

A Hierarchical Mechanism for the  
**Scalable Deployment of Services**  
over Large Programmable and  
Heterogeneous Networks

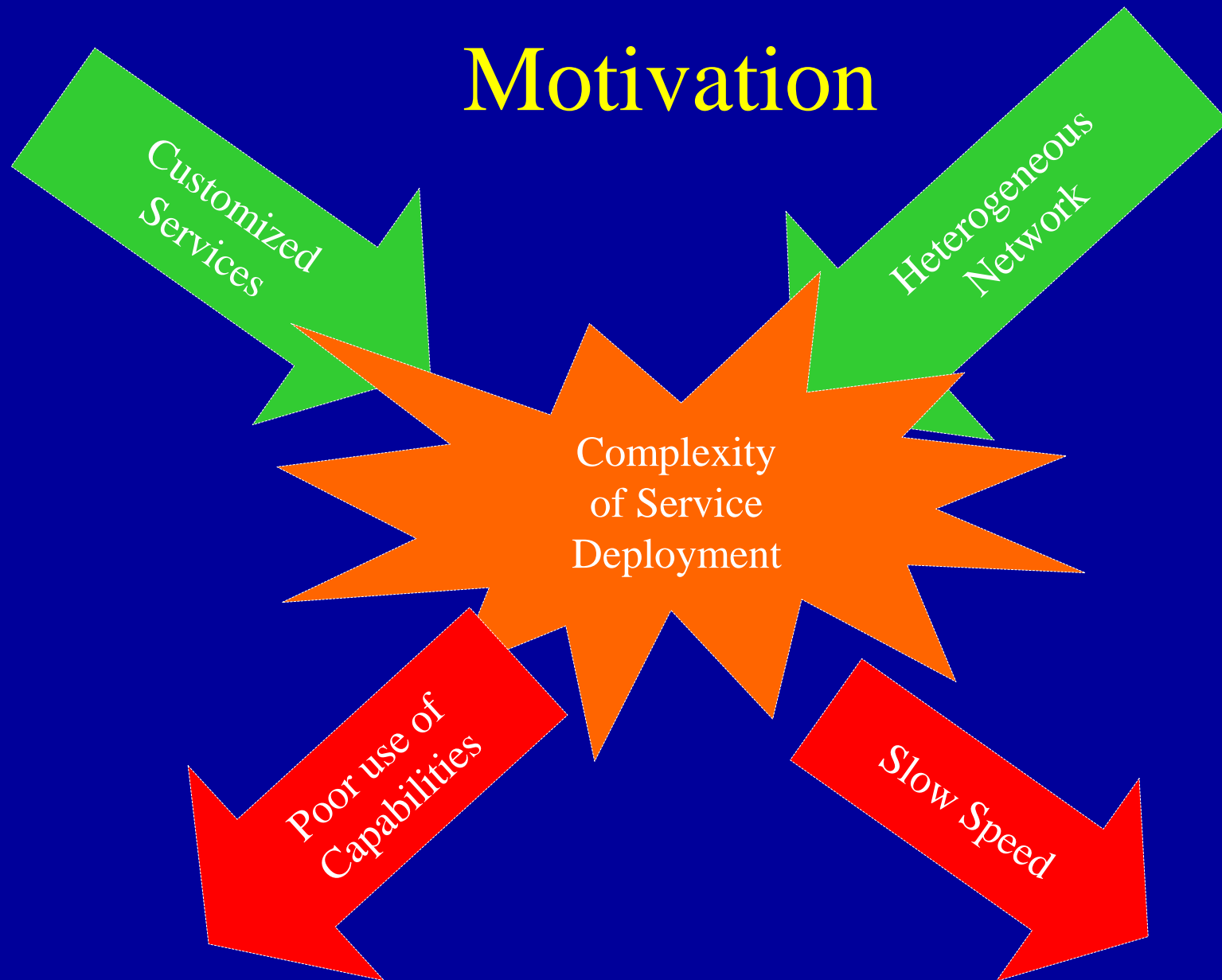
**Robert Haas\***, Patrick Droz\*, Burkhard Stiller\*\*,

\*IBM Zurich Research Lab,

\*\*Computer Engineering and Networks Lab, ETHZ.

ICC2001, Helsinki, June 2001

# Motivation



# Overview

- ➔ Provide automated service deployment
  - make "best use" of capabilities of the infrastructure
  - automatically organize installation and configuration
- Hierarchical deployment:
  - aggregated views of network capabilities
  - necessary for large networks
- Deployment is service-generic:

Basic Information Types
Topology
Capabilities
Performance
cost

Service Deployment Policy
Path-based
Node-based
Path- and Node-based

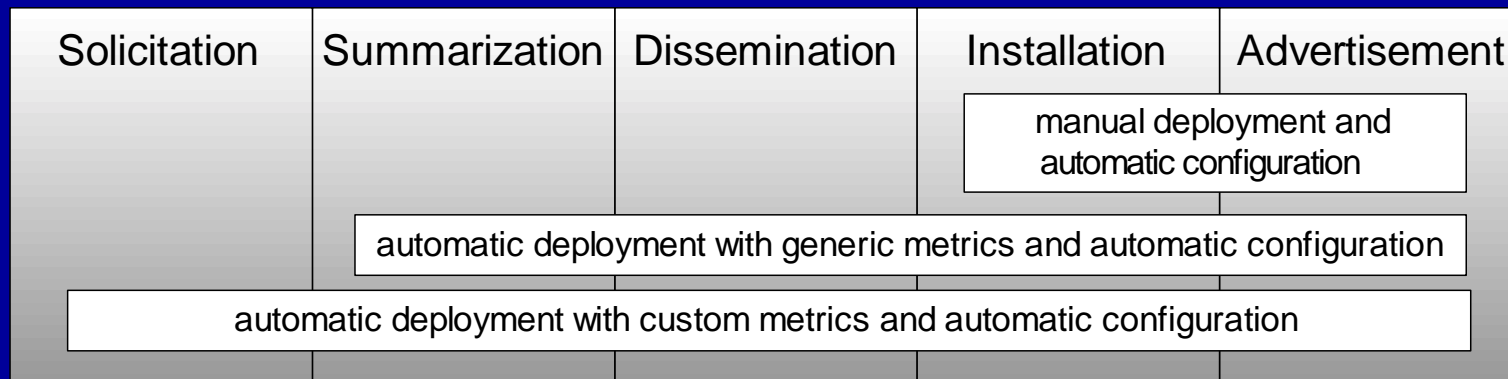
# Related Work

- in active networks
  - service deployed along the path taken by the capsule
  - service destined for that session
- in mobile agents
  - navigation model
- in programmable networks
  - service loaded on-demand, manual coordination
- in IN (Intelligent Network)
  - service deployed at a central point (SCP)

# Procedure Overview

- Solicitation
  - check if requirements are met for the service to be installed
- Summarization
  - suitable nodes advertise their support
  - in an aggregated format
- Dissemination
  - service is deployed on a subset of the suitable nodes
- Installed-services advertisement
  - to enable service-aware routing

