

# IN5020 - Distributed Systems Group Session

## Topic 6 – Communication Paradigms

Praveensankar Manimaran

Praveema@ifi.uio.no

October 6, 2023

## Question 1 -

what is the difference between transient and persistent storage in communication paradigms?

# Transient vs Persistent Storage

## **Transient Storage:**

- The messages are not stored in an intermediate-term storage (may be stored in temporary buffers)
- It doesn't provide time decoupling (both senders and receivers should be active for the communication to take place)
- Eg:- Sockets, MPI

## **Persistent storage:**

- The messages are stored in middlewares (or some permanent storage)
- It enables time decoupling (senders and receivers can send and receive messages asynchronously)
- Eg:- Message Oriented Middleware, Pub-Sub System

## Question 2 -

What is the responsibility of the Message Brokers in the Message Oriented Middleware paradigm?

# Message Brokers

1. Handle semantics conversion:
  1. Message brokers convert incoming messages so that they can be understood by the destination application
2. Act as matching process
  1. In publish-Subscribe systems, the publishers might be publishing on certain topic 'X' and the Subscribers those are interested in 'X' can be matched with the corresponding publisher

## Question 3 -

What are the different states of nodes in epidemic protocol?

# Node Stages

## 1. **Infected**

1. The node holds data and willing to spread the data to other nodes

## 2. **Susceptible**

1. The node that hasn't yet seen the data

## 3. **Removed**

1. The node that has seen the data and not willing the spread the data to other nodes

## Question 4 -

When searching for files in an unstructured peer-to-peer system, it may help to restrict the search to nodes that have files similar to yours. Explain how gossiping can help to find those nodes



# Gossip to find nodes having similar files

1. During gossiping, **nodes** exchange **membership information**
2. Nodes will **eventually** get to know about **all other nodes** in the system.
3. Each time it **discovers** a **new node**, it can be evaluated with respect to its semantic proximity, for example, by counting the number of files in common.
4. The semantically nearest nodes are then selected for submitting a search query.



Thank You