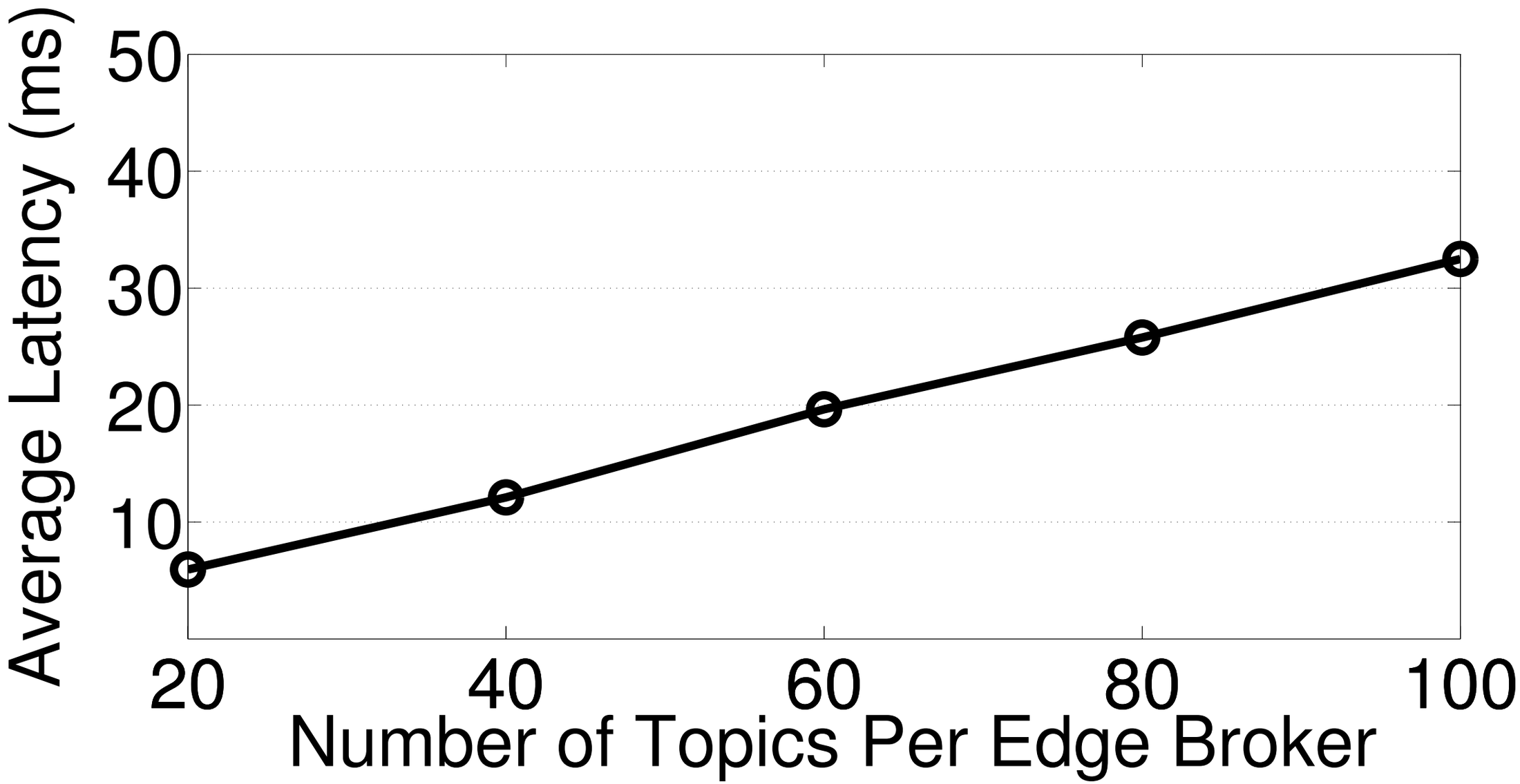
Experimental Results

- **Testbed**
  - **OpenStack-based private cloud comprising 60 machines (Each machine with 12 cores and 32GB of memory)**
  - **To experiment WAN environments, we use Neutron, an OpenStack project for networking as a service, that allows users to create virtual networks by using a Open vSwich plugin**
  - **120 virtual networks**
  - **380 virtual machines (Each VM with 2 vCPU and 2GB of memory)**
  - **RTI Connext 5.1 for Routing Service and test applications**

