SAPIENZA
UNIVERSITÀ DI ROMA

HSIENA

a hybrid publish/subscribe system

DESEC4LCCI

Workshop on Dependable and Secure Computing
for Large-scale Complex Critical Infrastructures

Magdeburg, 25 september 2012

Fabio Petroni
Leonardo Querzoni

MIDLAB Middleware Laboratory

Department of Computer, Control, and Management Engineering Antonio Ruberti

- The **publish/subscribe interaction scheme** [1] provides a form of communication, alternative to the standard clients/server model, where participants are decoupled with respect to:

TIME

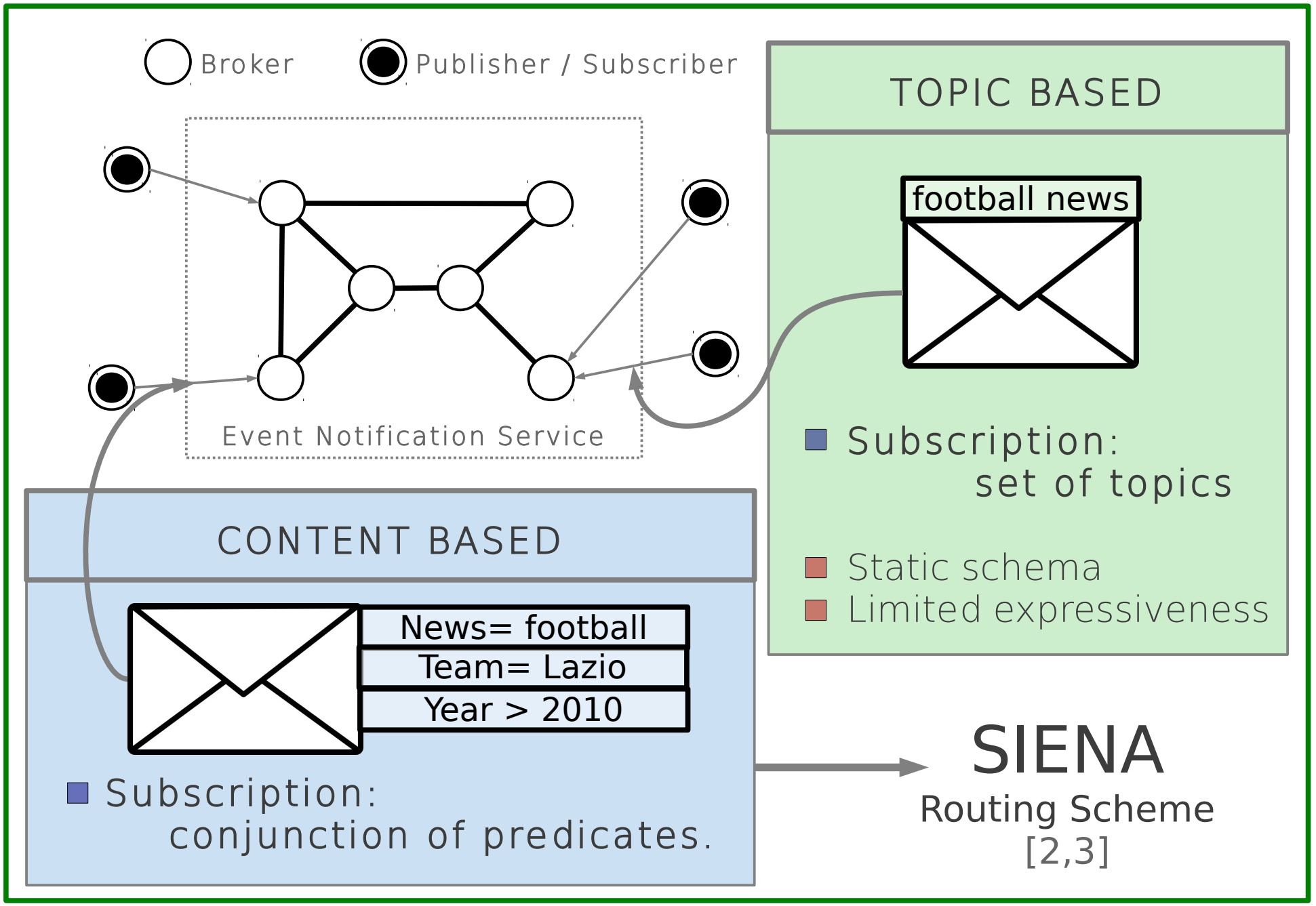
The participants do not need to be active at the same time, publishers might publish events while some subscribers are disconnected, and subscribers might get notified about some events while the original publisher is disconnected

SPACE

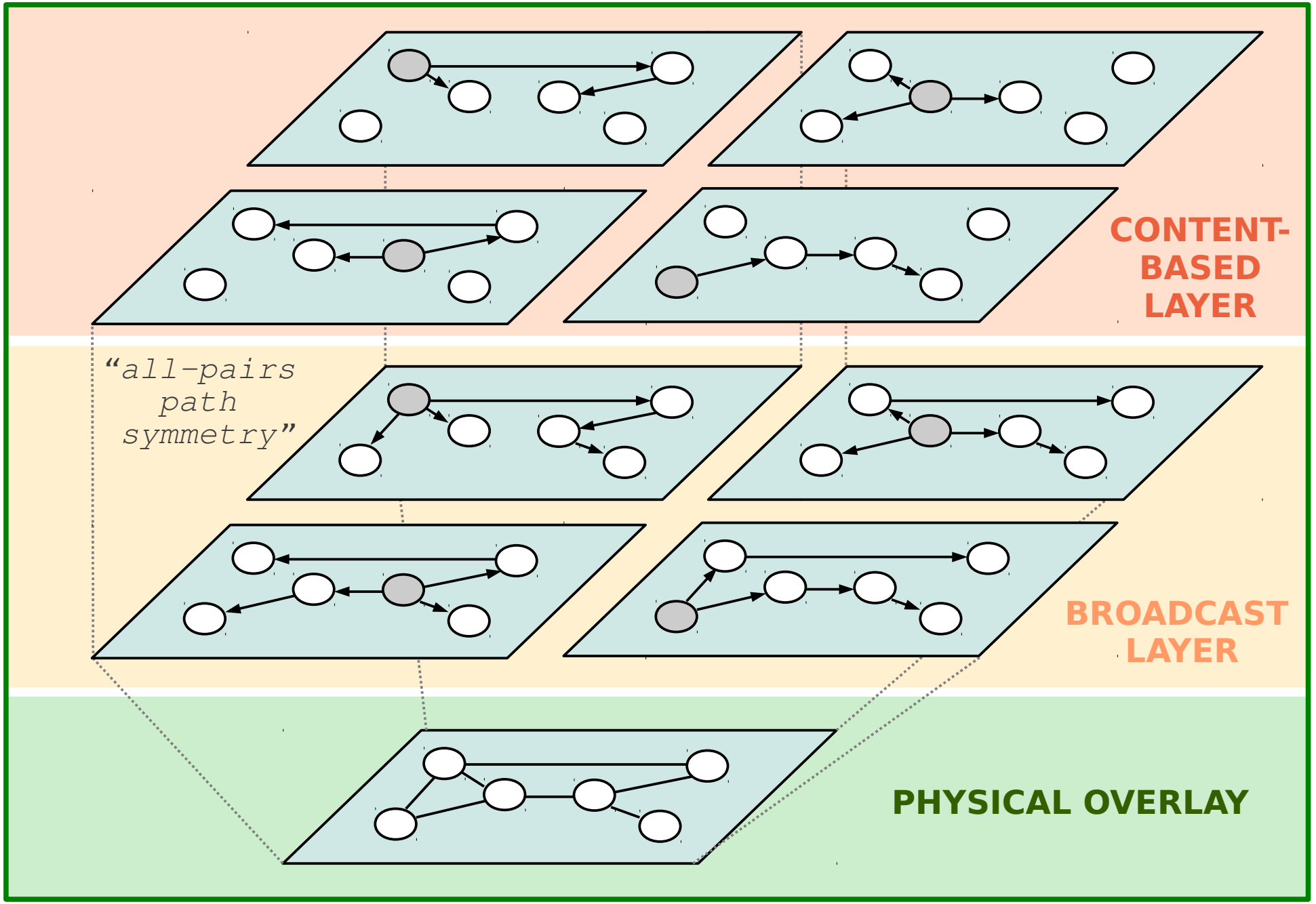
The interacting parties do not need to know each other. An event notification service (ENS) is responsible to gather publishers issued events and to diffuse them toward the subscribers