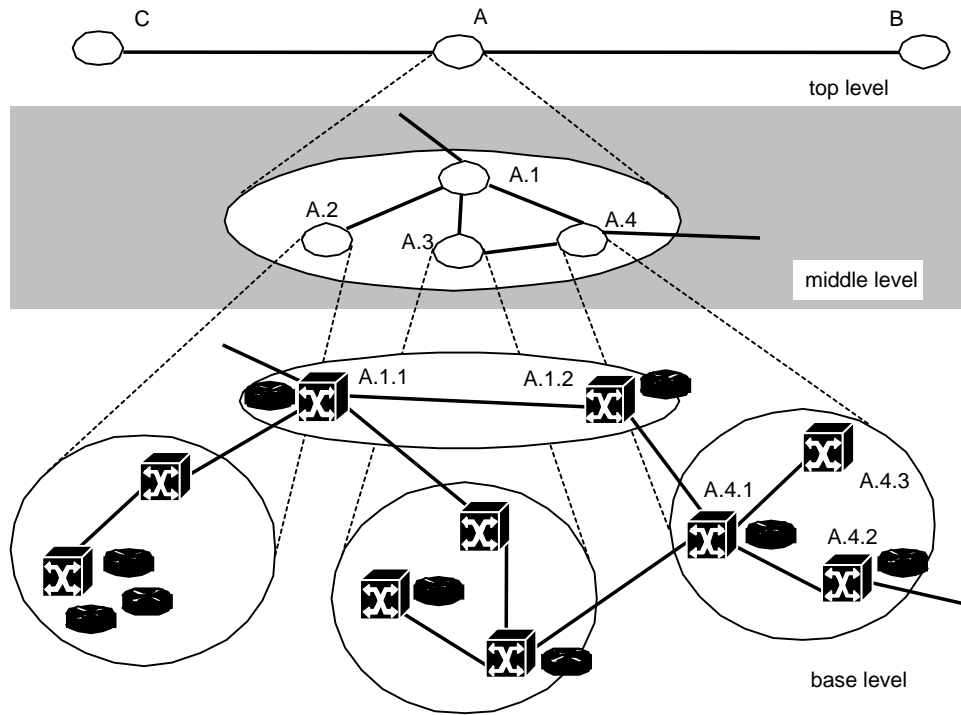
# 5-steps Procedure



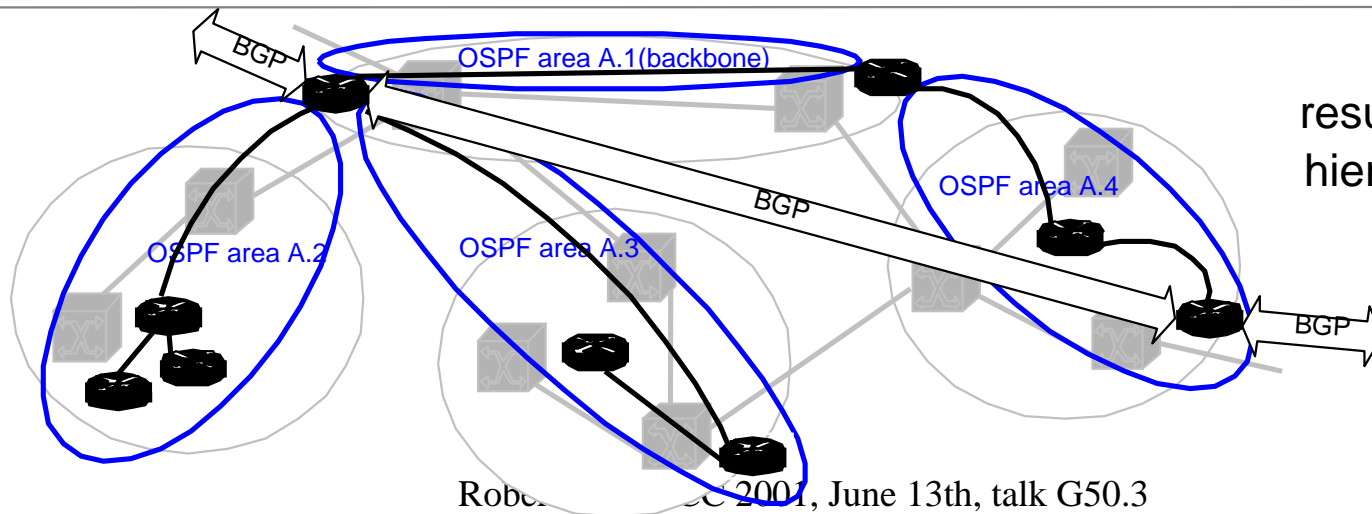
# Examples of Services

	Deployment	Addressing	Signaling
Hierarchical IP Routing	Node-based	Explicit	Out-of-band
Transparent Multi-layer Proxy-Caching	Node-based	Implicit (for clients) and Explicit	In-band
VPN	Path- and Node-based	Implicit (for RSVP) and Explicit (for endpoints)	Out-of-band
Diff-Serv++	Path-based	Implicit	In-band

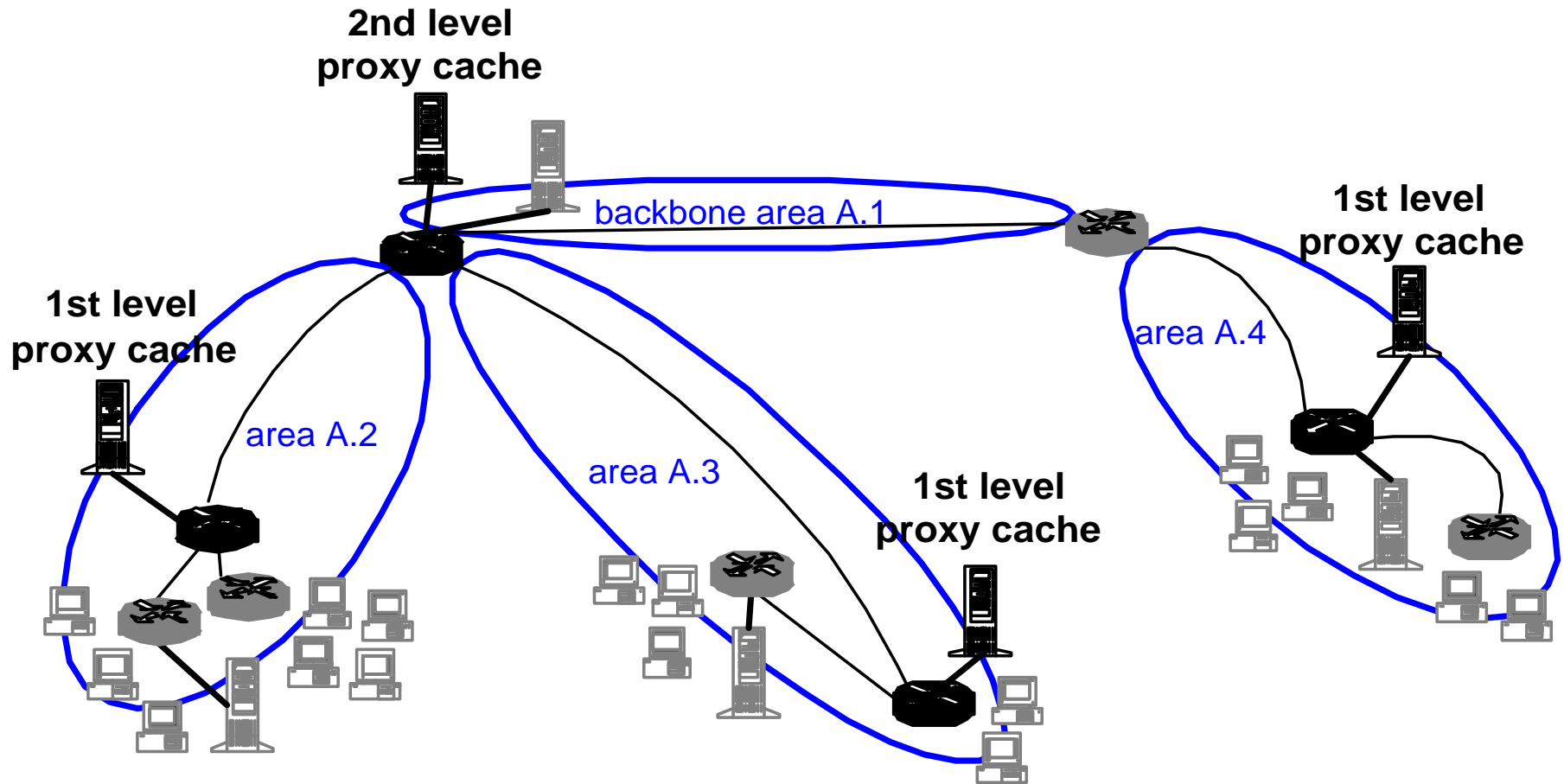
# Hierarchical IP Routing



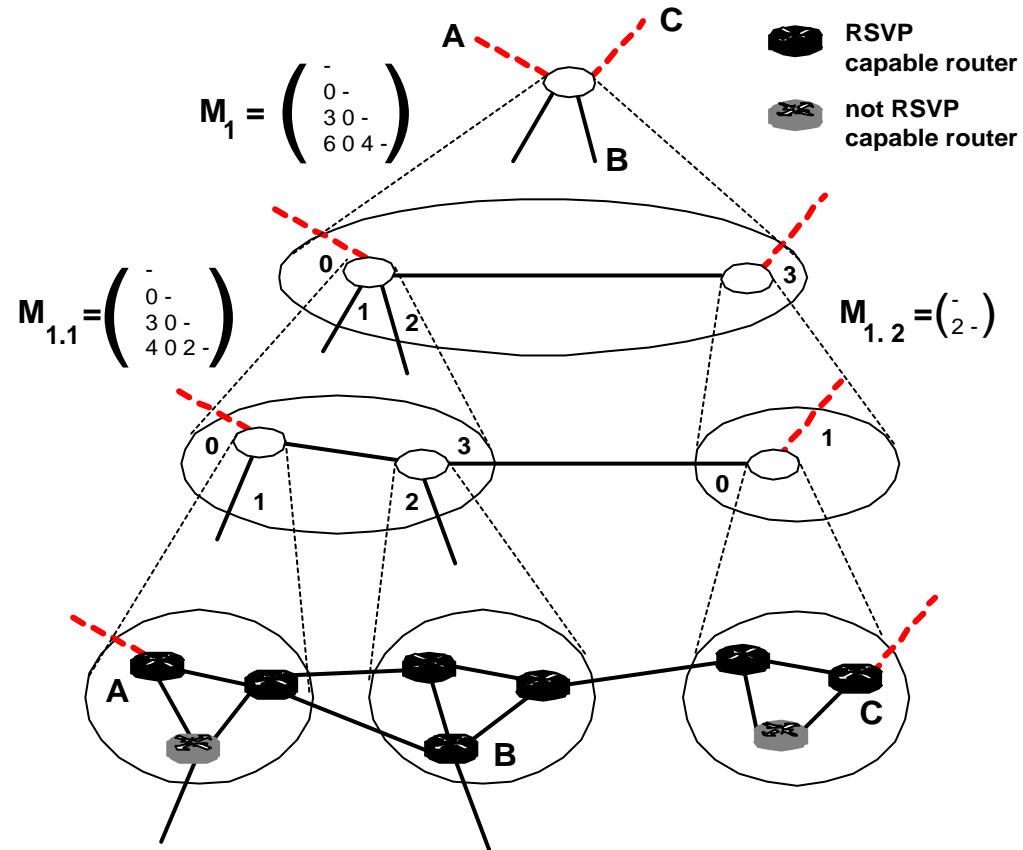
PNNI hierarchy  
(= service hierarchy)