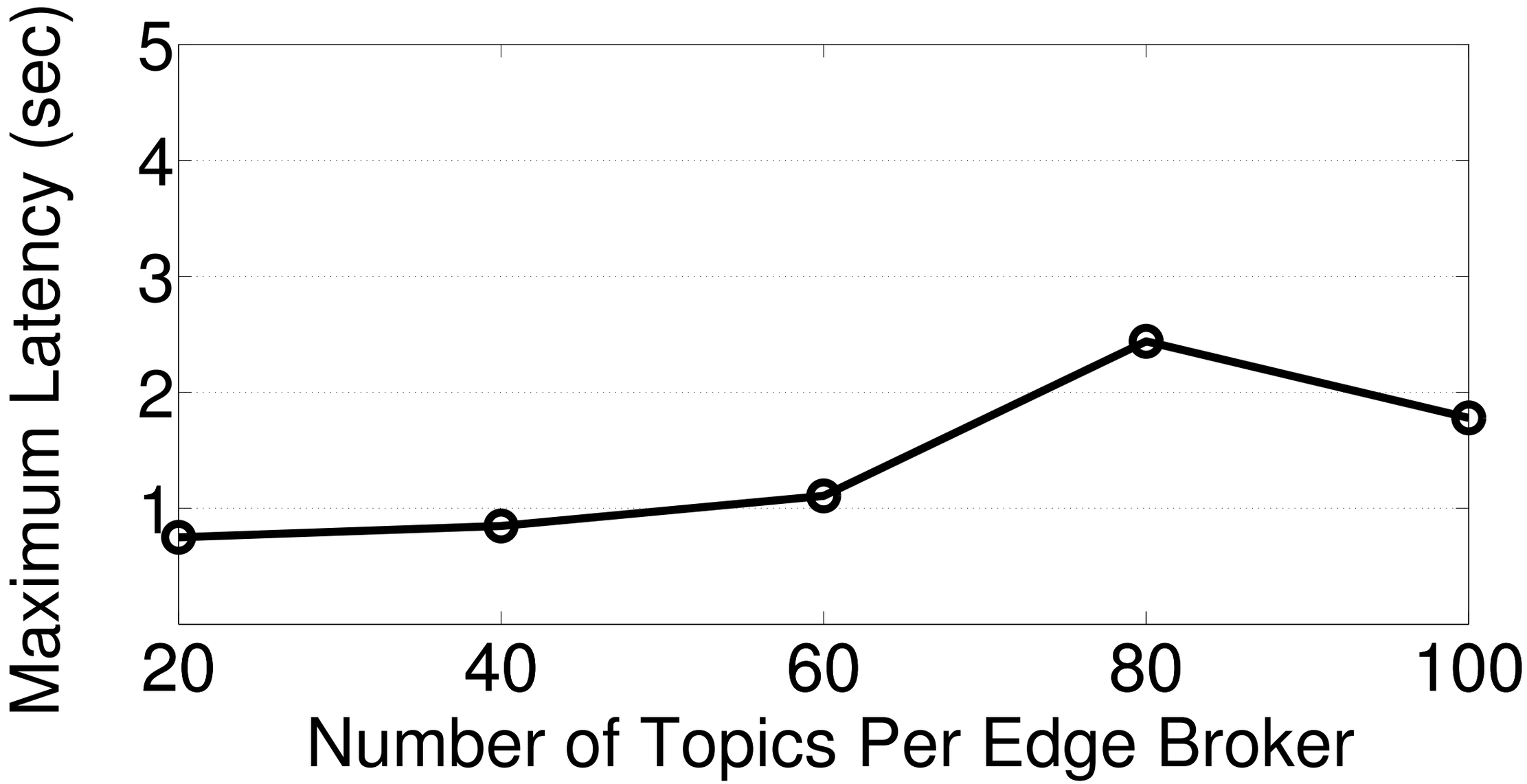
Experimental Results

- Configurations of Test Applications
  - RELIABLE reliability QoS (Reliable data delivery at transport-level)
  - KEEP_ALL history QoS (Keep all data history in memory)
  - TRANSIENT durability QoS (Deliver history data for late joiners)
  - 60 seconds lifespan QoS (Keep data history for 60 seconds)
  - 1,000 of publishers and topics
  - 10,000 of subscribers
  - 64 bytes of data sample size
  - 50 milliseconds of data publishing rate

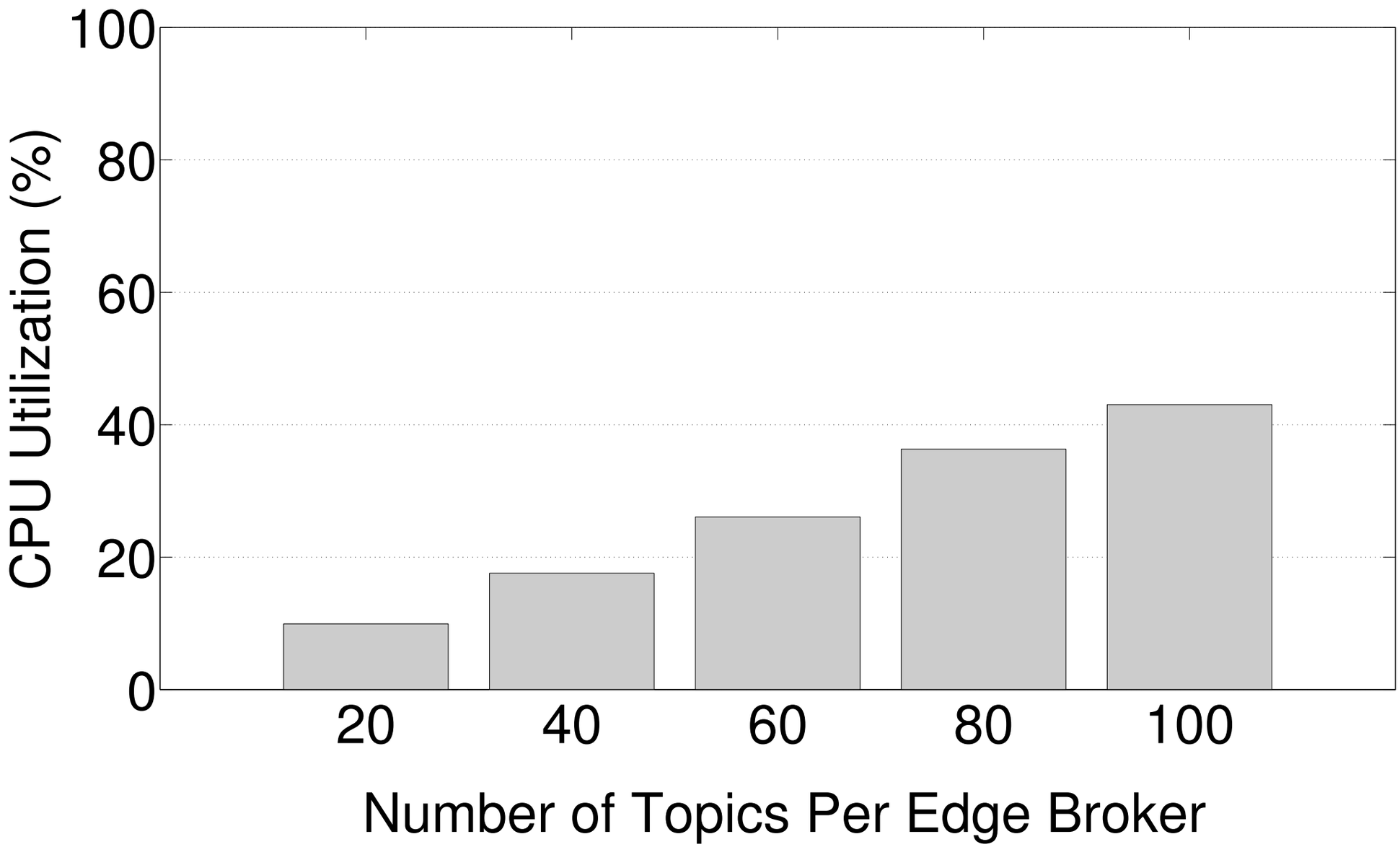
End-to-end Latency of Pub/Sub by Different Number of Topics Per Network