SYNCHRONIZATION

An event is asynchronously propagated to all subscribers that registered interest on it, and publishers are never blocked while producing events



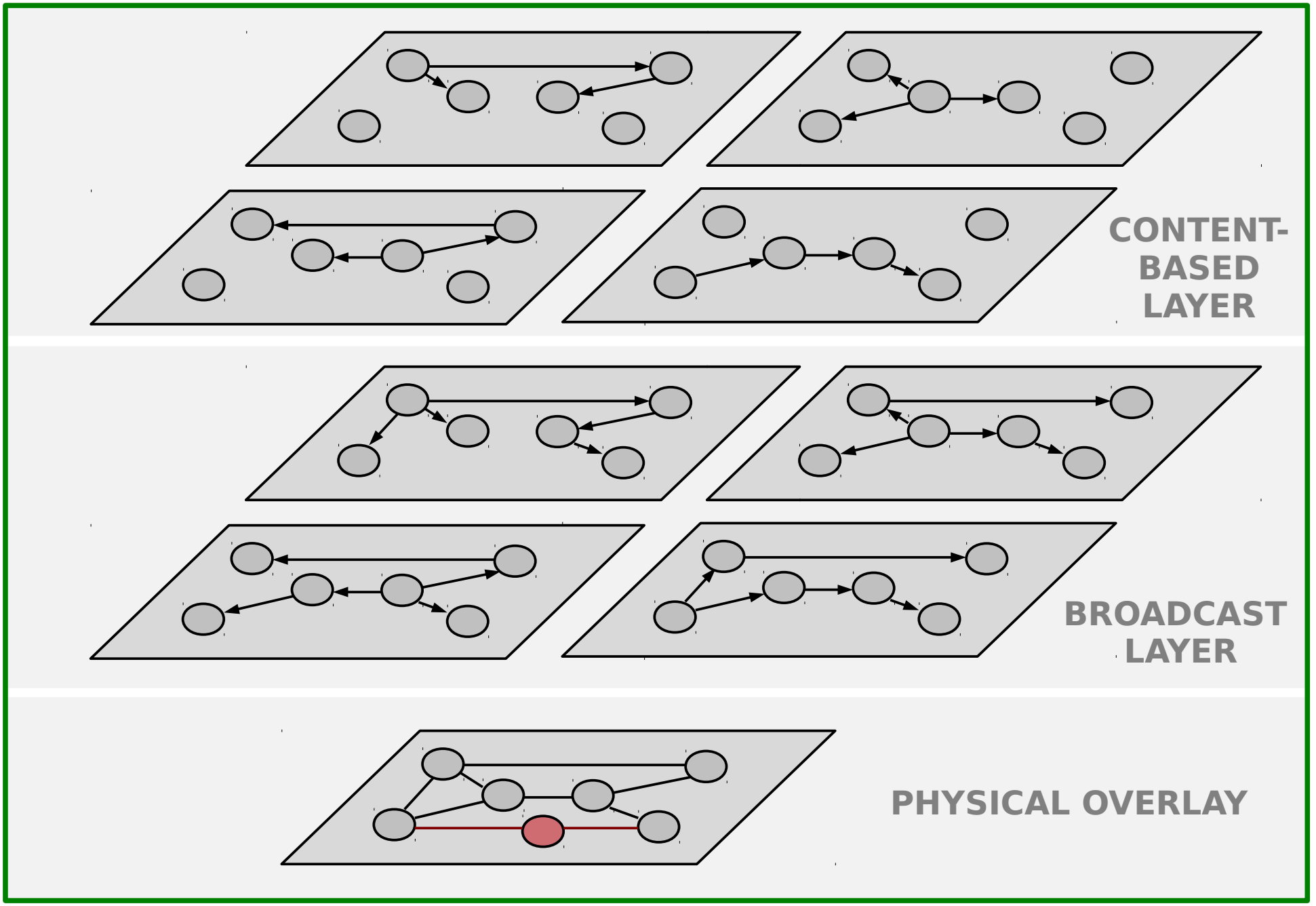
Topic-based vs Content-based


CONTENT-
BASED
LAYER
“all-pairs
path
symmetry”
BROADCAST
LAYER
PHYSICAL OVERLAY

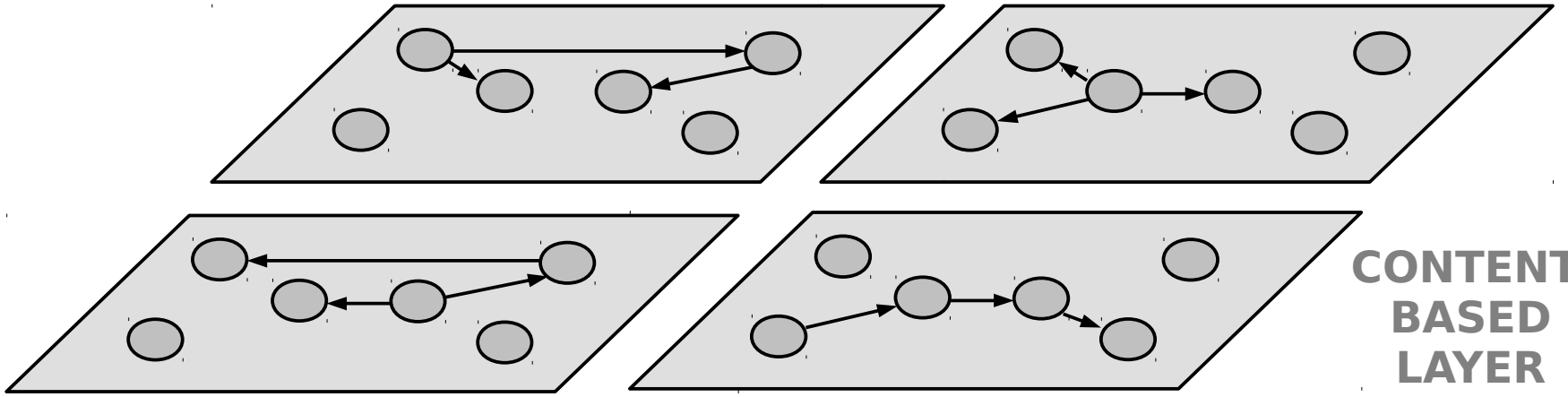
- SIENA suffers by the lack of adequate support to system reconfiguration
- In particular the addition or the removal of a new node to the system requires a full halt, followed by a manual reconfiguration of the broadcast and the content-based layers
- This can lead to large management overhead and reduced performance in dynamic or large-scale environments
- GOAL: make SIENA layers self-organizing, so that the whole system needs a reduced management by human administrators

The problems of SIENA

- XSIENA [4] proposes a soft state approach
- The idea is to use timed subscriptions, and re-issue them periodically, in order to automatically manage and restore the state of crashed subscribers and publishers
- Cugola et al. [5] proposed a solution for the single broadcast tree case, limiting the reconfiguration to a well defined path
- Moreover, they have clearly defined the **reconfiguration problem**