# Two-level Transparent Proxy-Cache

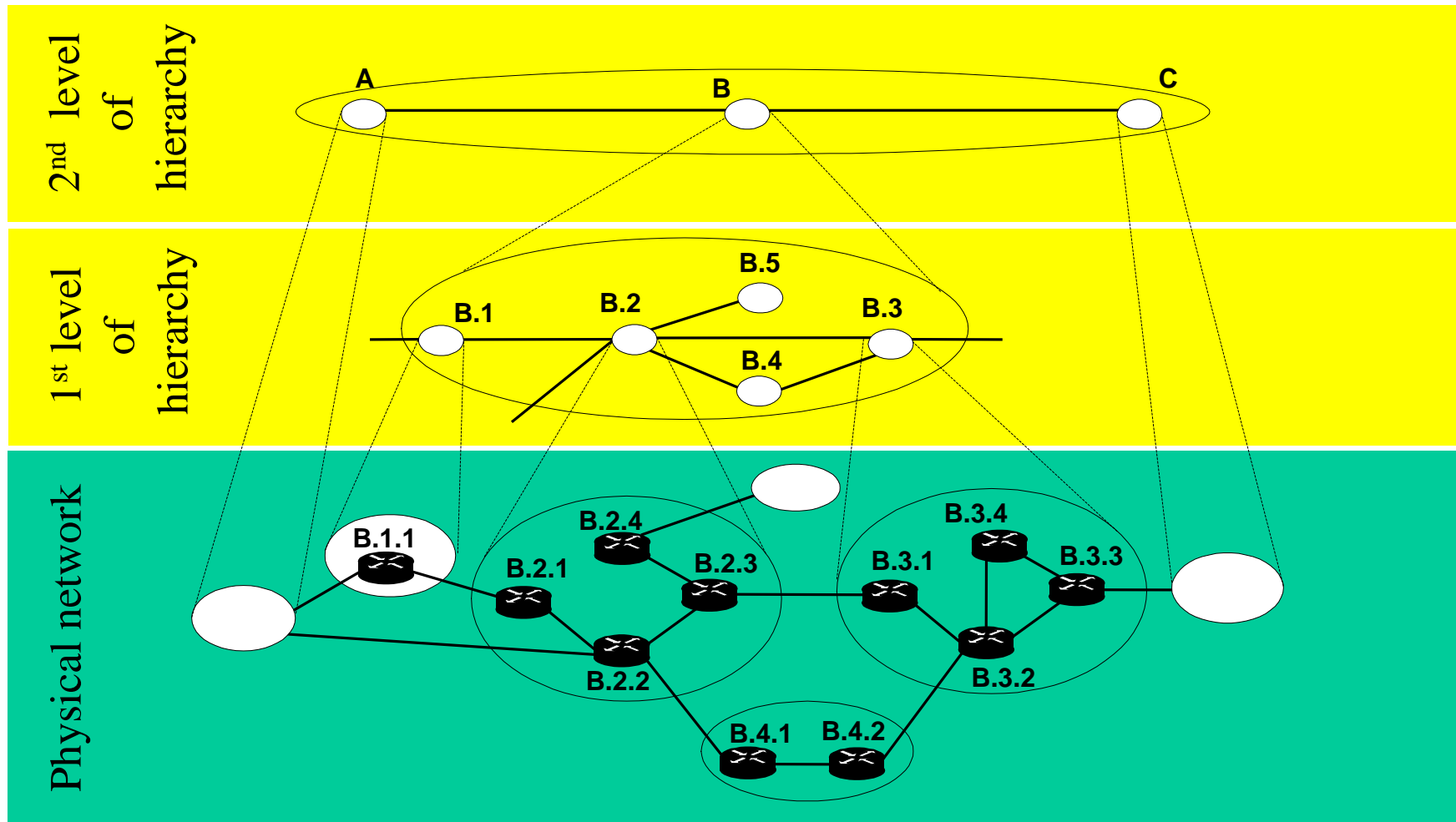


# Virtual Private Network



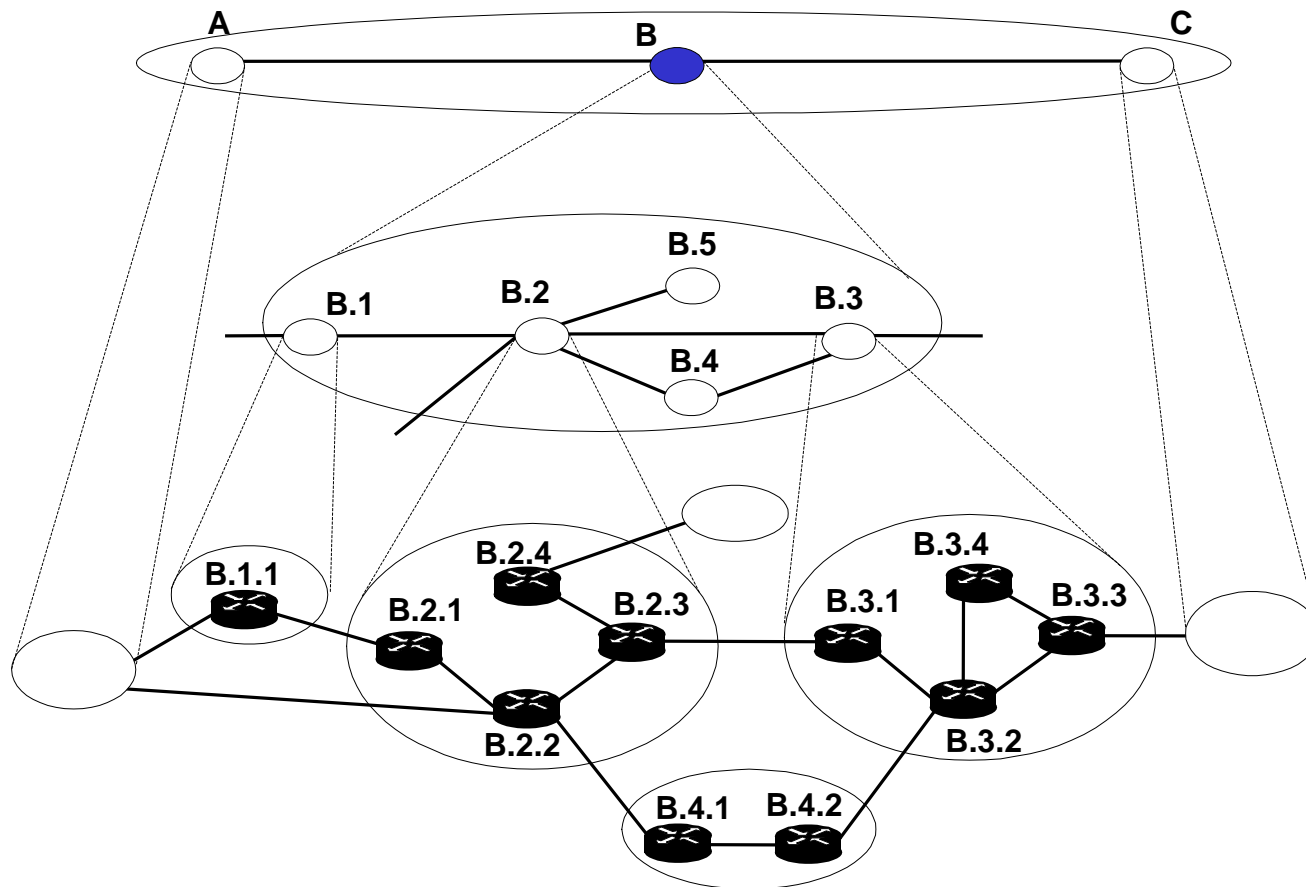
Solicitation ←  
 Summarization  
 Dissemination  
 Installation  
 Advertisement

# Diff-Serv++



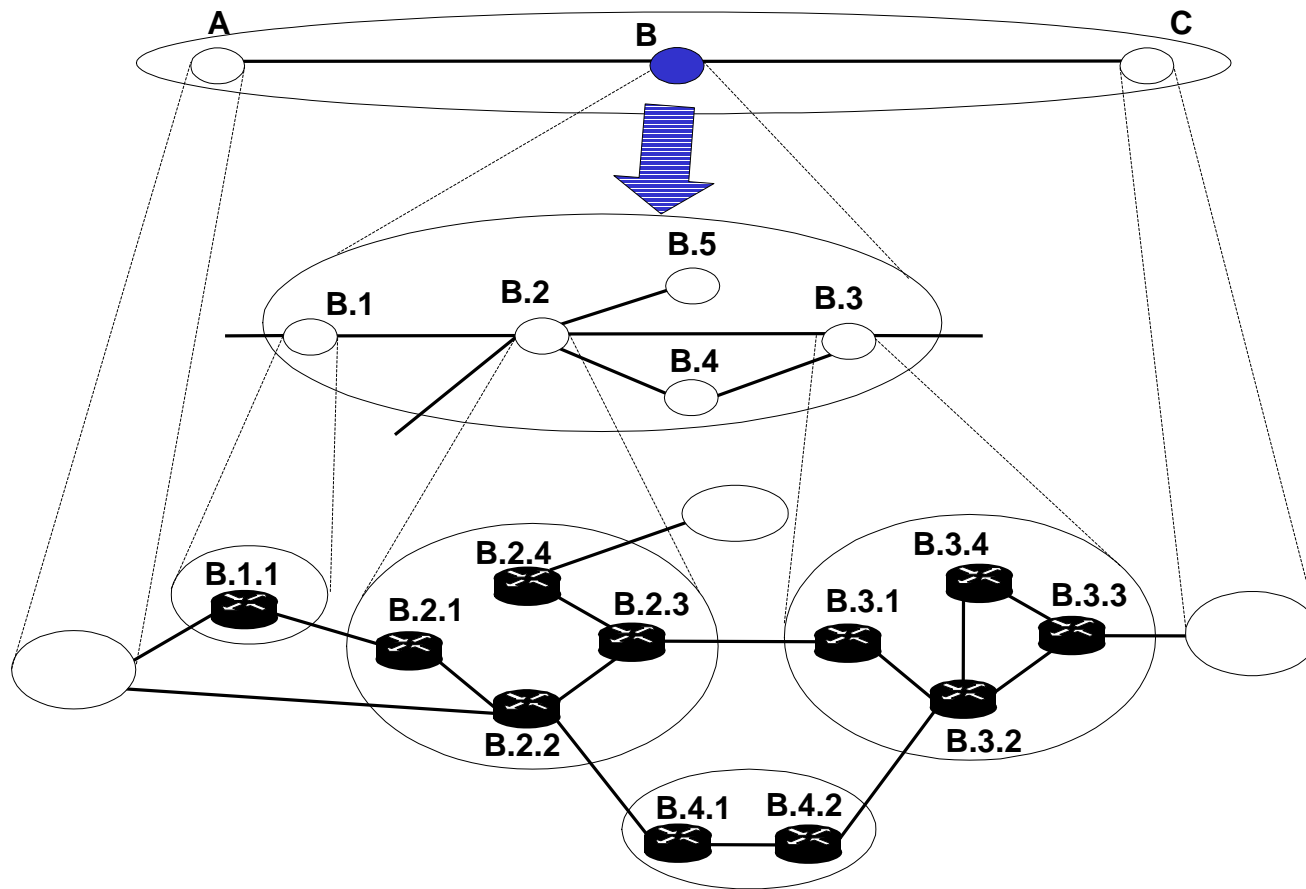
Solicitation ←  
Summarization  
Dissemination  
Installation  
Advertisement