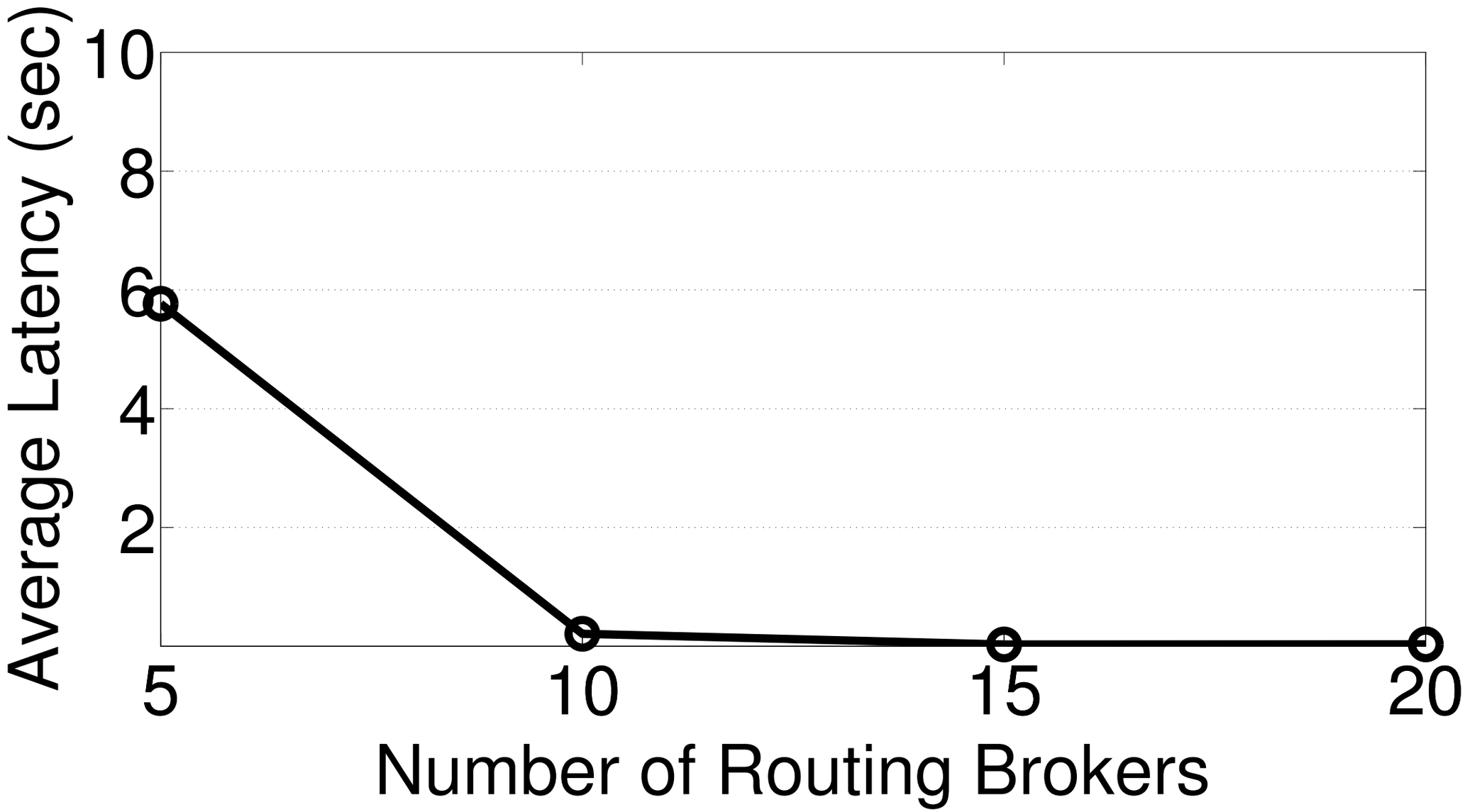
End-to-end Latency of Pub/Sub by Different Number of Topics Per Network



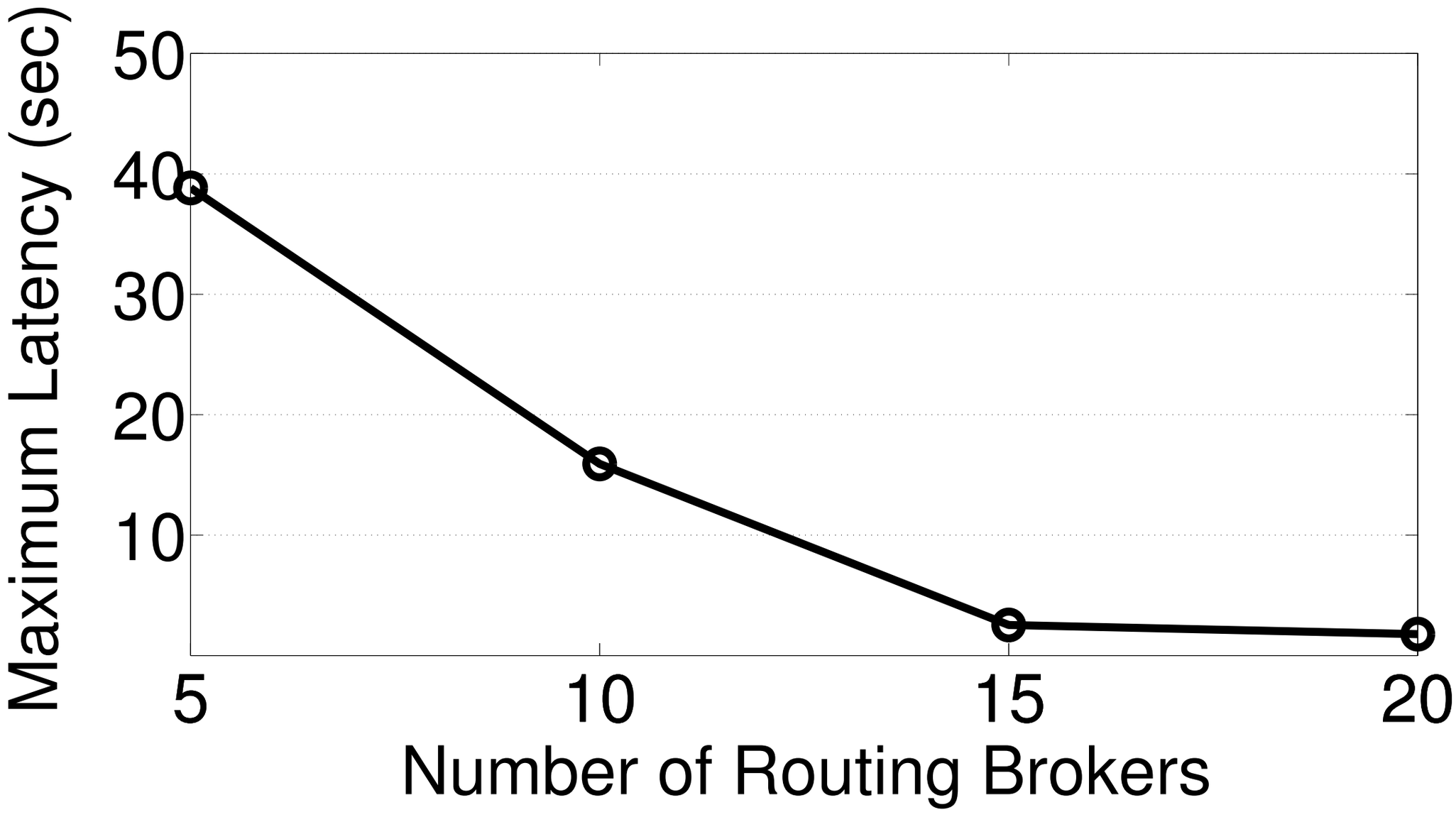
CPU Utilization
by Different Number of Topics Per Network