Reconfiguration problem



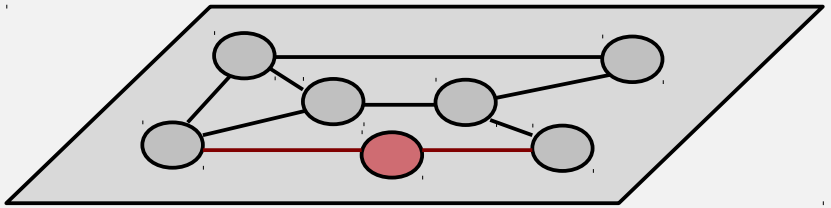
CONTENT-BASED LAYER

1

reconfiguration of the overlay network

to maintain connectivity among participants

BROADCAST LAYER



PHYSICAL OVERLAY

Reconfiguration problem

2

reconfiguration of the subscription information

to bring the routing tables up-to-date with the changes

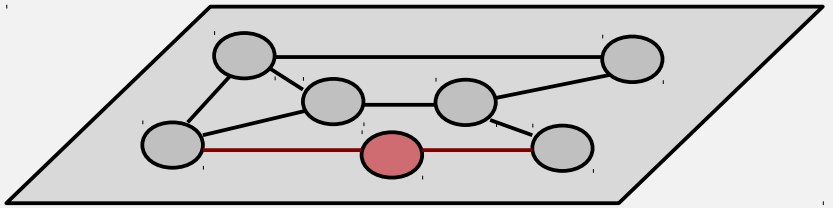
CONTENT-BASED LAYER

1

reconfiguration of the overlay network

to maintain connectivity among participants

BROADCAST LAYER



PHYSICAL OVERLAY

Reconfiguration problem

2

reconfiguration of the subscription information

to bring the routing tables up-to-date with the changes

CONTENT-BASED LAYER

1

reconfiguration of the overlay network

to maintain connectivity among participants

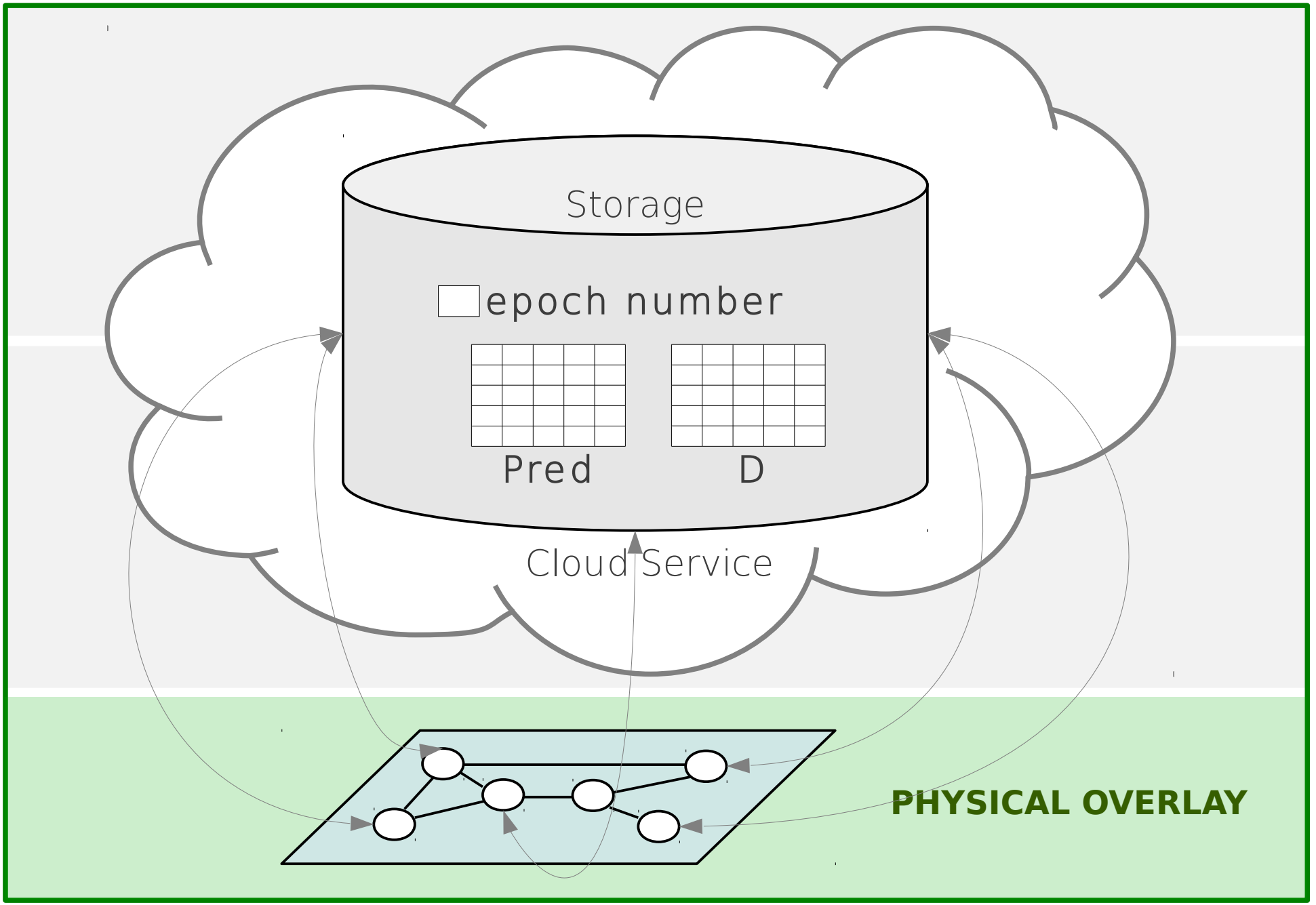
BROADCAST LAYER

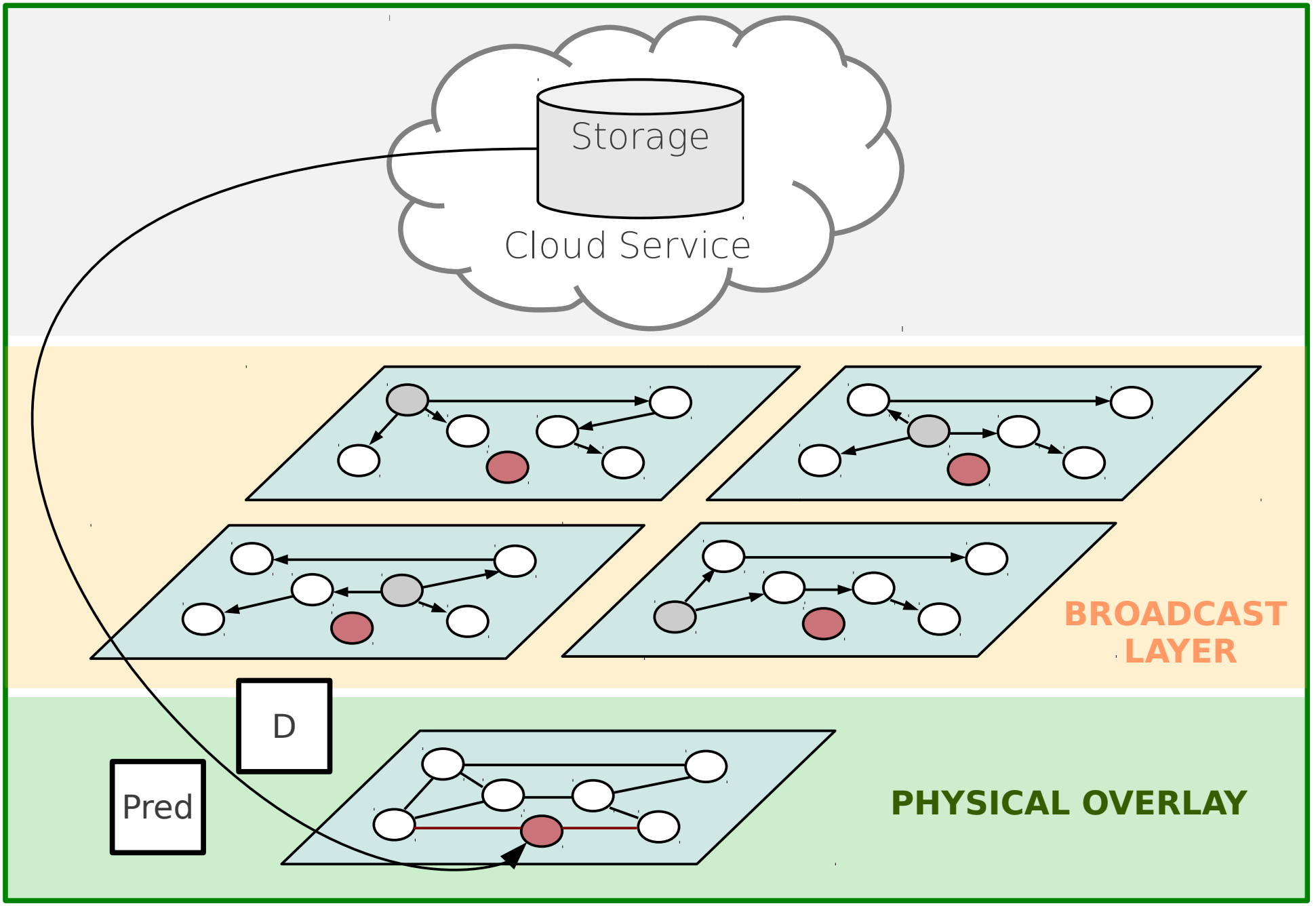