# Solicitation (1)



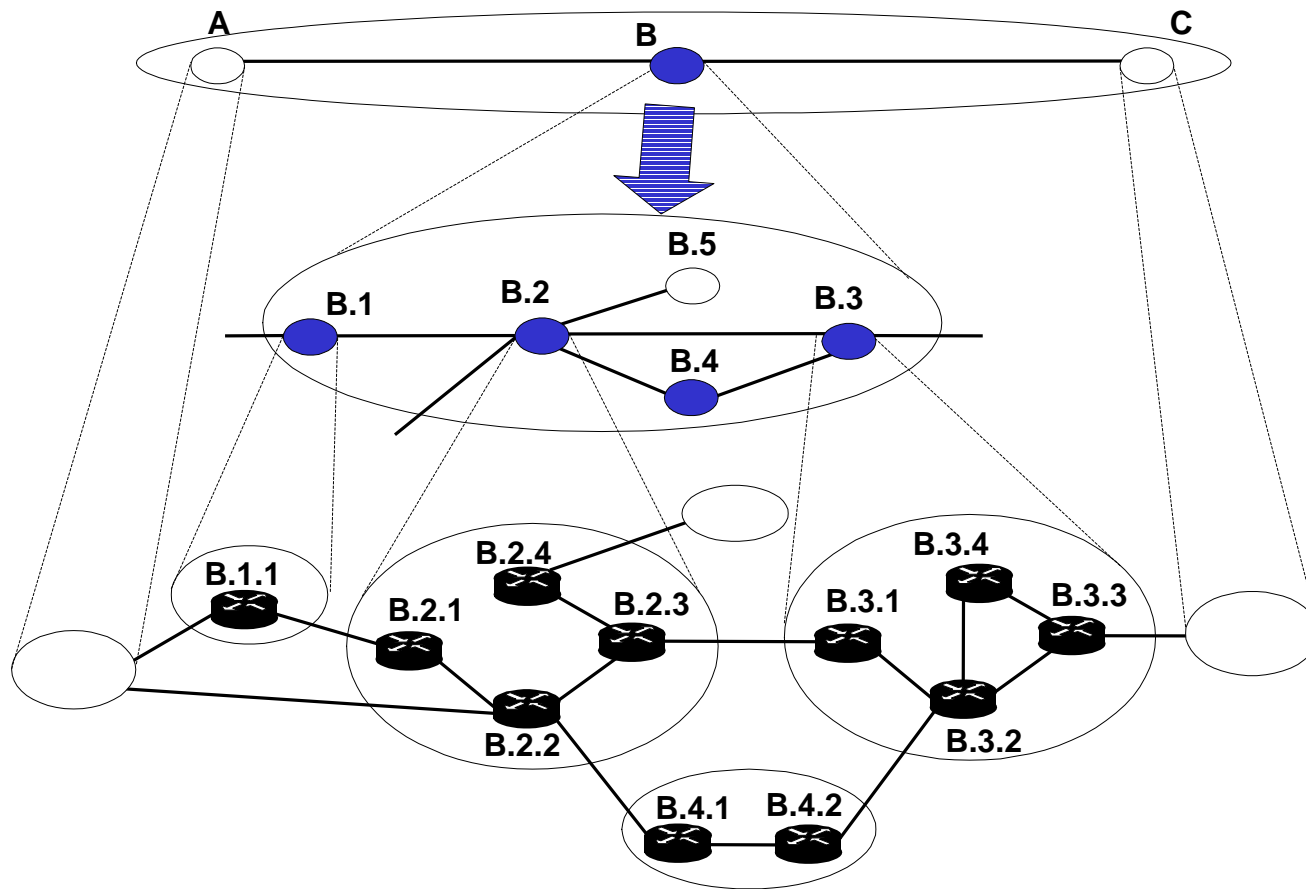
Solicitation ←  
Summarization  
Dissemination  
Installation  
Advertisement

# Solicitation (2)



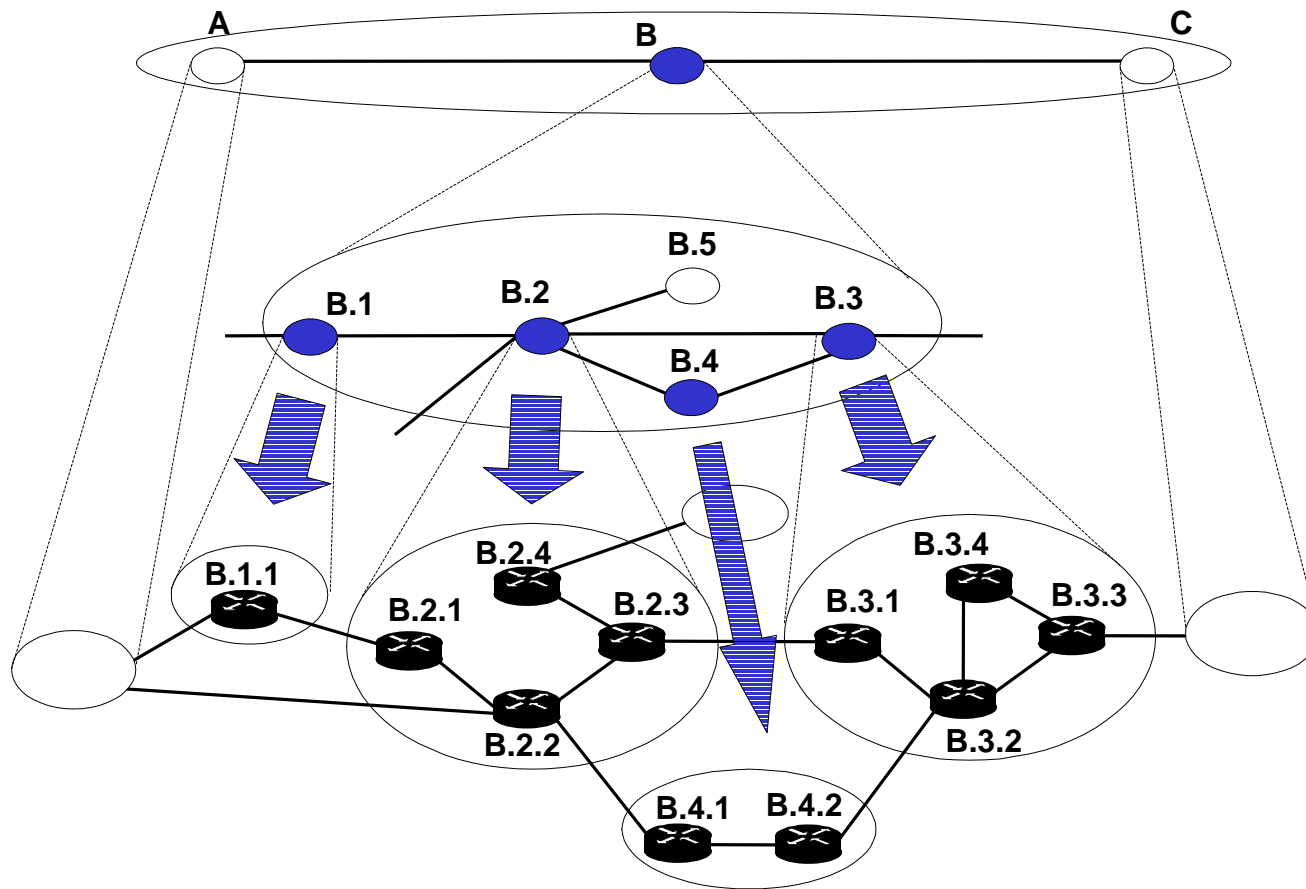
Solicitation ←  
Summarization  
Dissemination  
Installation  
Advertisement

# Solicitation (3)



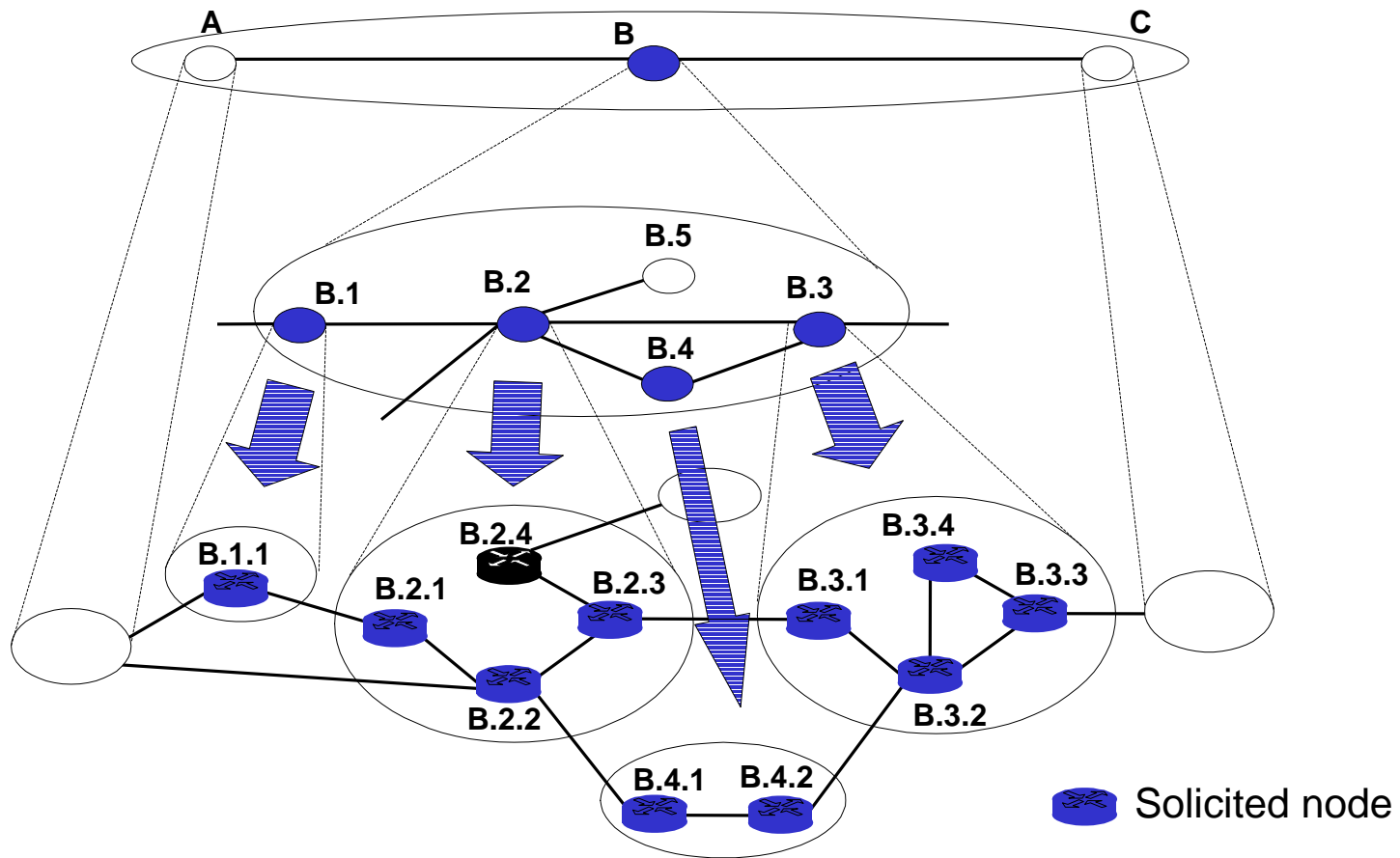
Solicitation ←  
Summarization  
Dissemination  
Installation  
Advertisement