End-to-end Latency of Pub/Sub with Load Balance in Routing Brokers

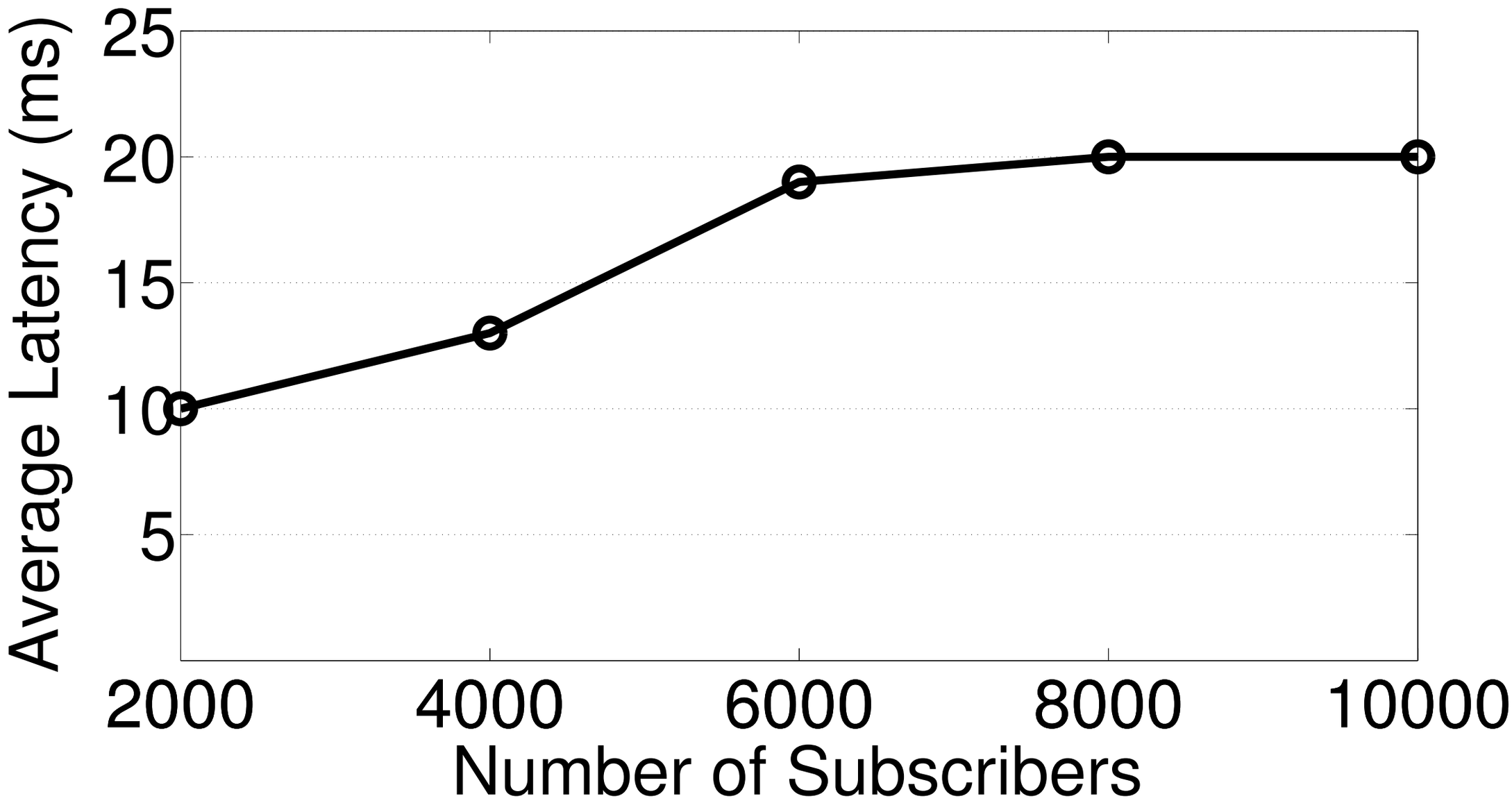


End-to-end Latency of Pub/Sub with Load Balance in Routing Brokers

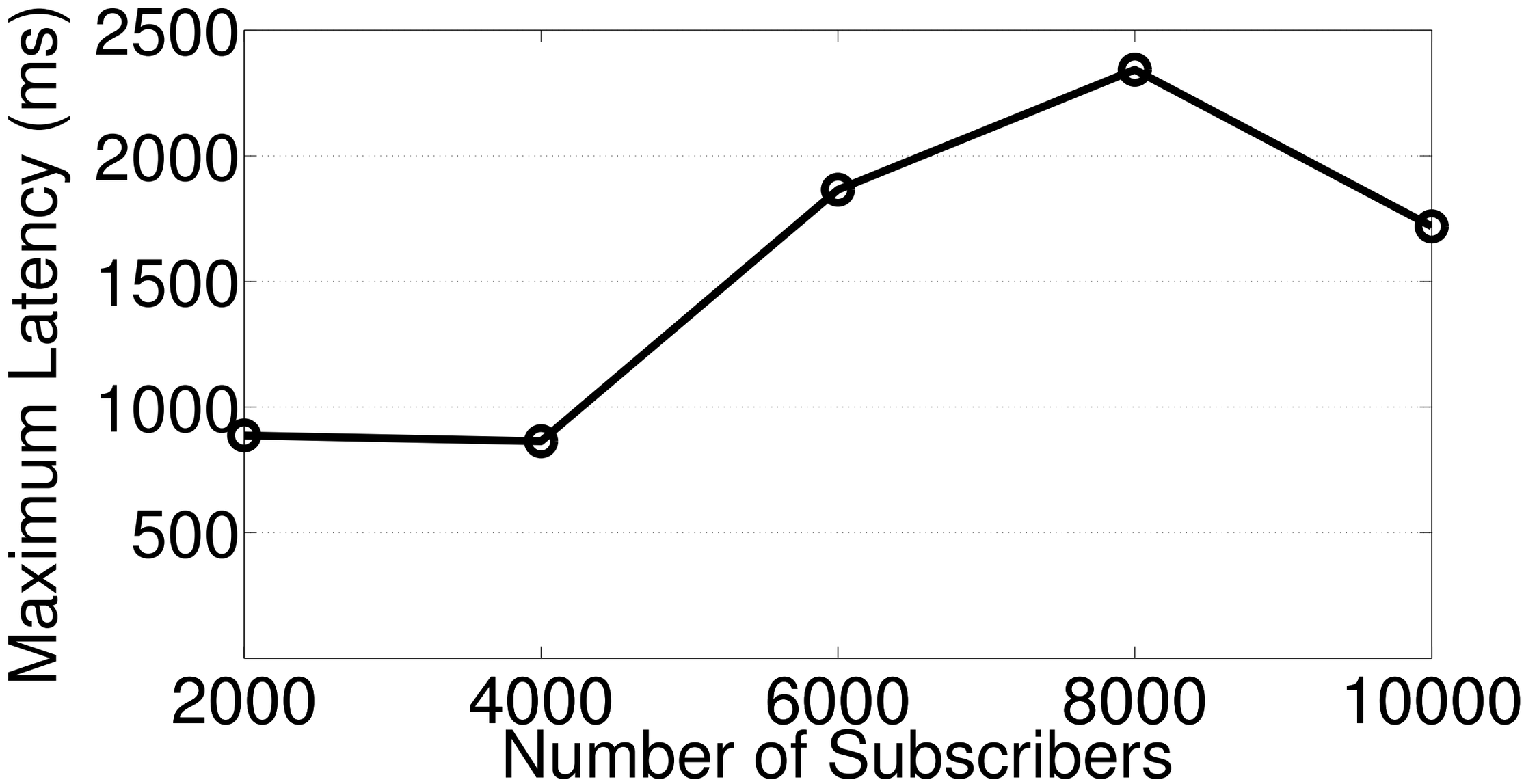


CPU Utilization