3

minimization of event loss during reconfiguration

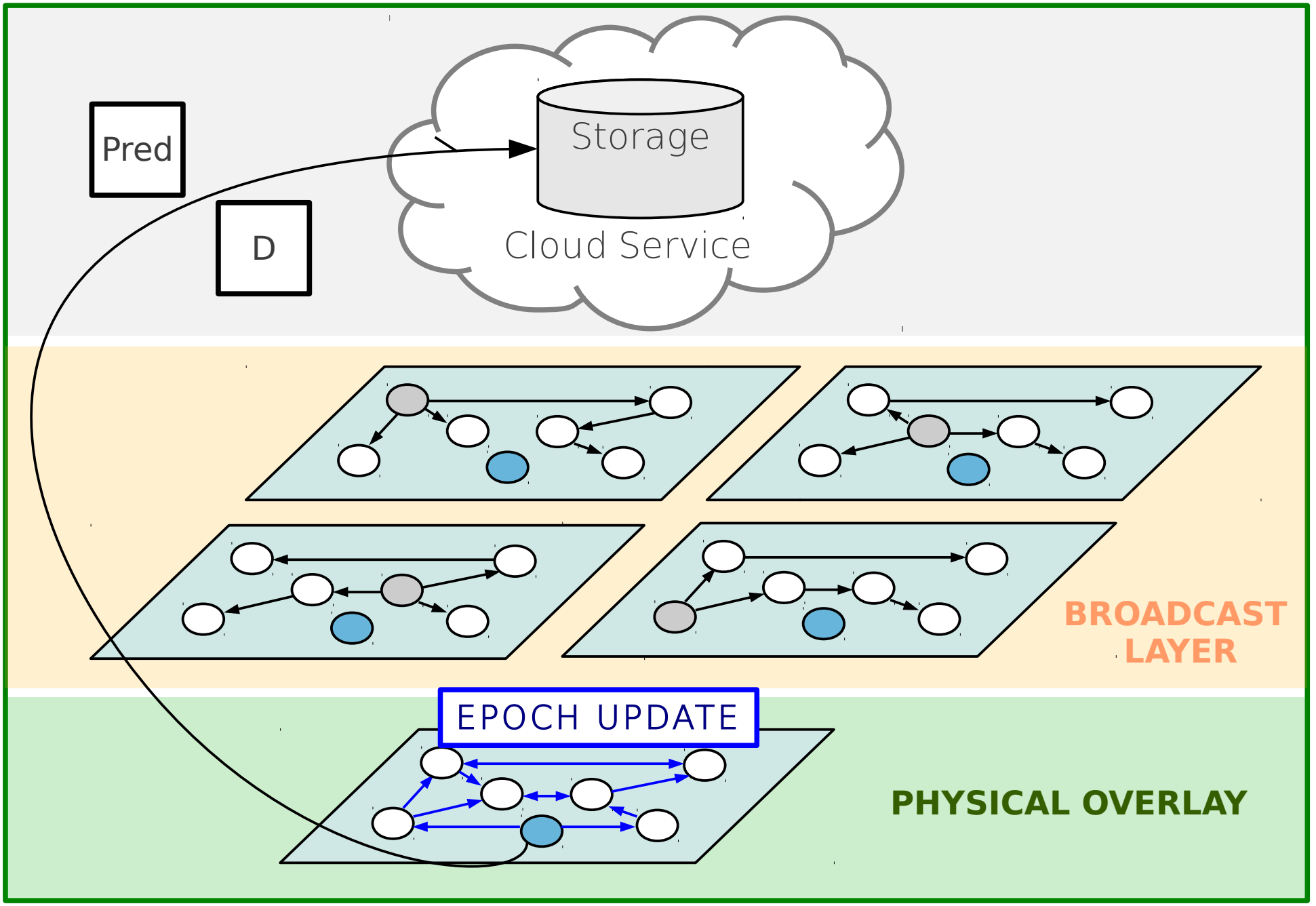
PHYSICAL OVERLAY

- How do we coordinate the brokers in the system, since they are geographically spread, independent and possibly linked by networks with unpredictable latency (like internet)?
- Solutions based on a central coordinator may be inefficient for several reasons:
  - fault tolerance (single point of failure)
  - organizational (different administrative domains)
  - scalability
- PROPOSAL: use a storage system reliable and easily accessible by all processes in the system, supplied by a cloud provider
- The cloud provider takes care of maintaining the storage available and consistent
- We only have to manage concurrent accesses of brokers to this storage
- Montresor and Abeni (2011)[6] proposed a similar idea in the context of gossip algorithms