# Solicitation (4)



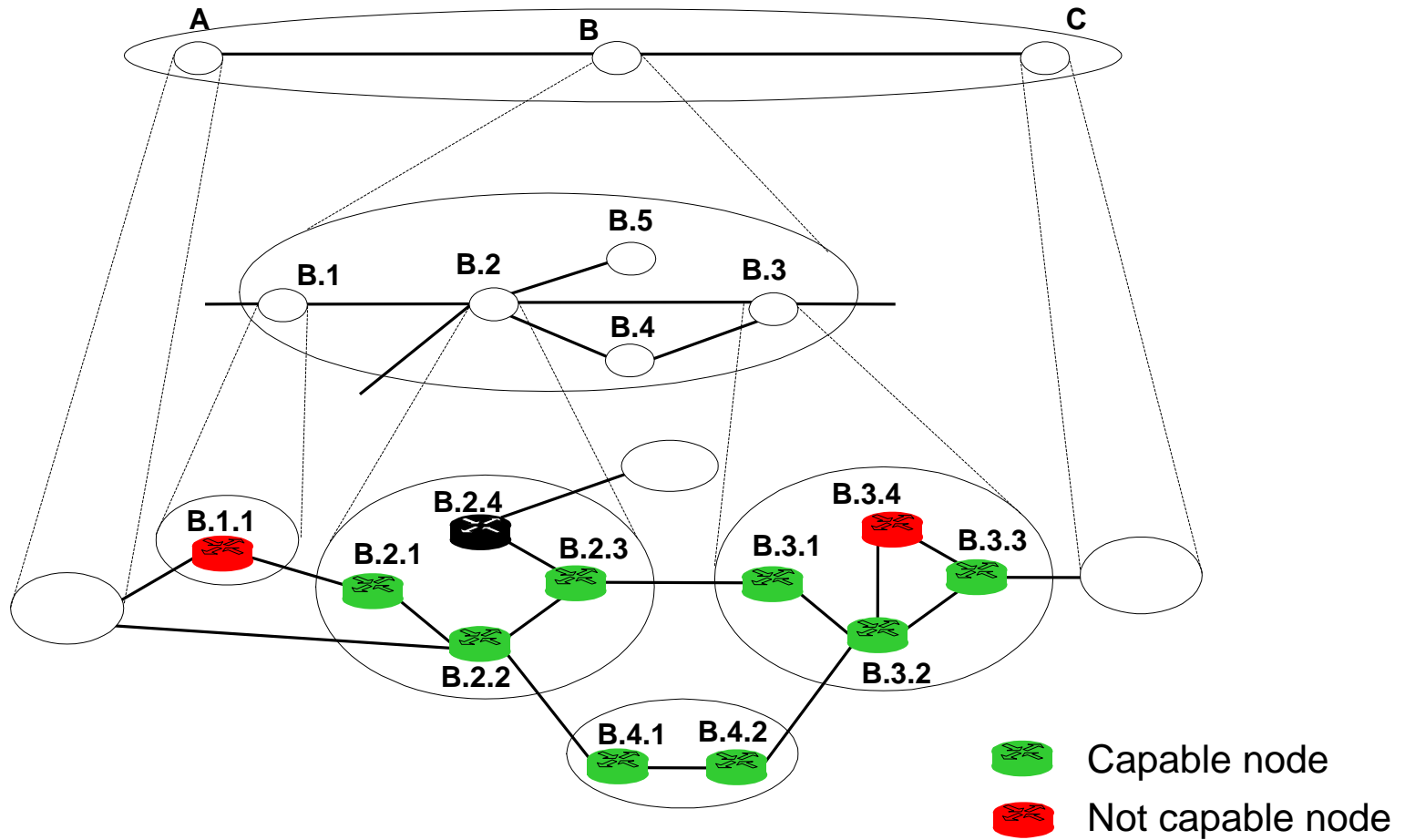
- Solicitation ←
- Summarization
- Dissemination
- Installation
- Advertisement

# Solicitation (5)



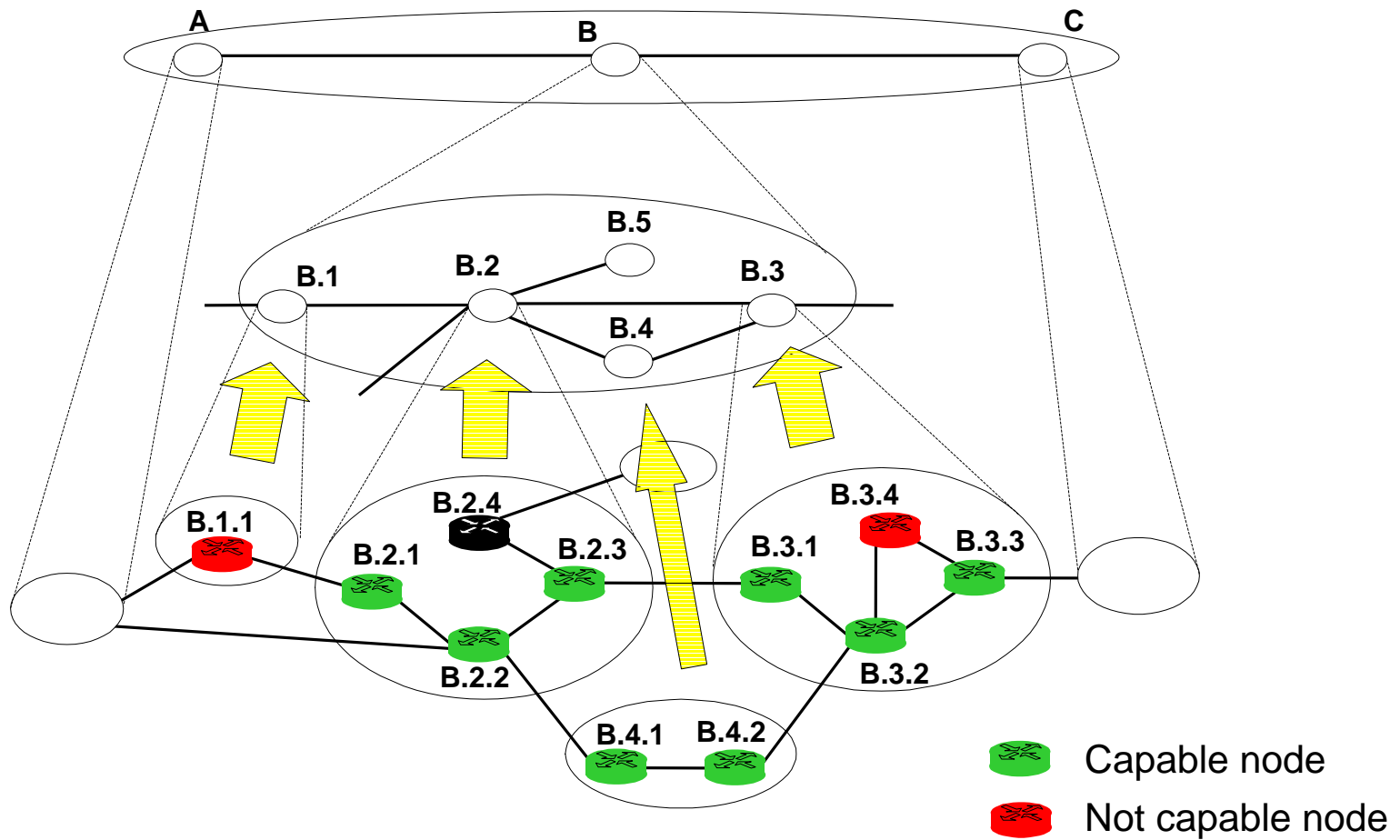
Solicitation ←  
 Summarization  
 Dissemination  
 Installation  
 Advertisement

# Evaluation



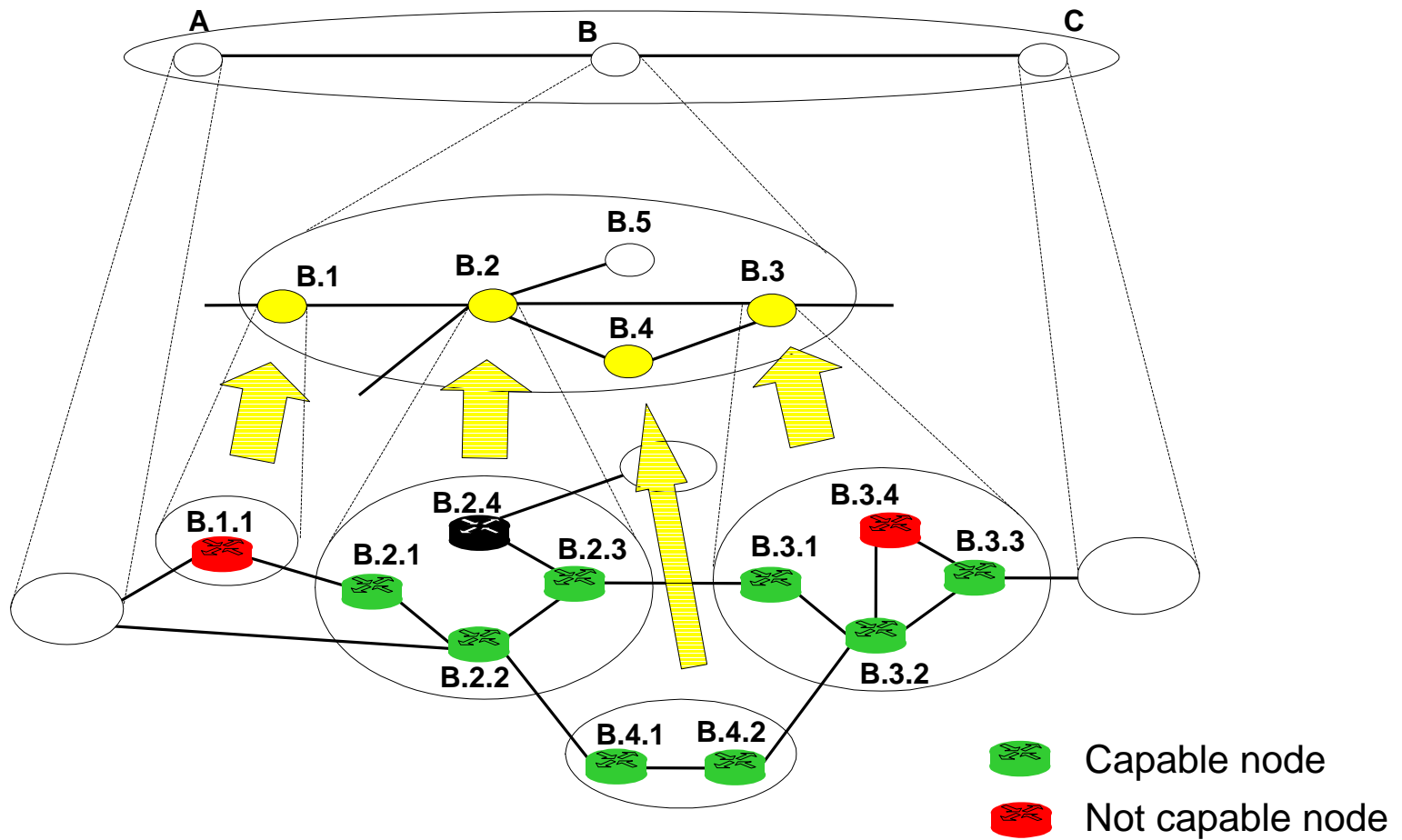
Solicitation  
 Summarization ←  
 Dissemination  
 Installation  
 Advertisement