with Load Balance in Routing Brokers

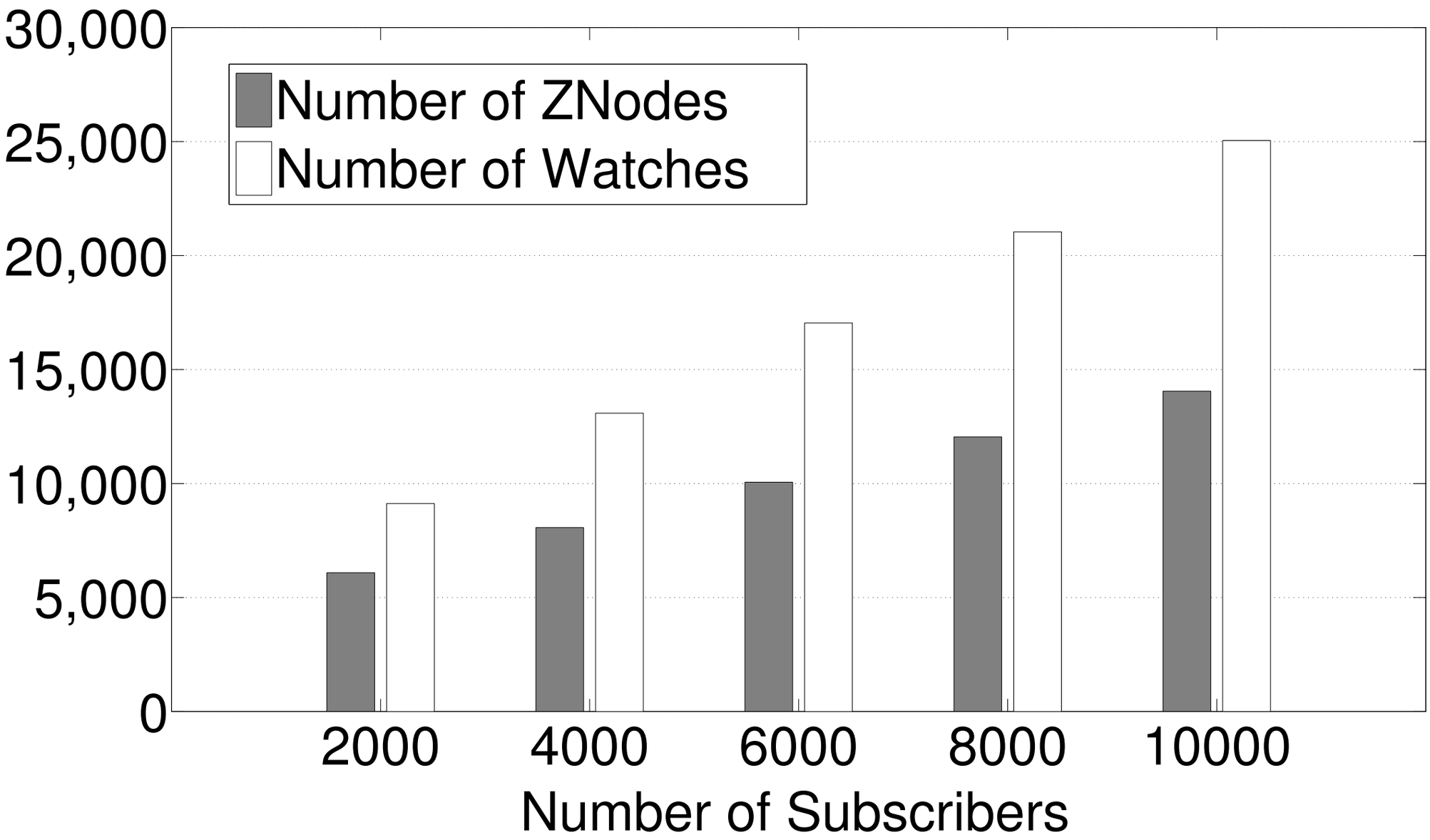
Latency of Coordination Service by Different Number of Joining Subscribers



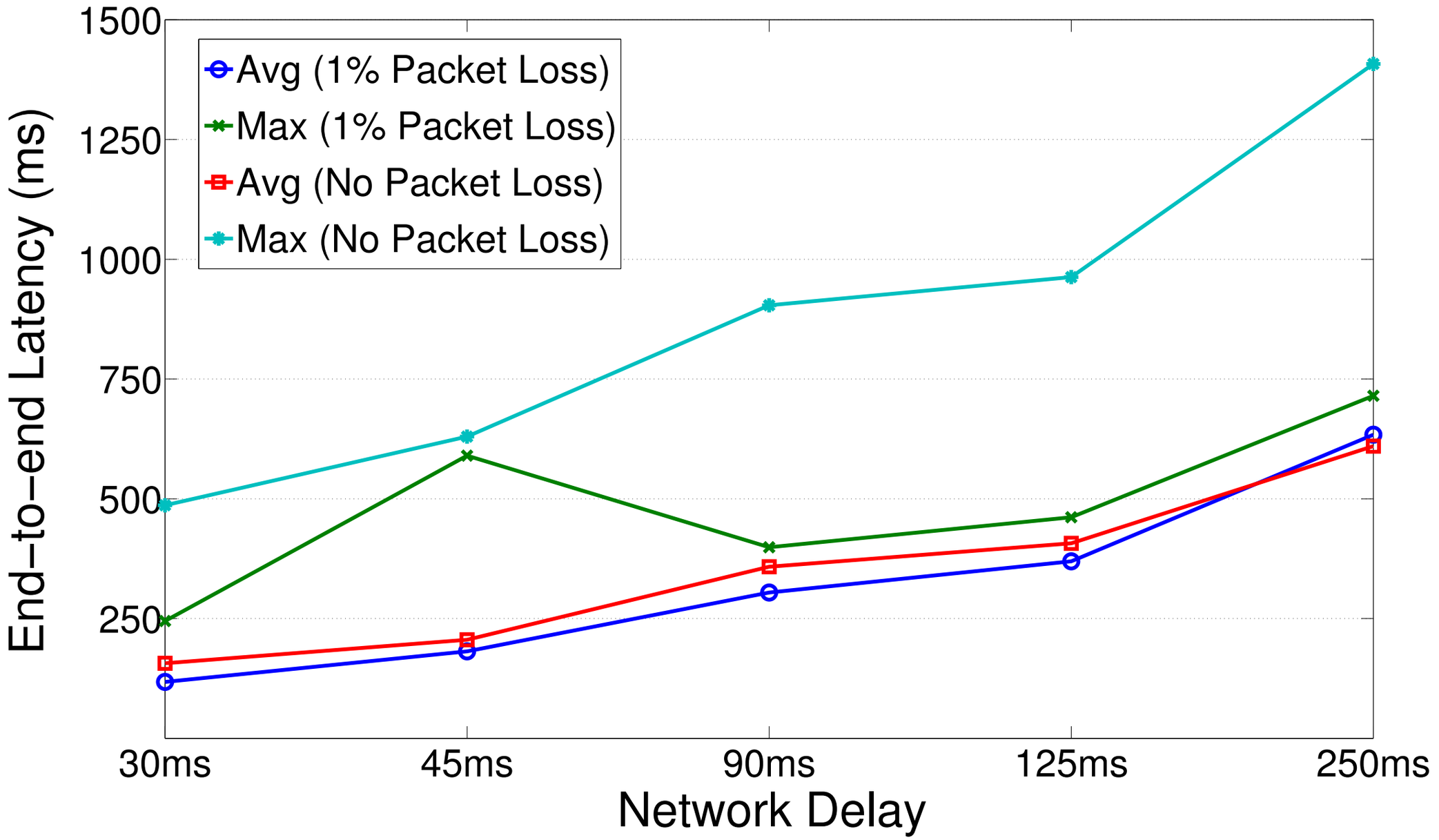
Latency of Coordination Service by Different Number of Joining Subscribers