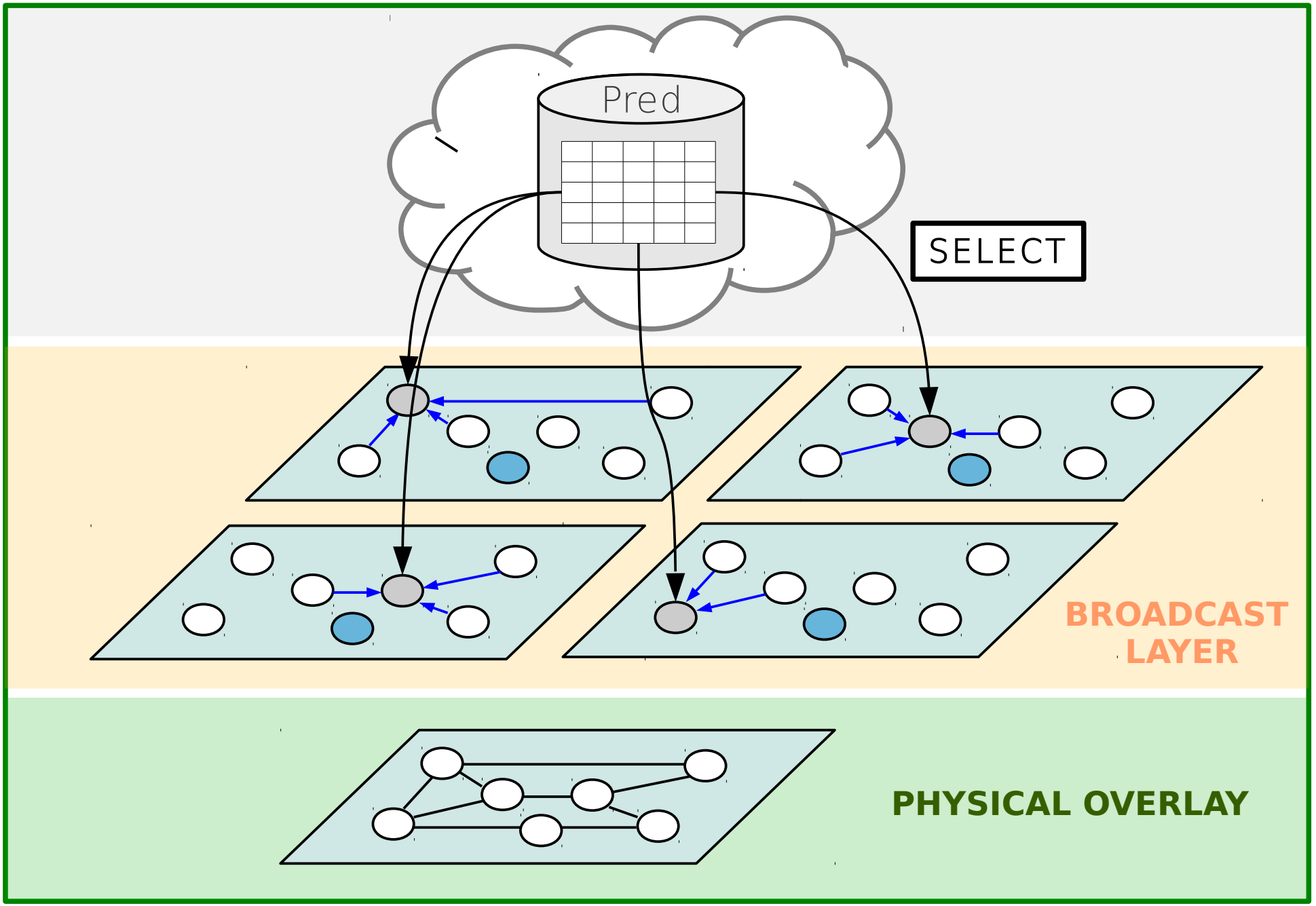

Storage
epoch number
Pred
D
Cloud Service
PHYSICAL OVERLAY



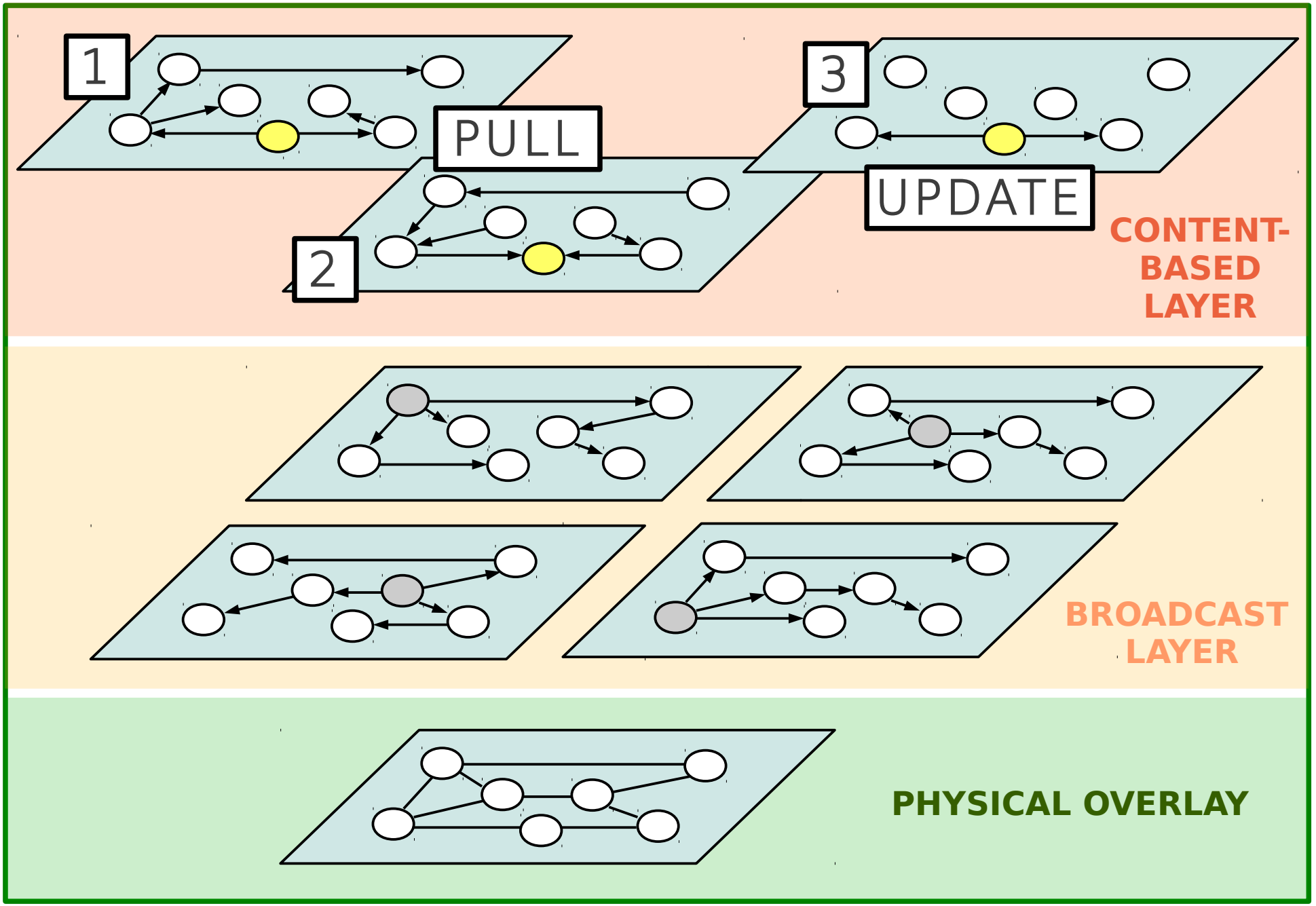
HSIENA – insertion of a new broker 1/2



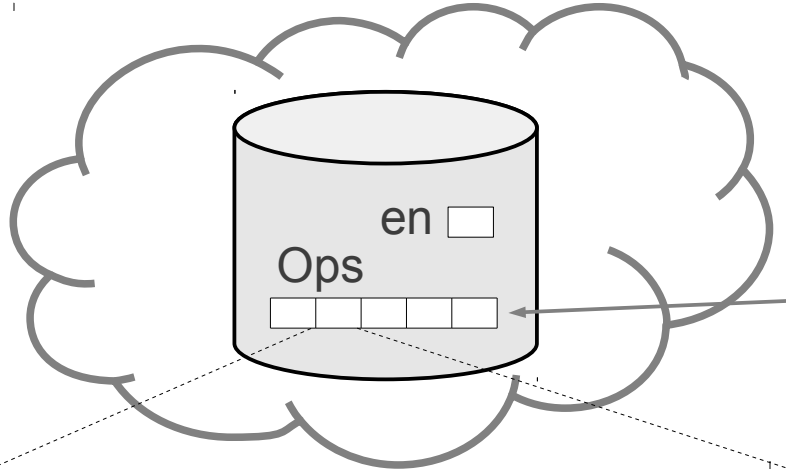
HSIENA – insertion of a new broker 2/2



HSIENA – reconfiguration of the overlay network



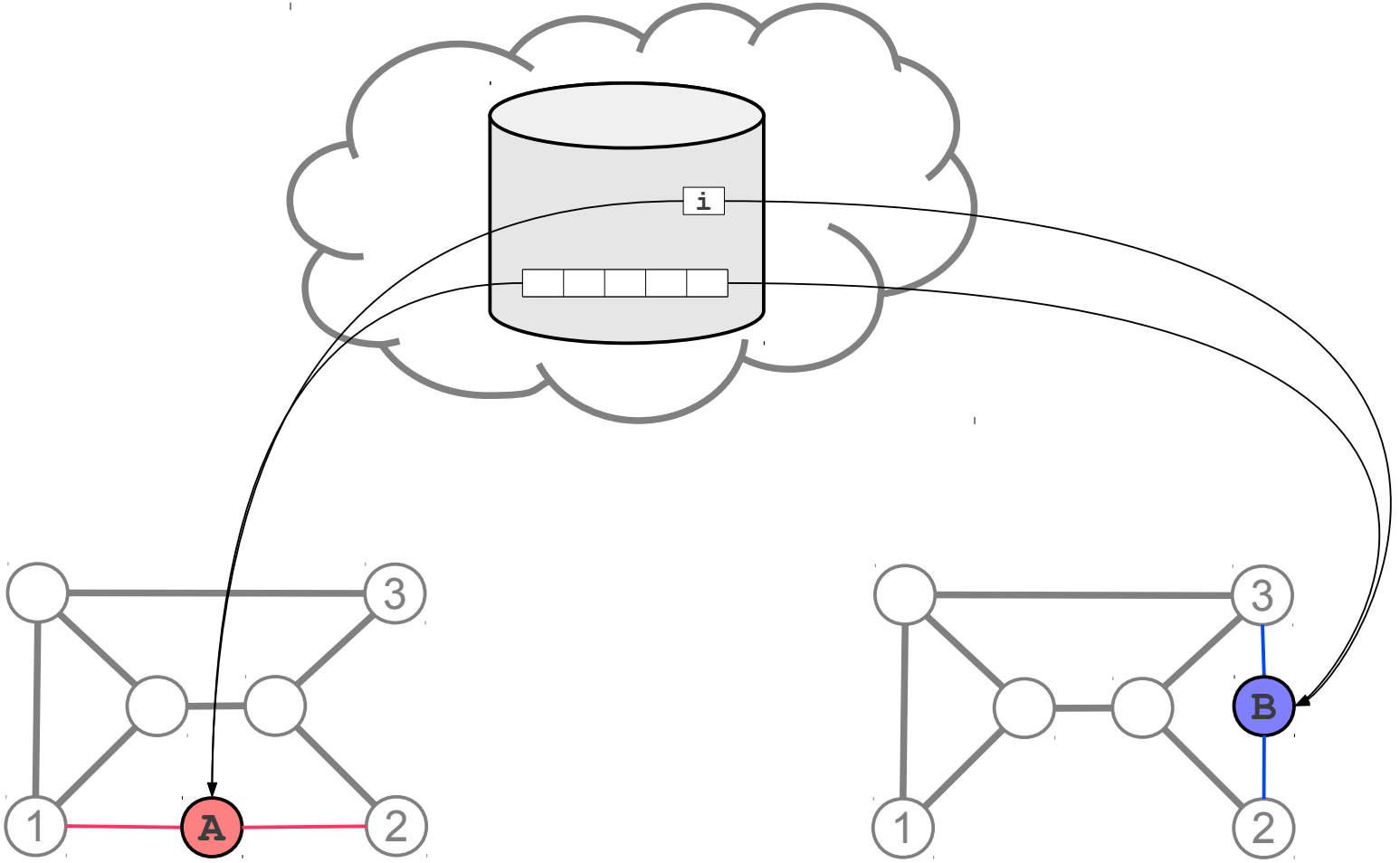