# Summarization (1)



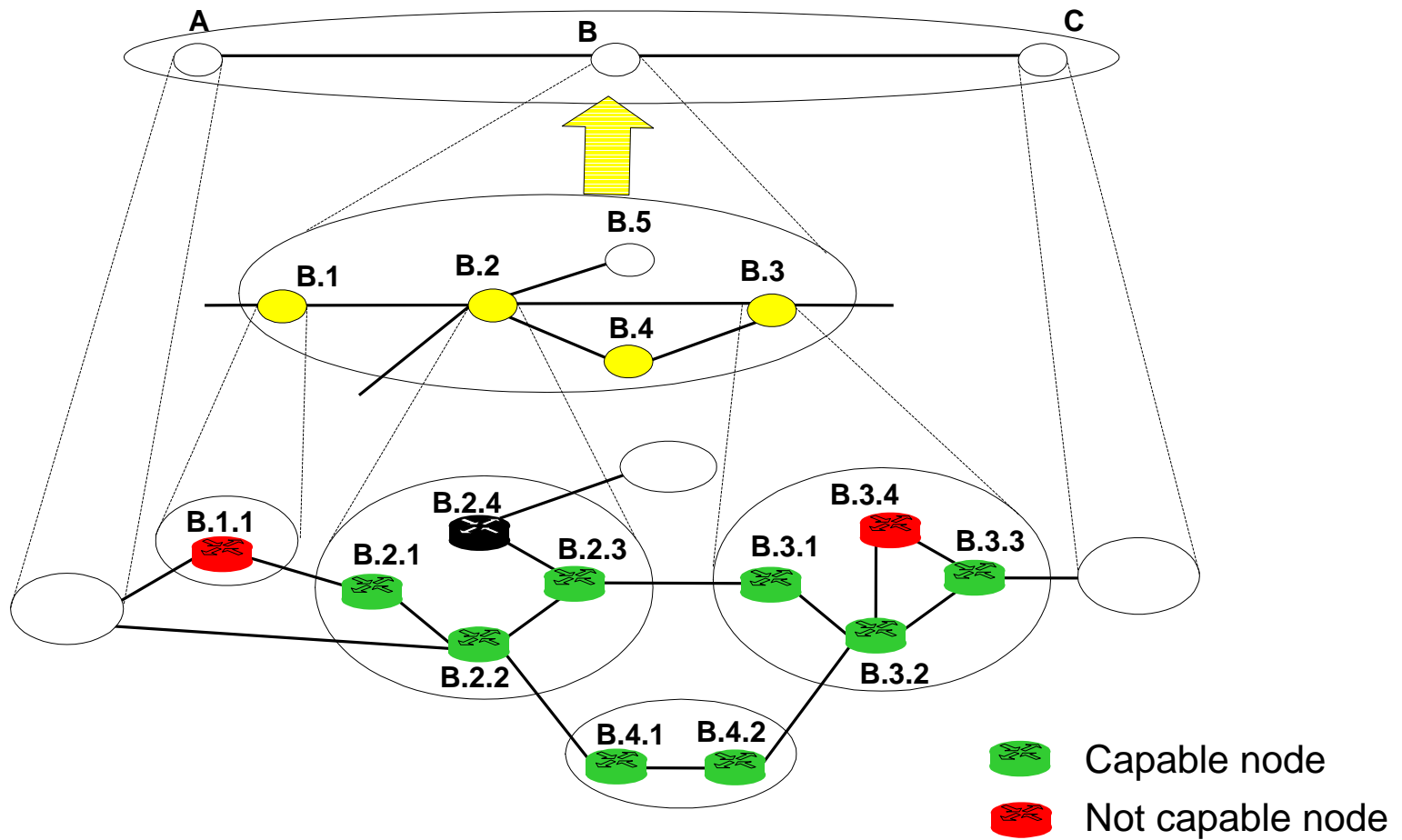
Solicitation  
 Summarization ←  
 Dissemination  
 Installation  
 Advertisement

# Summarization (2)



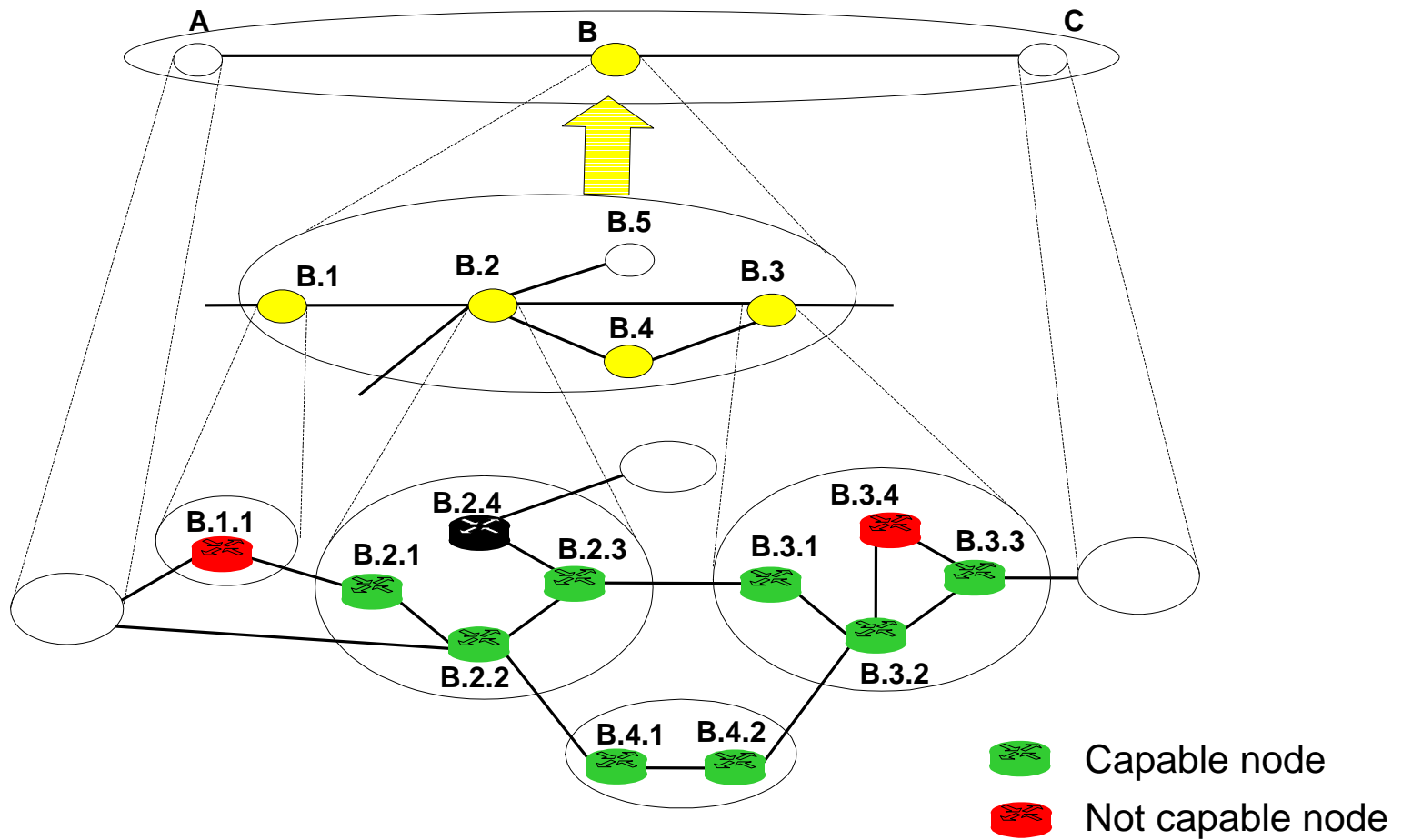
Solicitation  
 Summarization ←  
 Dissemination  
 Installation  
 Advertisement

# Summarization (3)



Solicitation  
 Summarization ←  
 Dissemination  
 Installation  
 Advertisement