Number of ZNodes and Watches by Different Number of Joining Subscribers



End-to-end Latency of Pub/Sub with Single-path Overlays



End-to-end Latency of Pub/Sub with Multi-path Overlays

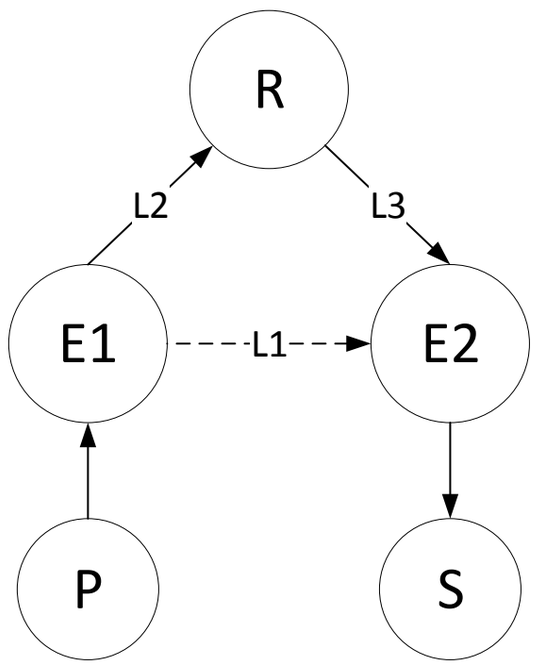
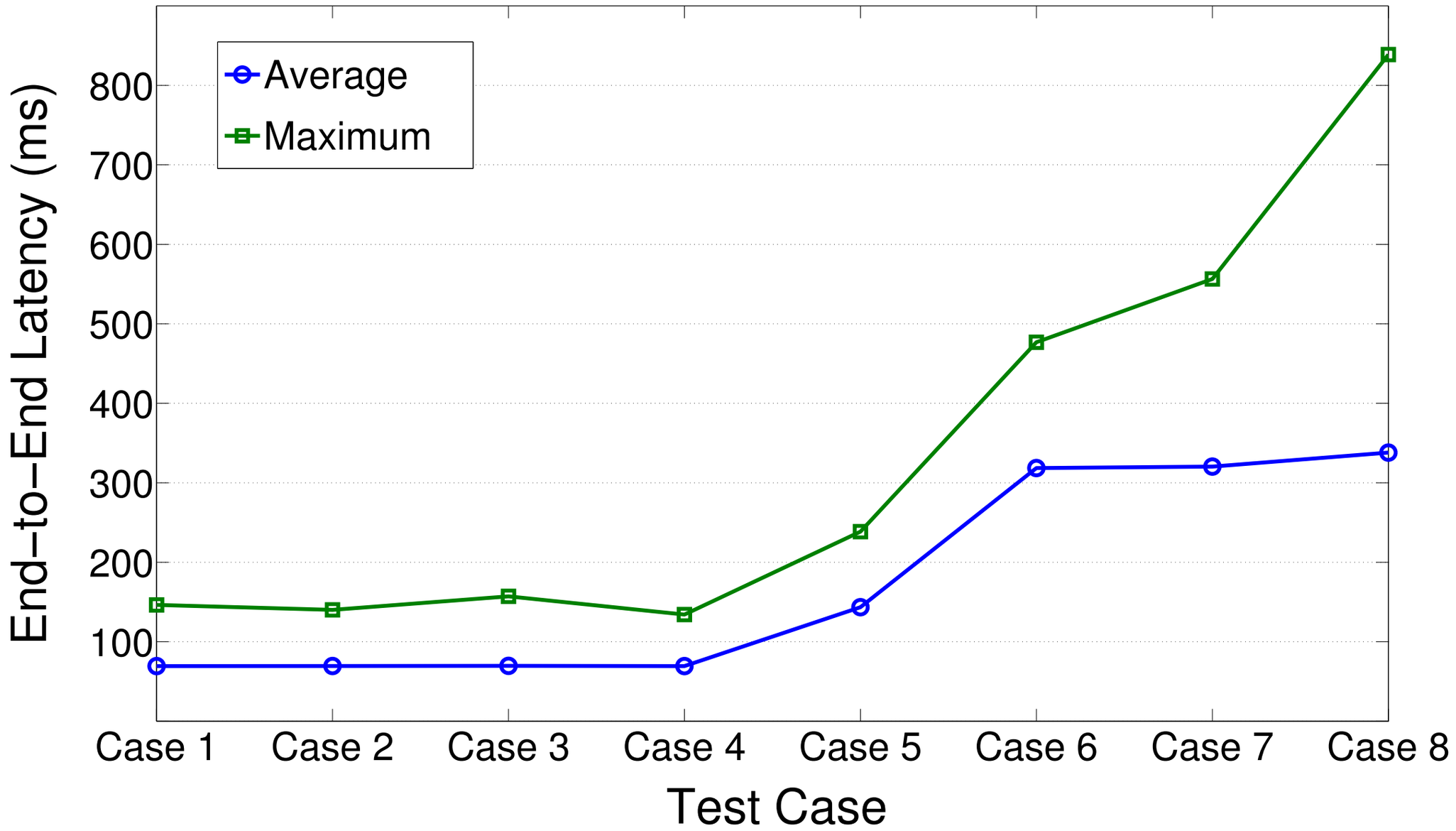