HSIENA – reconfiguration of the subscription info



queue of ongoing operations

< **i**, [**INS**/**REM**], **y**, **V** >

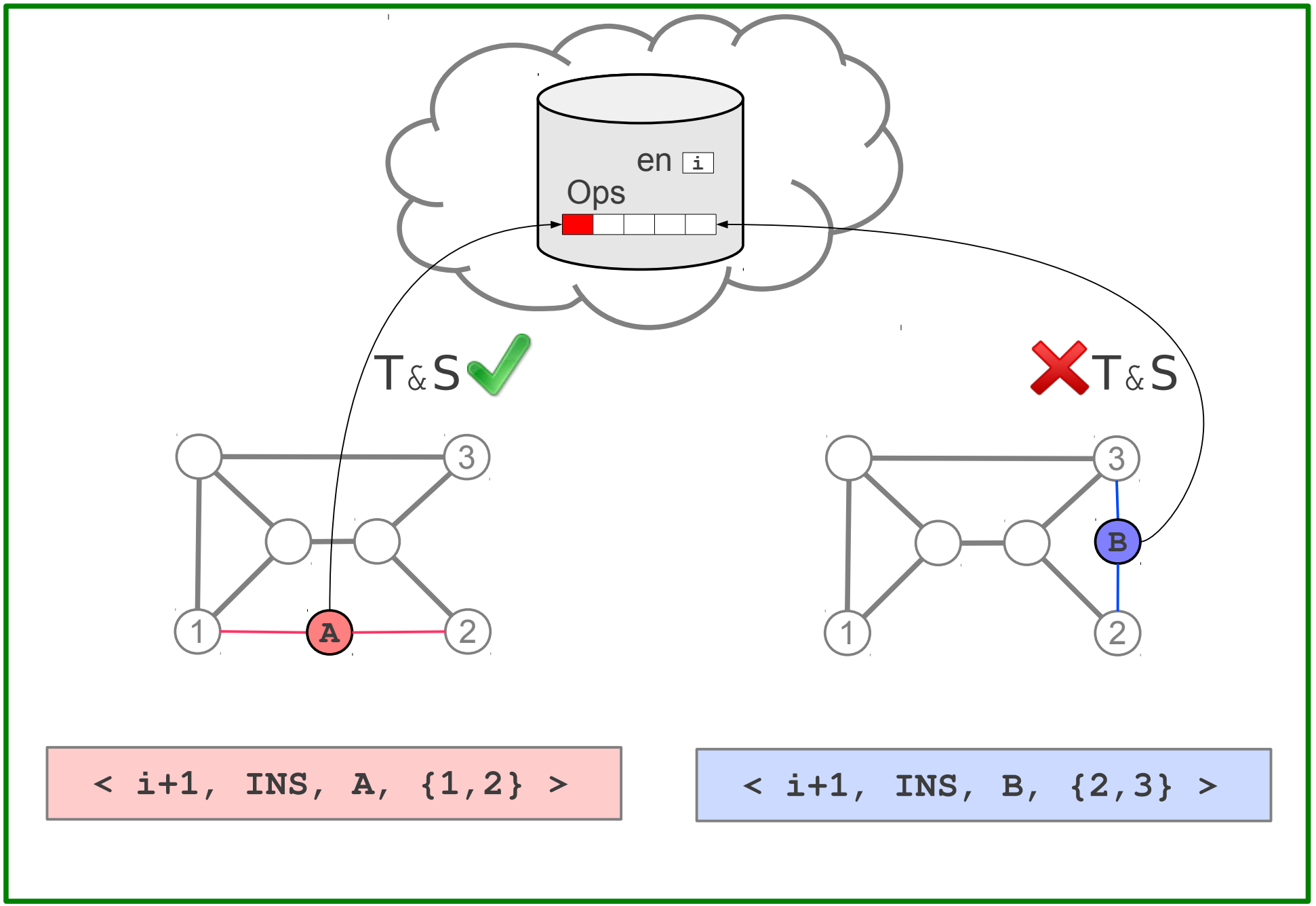
- *"The broker y is performing a [INS/REM] operation that will bring the system to epoch i. V is a set of broker ids used only for INS operations (neighbors of y)"*
- Assumption: the storage service provides a **test-and-set** primitive