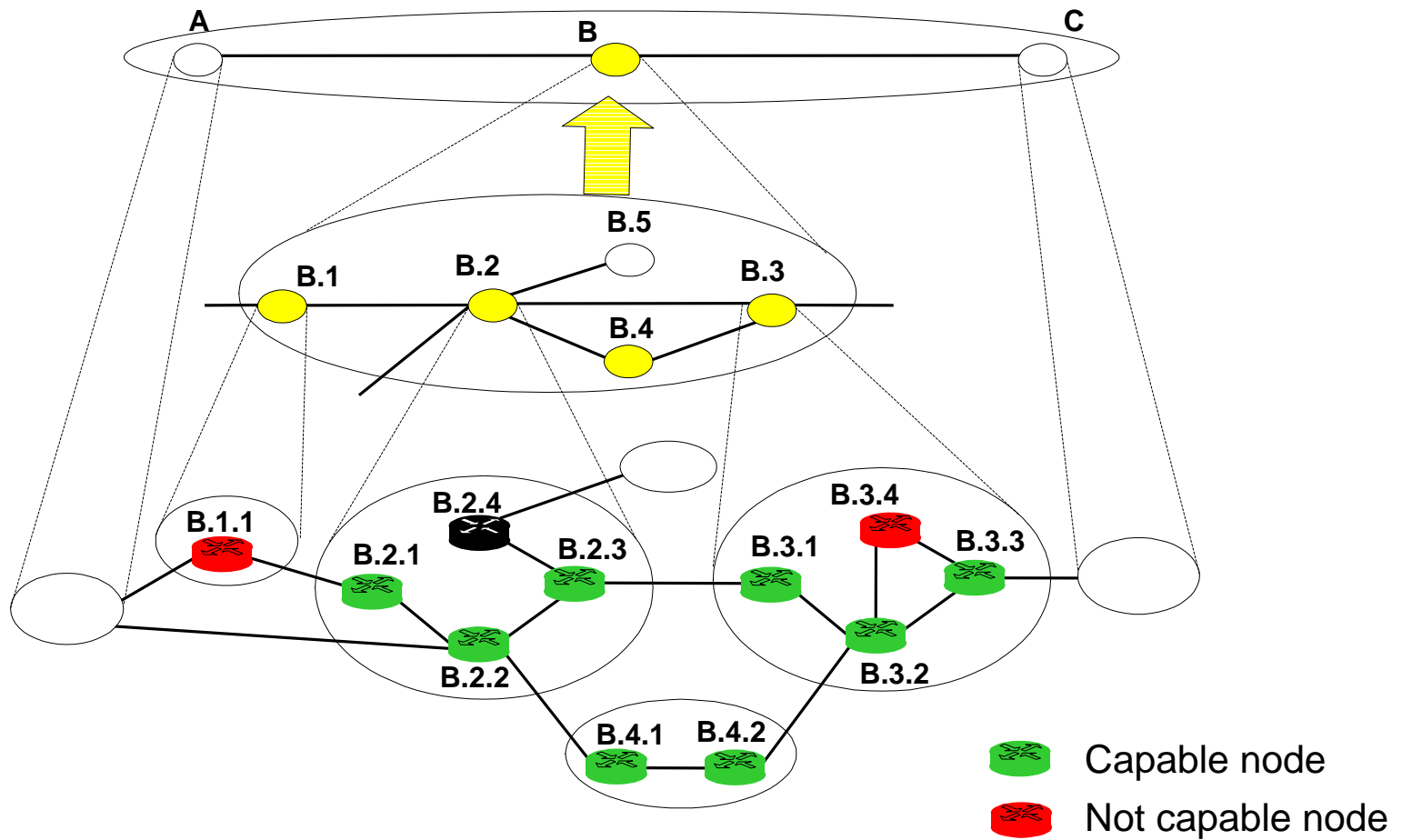
# Summarization (4)



# Summarized Information

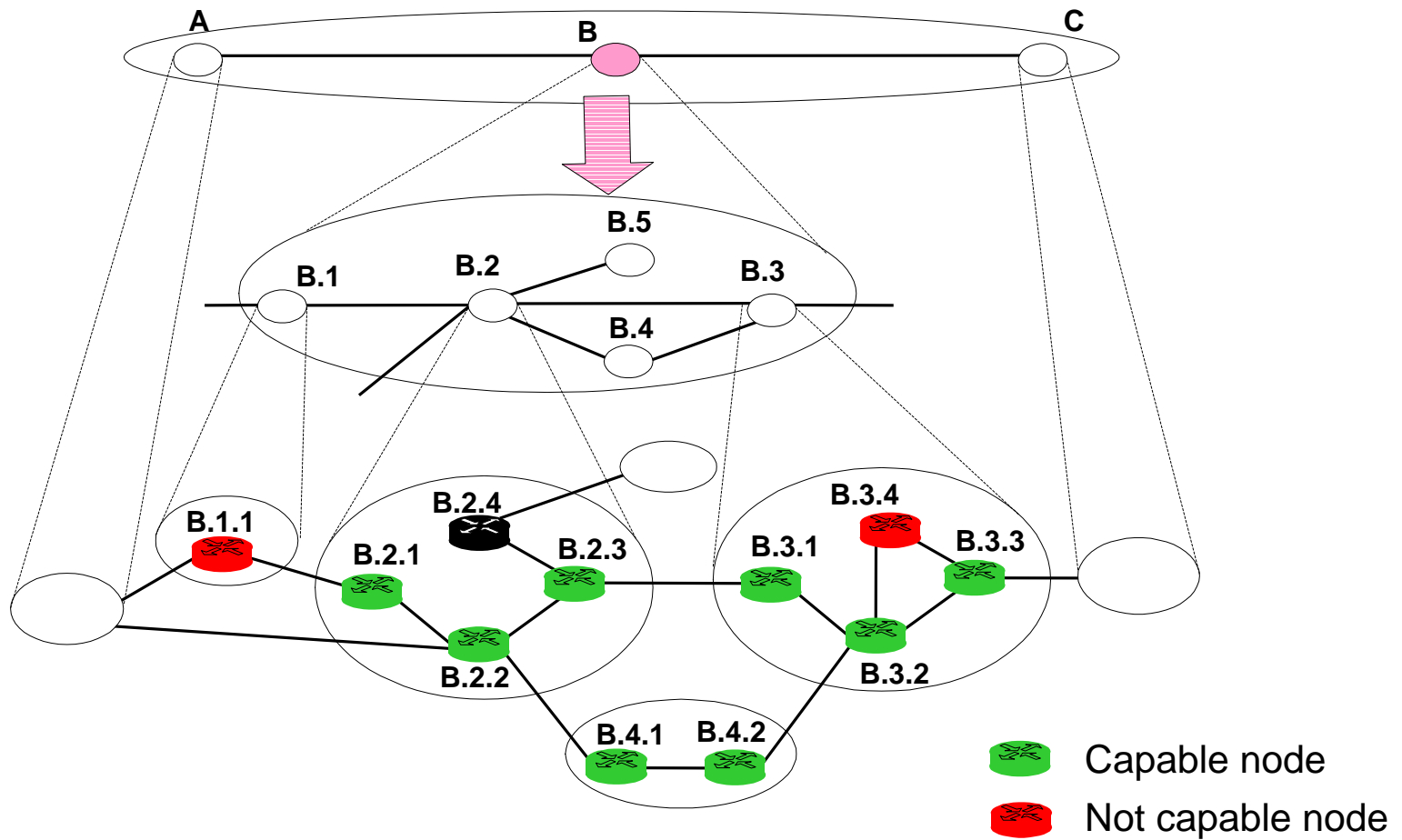
- Transition matrices contain:
  - Topology
  - Node capabilities
- Used during the next step
  - Select shortest path

$$T_{B_2} = T_{B_3} = \begin{pmatrix} 1 & \dots \\ 2 & 1 & \dots \\ 3 & 2 & 1 \end{pmatrix}; T_B = \begin{pmatrix} 0 & \dots \\ 0 & 1 & \dots \\ 0 & 5 & 1 \end{pmatrix}$$



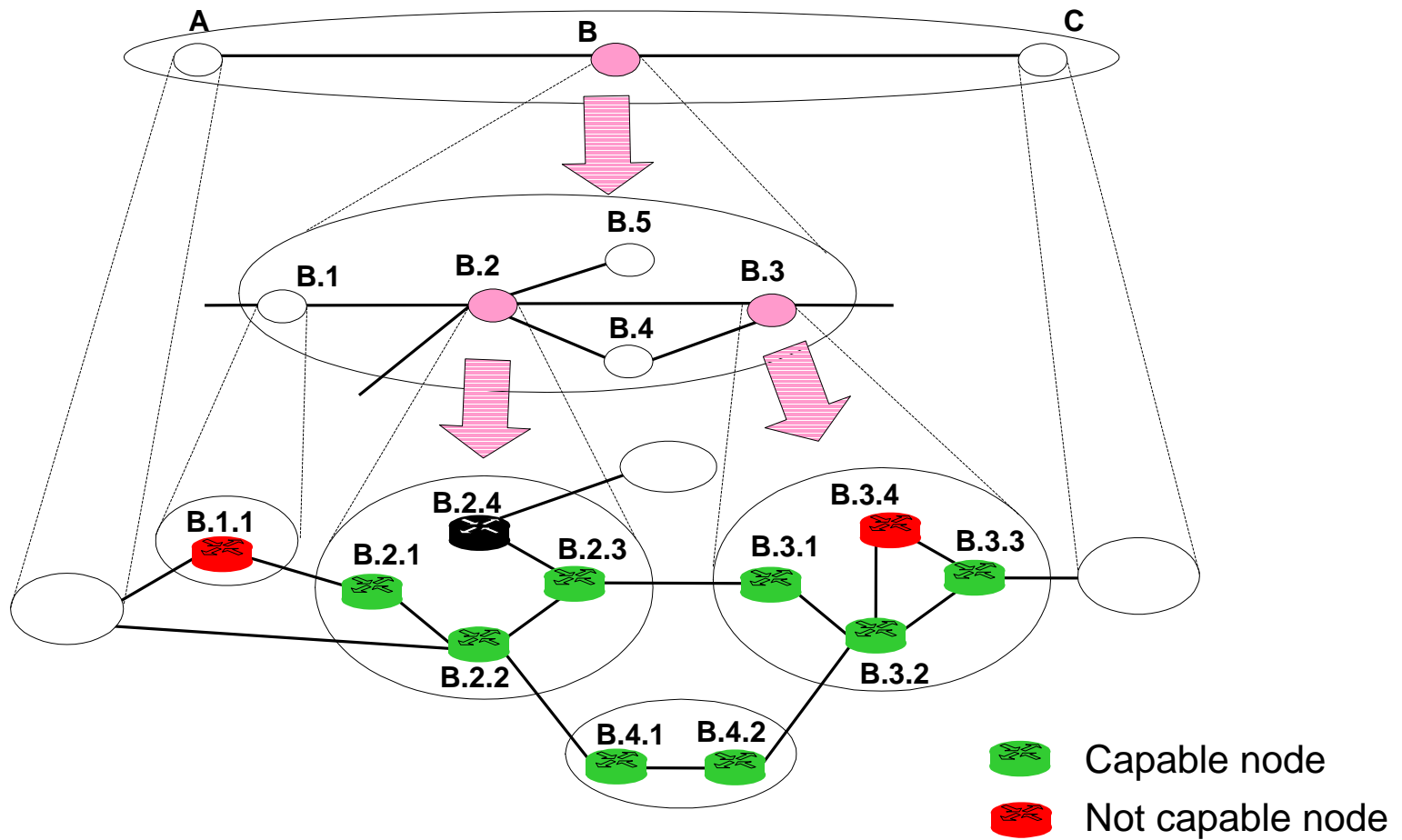
Solicitation  
 Summarization  
 Dissemination ←  
 Installation  
 Advertisement