Table I: Deadline-aware Overlays Experiment Cases

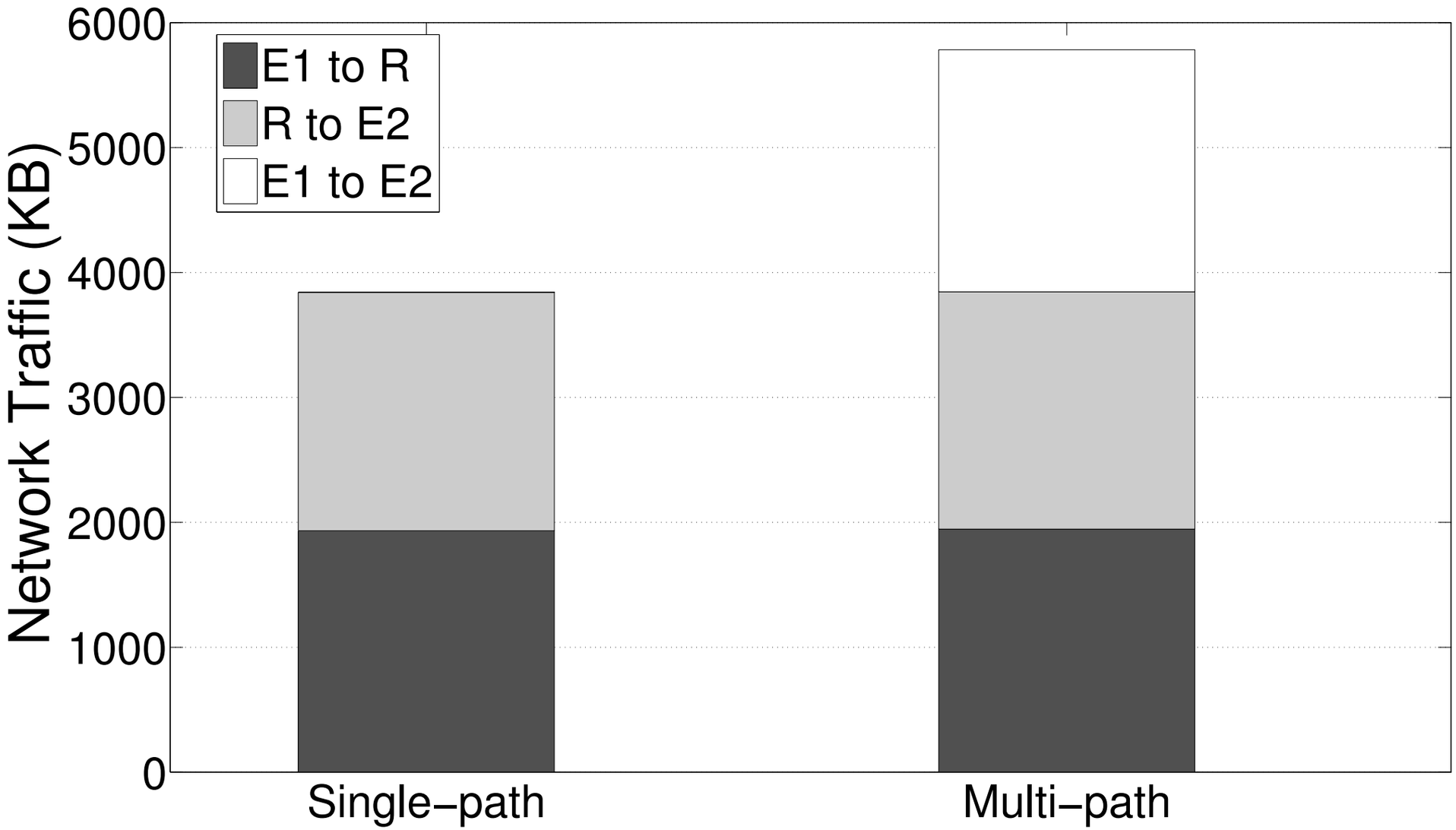
Test Cases	L1	L2	L3
Case 1	A	A	A
Case 2	A	A	B
Case 3	A	B	A
Case 4	A	B	B
Case 5	B	A	A
Case 6	B	A	B
Case 7	B	B	A
Case 8	B	B	B

A = 30ms delay, no packet loss
B = 250ms delay, 1% packet loss

End-to-end Latency of Pub/Sub with Multi-path Overlays



Overhead Comparison



Lessons Learned

- PubSubCoord disseminates data in a scalable and reliable manner for systems having many pub/sub endpoints and topics in WANs

- Centralized coordination service like ZooKeeper can serve as a pub/sub control plane for large-scale systems

- Configurable QoS supported by DDS can be used for low-latency data delivery in WANs by building multi-path overlays

Kyoungho An and Aniruddha Gokhale, *"A Cloud-enabled Coordination Service for Internet-scale OMG DDS Applications"*, Poster paper at the 8th ACM International Conference on Distributed Event-Based Systems (DEBS 2014), Mumbai, India, May 26-29, 2014.

Kyoungho An, Takayuki Kuroda, and Aniruddha Gokhale, **"A Coordination and Discovery Service for QoS-enabled Data-Centric Publish/Subscribe in Wide Area Networks"**, 35th IEEE International Conference on Distributed Computing Systems (ICDCS 2015), Columbus, OH, June 29-July 2, 2015. (In Submission)

Summary:
Doctoral Research Contributions

Focus Area	Challenges	Contributions
Scalability of DDS discovery protocol	Discovery scalability of a number of peers and endpoints in a system	Scalable DDS discovery protocol that utilizes content-based filtering to reduce resource usage during discovery phase
Coordination of DDS brokers in WANs	Data dissemination between peers located in different networks over WANs	Coordination for distributed DDS message brokers to establish scalable and consistent routing data dissemination paths over WANs
Placement of VM replicas	Resource management of VM replicas for highly available cloud enablers	Middleware for cloud infrastructures to guarantee both high availability and low latency through scheduling of VM backups
Testing performance with diverse DDS QoS	Design QoS configurations with considering expected performance impact	Model-based generative framework to run performance test applications with verifying combinations of QoS in the cloud

Future Work

- **Focus Area 1 - CFDP**
  - **Instance-based filtering**
  - **Multi-channel enabled filtering**
- **Focus Area 2 - PubSubCoord**
  - **Fine-grained and automatic load balancing mechanisms**
  - **End-to-end QoS management**

Summary of Publications

Journal Publications

1. Kyoungho An, Shashank Shekhar, Faruk Caglar, Aniruddha Gokhale, and Shivakumar Sastry, A Cloud Middleware for Assuring Performance and High Availability of Soft Real-time Applications, The Elsevier Journal of Systems Architecture (JSA): Embedded Systems Design, 2014.