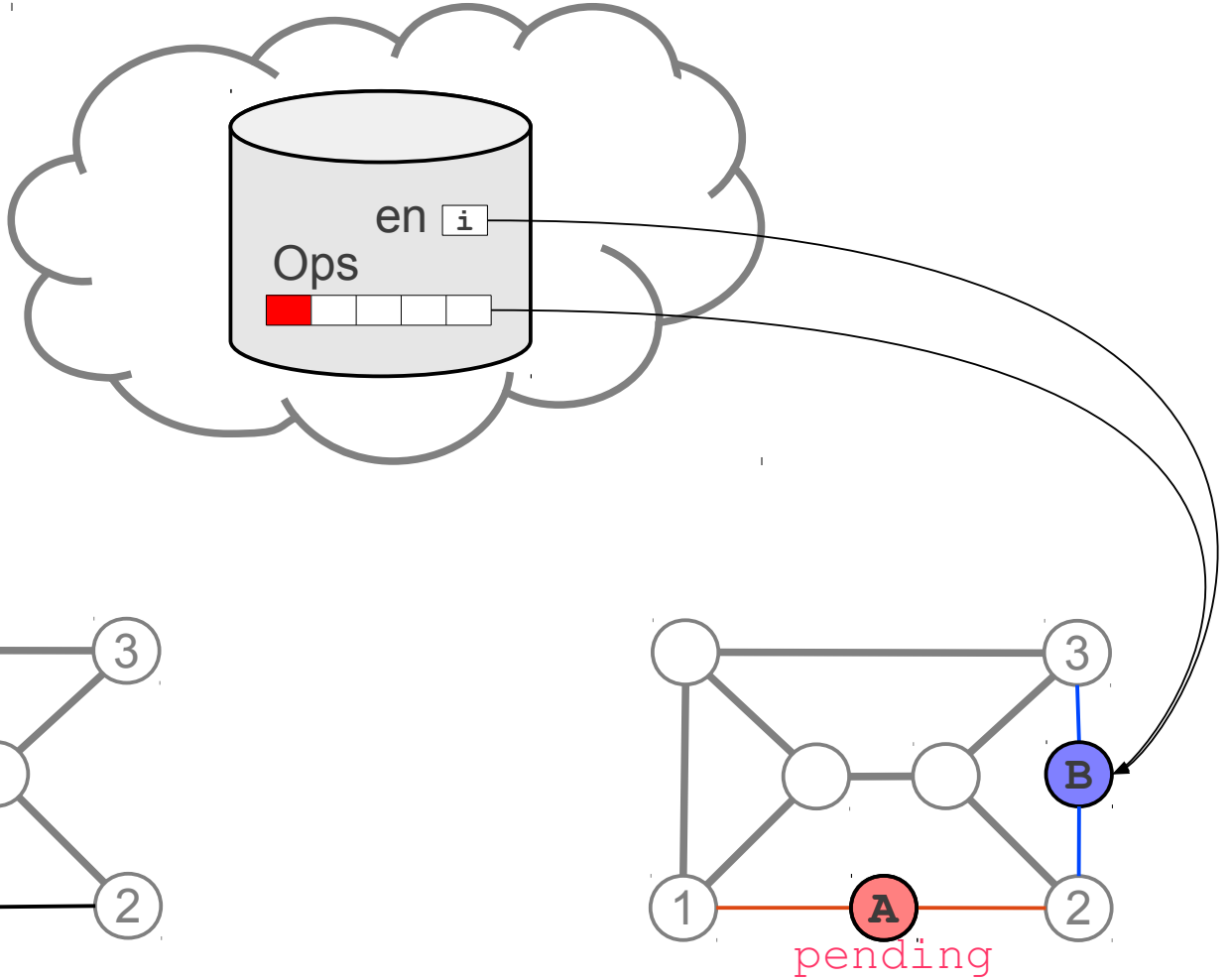


`< i+1, INS, A, {1,2} >`

`< i+1, INS, B, {2,3} >`

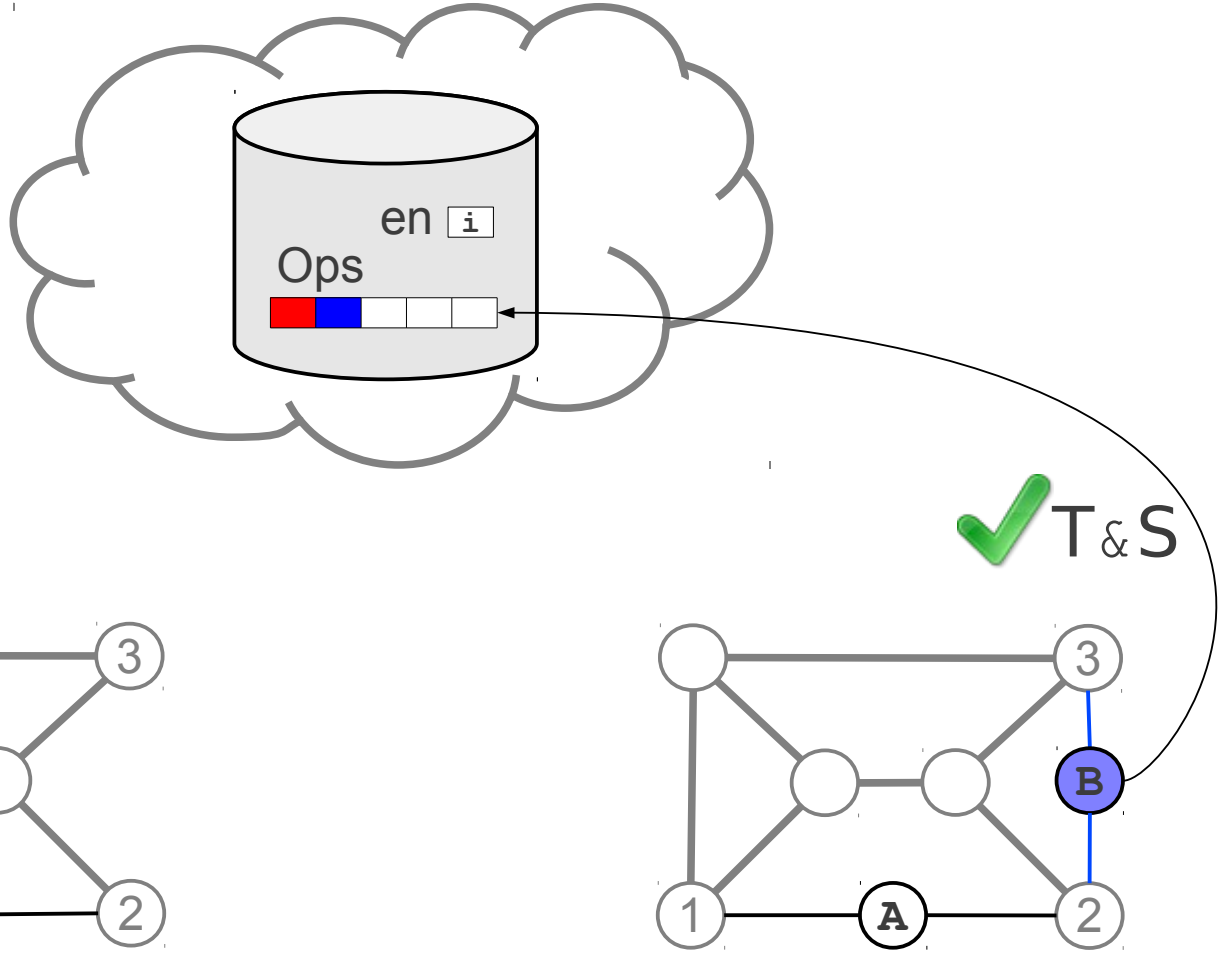
HSIENA – Concurrency 2/7


en i
Ops
T&S
T&S
3
1
A
2
3
B
1
2
< i+1, INS, A, {1,2} >
< i+1, INS, B, {2,3} >



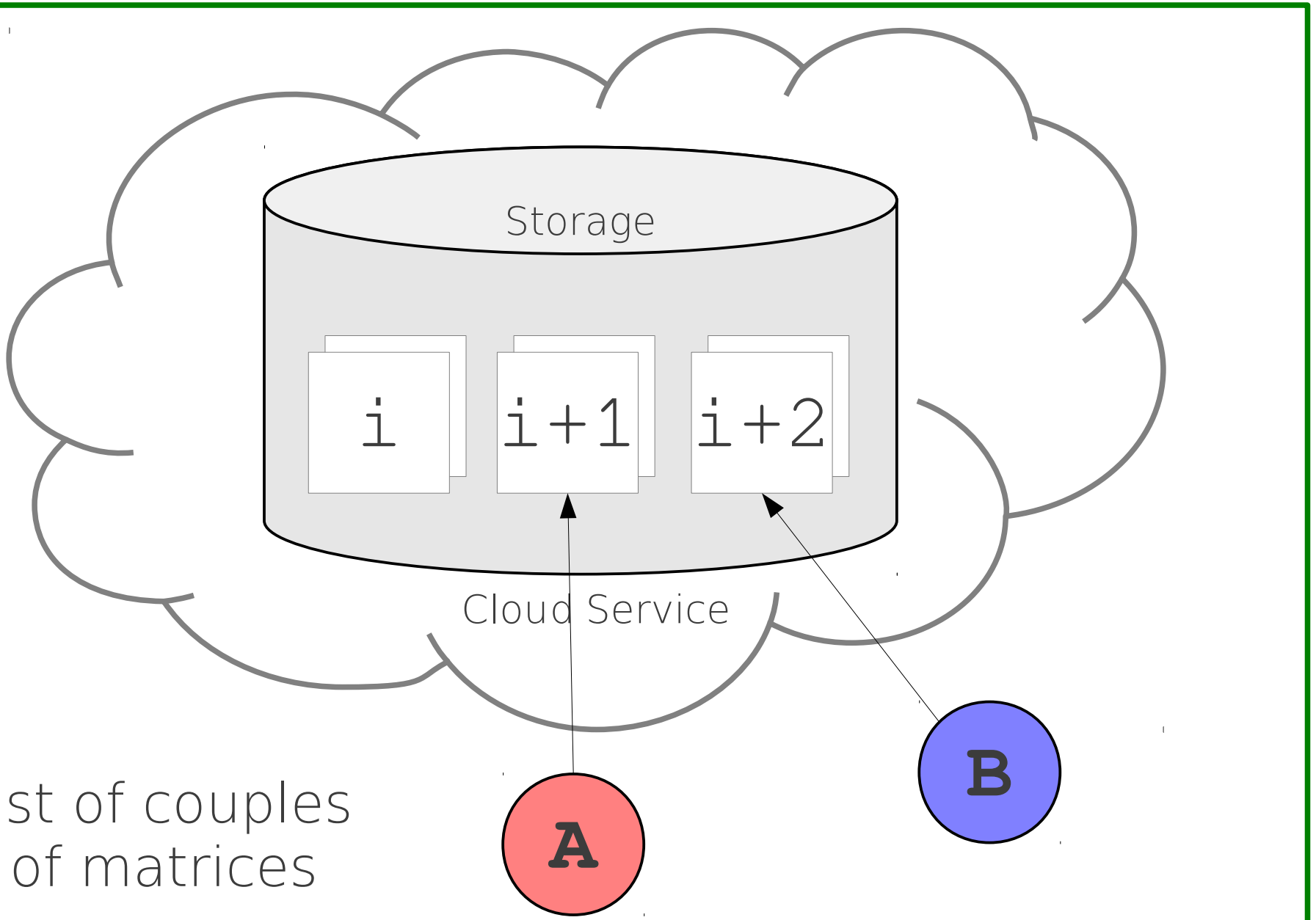
START INSERTION PROCEDURE

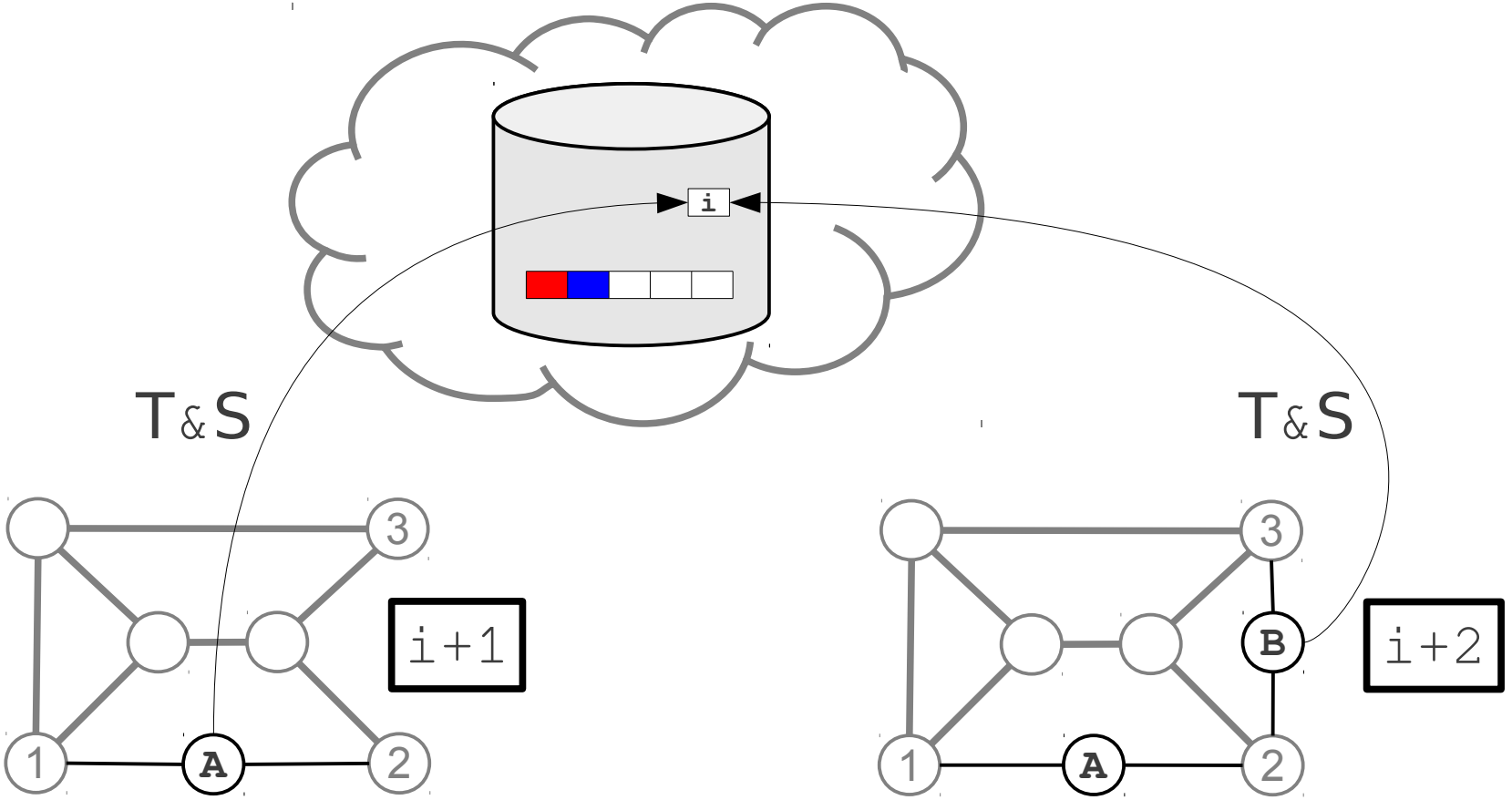
`< i+2, INS, B, {2,3} >`



START INSERTION PROCEDURE

PERFORM A's INSERTION
START INSERTION PROCEDURE


Storage
i
i+1
i+2
Cloud Service
list of couples
of matrices
A
B



- If B succedes, A omits the epoch update.
- As last step, the entry in Ops is deleted

- HSIENA is a hybrid system that complements the SIENA publish/subscribe system by adding the ability to self-reconfigure after brokers additions and removals.
- HSIENA has a novel design that mixes the classic SIENA's distributed architecture with a highly available cloud-based storage service that brokers use as a shared memory space they can rely-on to adapt at runtime the ENS application-level network without service disruption.
- We are implementing a prototype of HSIENA to test its behaviour under various realistic loads.
- Our purpose is to asses both its ability to support insertion and deletions while providing service continuity and to study the tradeoff existing between the level of service HSIENA can guarantee and the cost incurred for maintaining state information stored on a cloud service.

Conclusions and future works

[1] R. Baldoni, L. Querzoni, S. Tarkoma, A. Virgillito: “Distributed Event Routing in Publish/Subscribe Communication Systems”. Springer (2009)

[2] A. Carzaniga, D. Rosenblum, A. Wolf: “Design and evaluation of a wide area notification service”. TOCS (2001)

[3] A. Carzaniga, M.J. Rutherford, A. Wolf: “A routing scheme for content-based networking”. INFOCOM (2004)

[4] Z. Jerzak, C. Fetzer: “Soft state in publish/subscribe”. DEBS (2009)

[5] G. Cugola, D. Frey, A.L. Murphy, G.P. Picco: “Minimizing the Reconfiguration Overhead in Content-Based Publish-Subscribe”. SAC (2004)

[6] A. Montresor, L. Abeni: “Cloudy weather for P2P, with a chance of gossip”. P2P (2011)

References