# Dissemination (1)



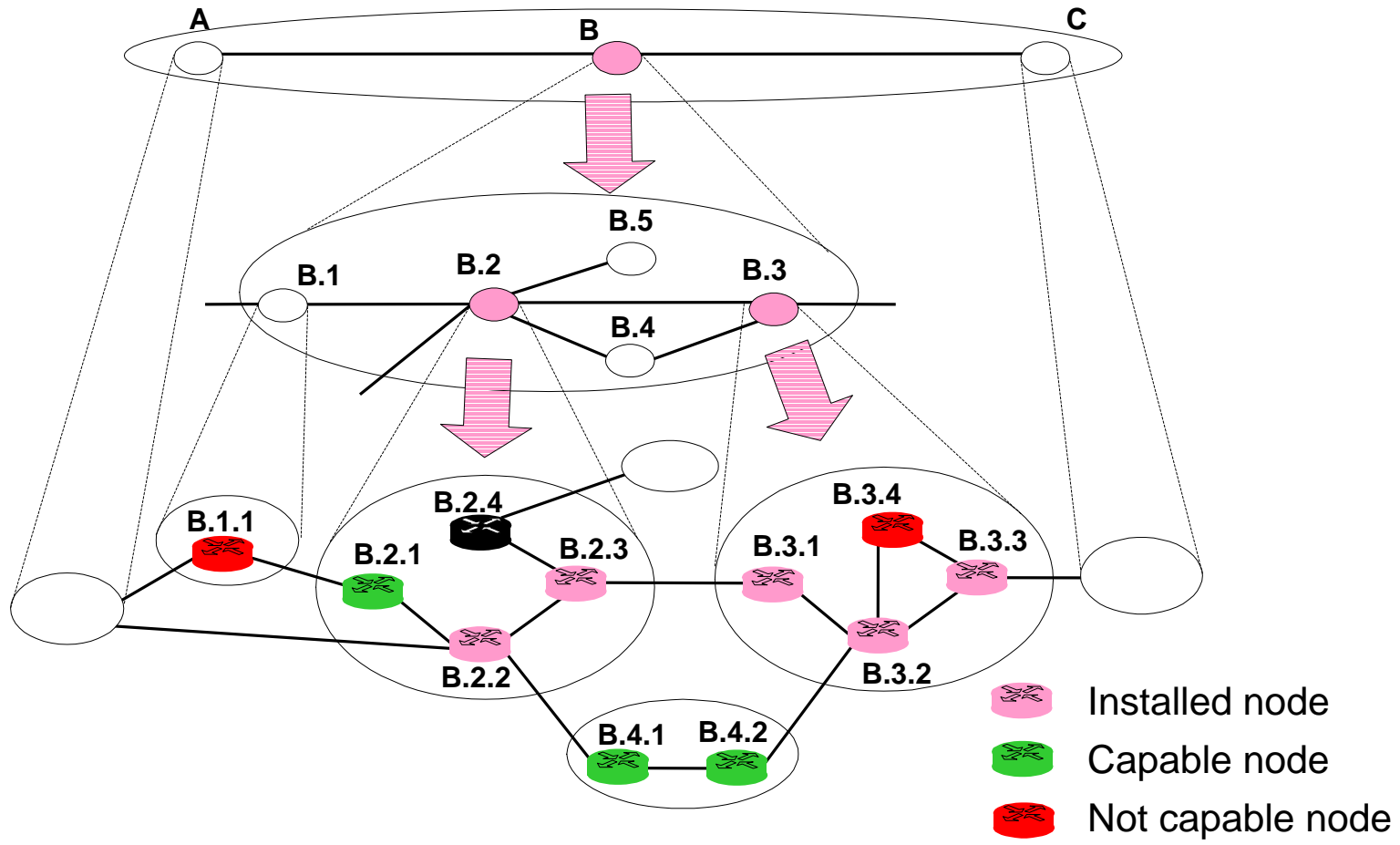
Solicitation  
 Summarization  
 Dissemination ←  
 Installation  
 Advertisement

# Dissemination (2)



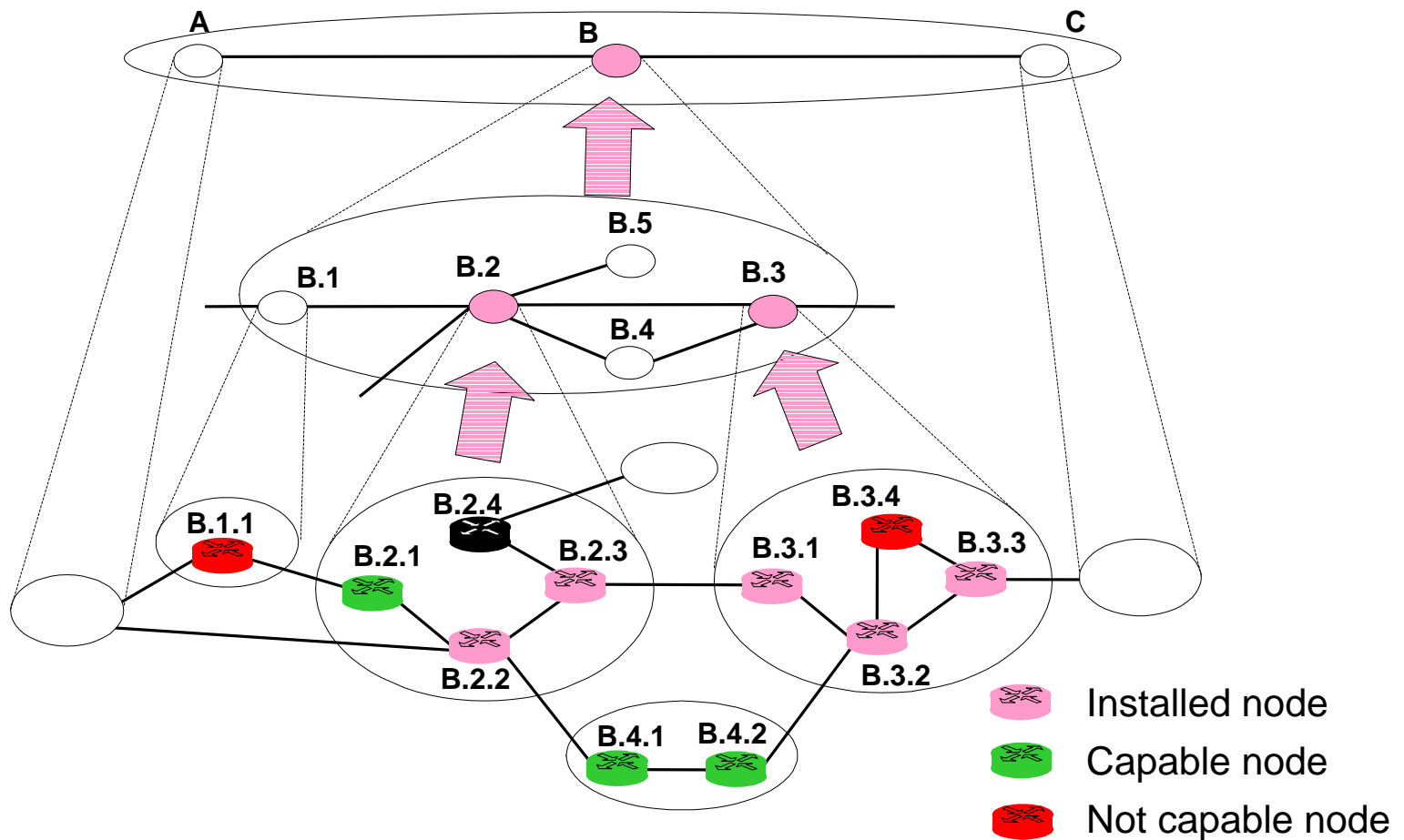
Solicitation  
 Summarization  
 Dissemination  
 Installation ←  
 Advertisement

# Installation



Solicitation  
 Summarization  
 Dissemination  
 Installation  
 Advertisement ←

# Advertisement



# Applicability

- Today: up to four levels of hierarchy
  - Routing: one type of information (QoS), refreshed often.
  - Service Deployment : different types of information, created once.
    - comparable complexity
- Combination of additive/restrictive metrics:
  - metrics and aggregation rules are service-dependent
    - avoid NP complexity

# Conclusion

- Internet evolves from providing pure connectivity, to high bandwidth, to custom services.
- Programmable networks bypass repetitive & lengthy protocol standardization
  - "once-for-all" standardized APIs
- Challenges for the intelligent network infrastructure:
  - rapidly deploying services
  - managing heterogeneity (in capabilities and interfaces)
  - supporting unpredicted services
  - remain scalable

# Future Work

- Investigating aggregate views of all four basic types of information
- Formalization of deployment protocol
- Simulation over very large scale networks
- Defining matchmaking procedure

# Further references

- R. Haas, P. Droz, B. Stiller, "A service deployment framework for programmable networks", IBM Research Report, December 2000.
- R. Haas, P. Droz, D. Bauer, "PNNI Augmented Routing and Proxy-PAR", Journal of Computer Networks, September 2000.
- T. Przygienda, P. Droz and R. Haas, "OSPF over ATM and Proxy-PAR", Internet RFC 2844, May 2000.
- L. Frelechoux, M. Osborne, R. Haas, "Topology Optimization of IP over ATM", ECUMN 2000, October 2000, Colmar, France.