Book Chapters

2. Kyoungho An, Adam Trewyn, Aniruddha Gokhale and Shivakumar Sastry, Design and Transformation of Domain-specific Language for Reconfigurable Conveyor Systems, Book chapter in Formal and Practical Aspects of Domain-Specific Languages: Recent Developments, IGI Global publishers, Editor: Marjan Mernik, 2012.

Conference & Symposium Publications

3. Kyoungho An, Sumant Tambe, Paul Pazandak, Gerardo Pardo-Castellote, Aniruddha Gokhale, and Douglas Schmidt, Content-based Filtering Discovery Protocol (CFDP): Scalable and Efficient OMG DDS Discovery Protocol, 8th ACM International Conference on Distributed Event-Based Systems (DEBS 2014), Mumbai, India, May 26-29, 2014.

4. Kyoungho An, Takayuki Kuroda, Aniruddha Gokhale, Sumant Tambe, and Andrea Sorbini, Model-driven Generative Framework for Automated DDS Performance Testing in the Cloud, 12th ACM International Conference on Generative Programming: Concepts & Experiences (GPCE 2013), Indianapolis, IN, Oct 27-28, 2013.

First Author

Summary of Publications

5. Kyoungho An, Resource Management and Fault Tolerance Principles for Supporting Distributed Real-time and Embedded Systems in the Cloud, 9th Middleware Doctoral Symposium (MDS 2012), co-located with ACM/IFIP/USENIX 13th International Conference on Middleware (Middleware 2012), Montreal, Quebec, Canada, Dec 3-7, 2012.

6. Kyoungho An, Adam Trewyn, Aniruddha Gokhale and Shivakumar Sastry, Model-driven Performance Analysis of Reconfigurable Conveyor Systems used in Material Handling Applications, Second ACM/IEEE International Conference on Cyber Physical Systems (ICCPS 2011), Chicago, IL, Apr 11-14, 2011.

7. Anushi Shah, Kyoungho An, Aniruddha Gokhale and Jules White, Maximizing Service Uptime of Smartphone-based Distributed Real-time and Embedded Systems, 14th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing (ISORC 2011), Newport Beach, CA, Mar 28-31, 2011.

Workshop, Work in Progress, and Poster Publications

8. Kyoungho An and Aniruddha Gokhale, A Cloud-enabled Coordination Service for Internet-scale OMG DDS Applications, Poster paper at the 8th ACM International Conference on Distributed Event-Based Systems (DEBS 2014), Mumbai, India, May 26-29, 2014.

9. Shashank Shekhar, Faruk Caglar, Kyoungho An, Takayuki Kuroda, Aniruddha Gokhale and Swapna Gokhale, A Model-driven Approach for Price/Performance Tradeoffs in Cloud-based MapReduce Application Deployment, MODELS 2013 workshop on Model-Driven Engineering for High Performance and CLoud computing (MDHPCL 2013), Miami, FL, Sep 29, 2013.

Summary of Publications

10. Kyoungho An and Aniruddha Gokhale, Model-driven Performance Analysis and Deployment Planning for Real-time Stream Processing, Work-in-Progress (WiP) session at 19th IEEE Real-time and Embedded Technology and Applications Symposium (RTAS 2013), Philadelphia PA, Apr 9-11, 2013.

11. Faruk Caglar, Shashank Shekhar, Kyoungho An and Aniruddha Gokhale, WiP Abstract: Intelligent Power- and Performance-aware Tradeoffs for Multicore Servers in Cloud Data Centers, Work-in-Progress (WiP) session at 4th ACM/IEEE International Conference on Cyber Physical Systems (ICCPS 2013), Philadelphia PA, Apr 9-11, 2013.

12. Kyoungho An, Faruk Caglar, Shashank Shekhar and Aniruddha Gokhale, A Framework for Effective Placement of Virtual Machine Replicas for Highly Available Performance-sensitive Cloud-based Applications, RTSS 2012 workshop on Real-time and Distributed Computing in Emerging Applications (REACTION 2012), San Juan, Puerto Rico, Dec 4-7, 2012.

13. Kyoungho An, Subhav Pradhan, Faruk Caglar and Aniruddha Gokhale, A Publish/Subscribe Middleware for Dependable and Real-time Resource Monitoring in the Cloud, Middleware 2012 workshop on Secure and Dependable Middleware for Cloud Monitoring and Management (SDMCMM 2012), Montreal, Quebec, Canada, Dec 3-7, 2012.

14. Kyoungho An, Strategies for Reliable, Cloud-based Distributed Real-time and Embedded Systems, Extended abstract for PhD Forum in 31st IEEE International Symposium on Reliable Distributed Systems (SRDS 2012), Irvine, CA, Oct 8-11, 2012.

Summary of Publications

15. Faruk Caglar, Kyoungho An, Aniruddha Gokhale and Tihamer Levendovszky, Transitioning to the Cloud? A Model-driven Analysis and Automated Deployment Capability for Cloud Services, MODELS 2012 workshop on Model-Driven Engineering for High Performance and CLoud computing (MDHPCL 2012), Innsbruck, Austria, Sep 30 - Oct 5, 2012.

Technical Reports

16. Shweta Khare, Sumant Tambe, Kyoungho An, Aniruddha Gokhale, and Paul Pazandak, Scalable Reactive Stream Processing Using DDS and Rx: An Industry-Academia Collaborative Research Experience, ISIS Technical Report, no. ISIS-14-103: Institute for Software Integrated Systems, Vanderbilt University, Nashville TN, April, 2014.

17. Kyoungho An, Sumant Tambe, Andrea Sorbini, Sheeladitya Mukherjee, Javier Povedano-Molina, Michael Walker, Nirjhar Vermani, Aniruddha Gokhale, and Paul Pazandak, Real-time Sensor Data Analysis Processing of a Soccer Game Using OMG DDS Publish/Subscribe Middleware, ISIS Technical Report, no. ISIS-13-102: Institute for Software Integrated Systems, Vanderbilt University, Nashville TN, June, 2013.

Summary of Publications

Submitted Papers

18. Kyoungho An, Takayuki Kuroda, and Aniruddha Gokhale, A Coordination and Discovery Service for QoS-enabled Data-Centric Publish/Subscribe in Wide Area Networks, 35th IEEE International Conference on Distributed Computing Systems (ICDCS 2015), Columbus, OH, June 29-July 2, 2015.

19. Shweta Khare, Kyoungho An, Aniruddha Gokhale, and Sumant Tambe, Functional Reactive Stream Processing for Data-centric Publish/Subscribe: Experiences using .NET Reactive Extensions with OMG Data Distribution Service, 9th ACM International Conference on Distributed Event-Based Systems (DEBS 2015), Oslo, Norway, June 29-July 3, 2015.

Thank you! Any Questions?