



Services

and Applications

Boris Koldehofe
Universität Stuttgart

Background



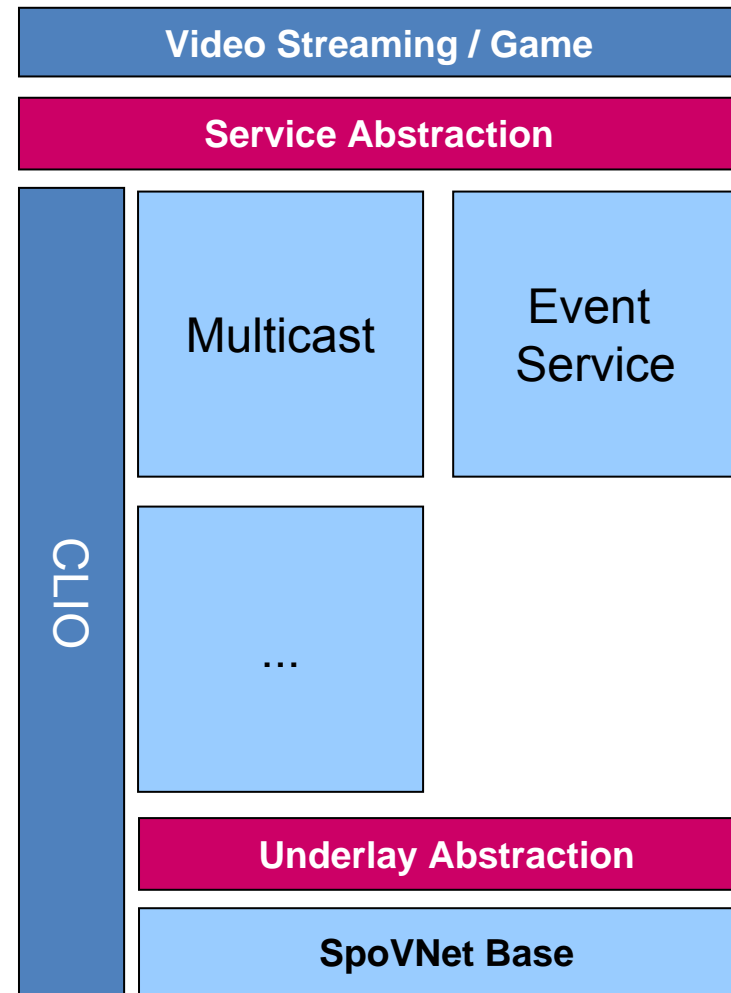
- Characteristics of applications for the NGI
 - large number of cooperating users
 - interact with many heterogeneous devices
 - operate on heterogeneity of devices and networks
 - provide resources in time and on demand
 - robustness
- Services facilitate the deployment of application
 - provide inherent properties
 - scalability by decentralized organization
 - requirements negotiated via QoS
- Objective of SpoVNet: Services should help the development of NGI applications

Outline



How does SpoVNet address requirements of NGI applications?

- Introduce applications + services
- Demonstrate how services provided by SpoVNet match the application requirements
- Show how the SpoVNet architecture supports such services
- Conclusion



Overview: Applications in SpoVNet



- **Objective:** Selected two challenging applications with respect to NGI requirements
 - Game
 - real-time interactions
 - multi-user interactions
 - Video Streaming
 - Bandwidth, latency and jitter requirements
- **General requirements**
 - many to many communication
 - scalable organization
 - adaptive to dynamic set of users
 - adaptive to network conditions

Overview: Services in SpoVNet



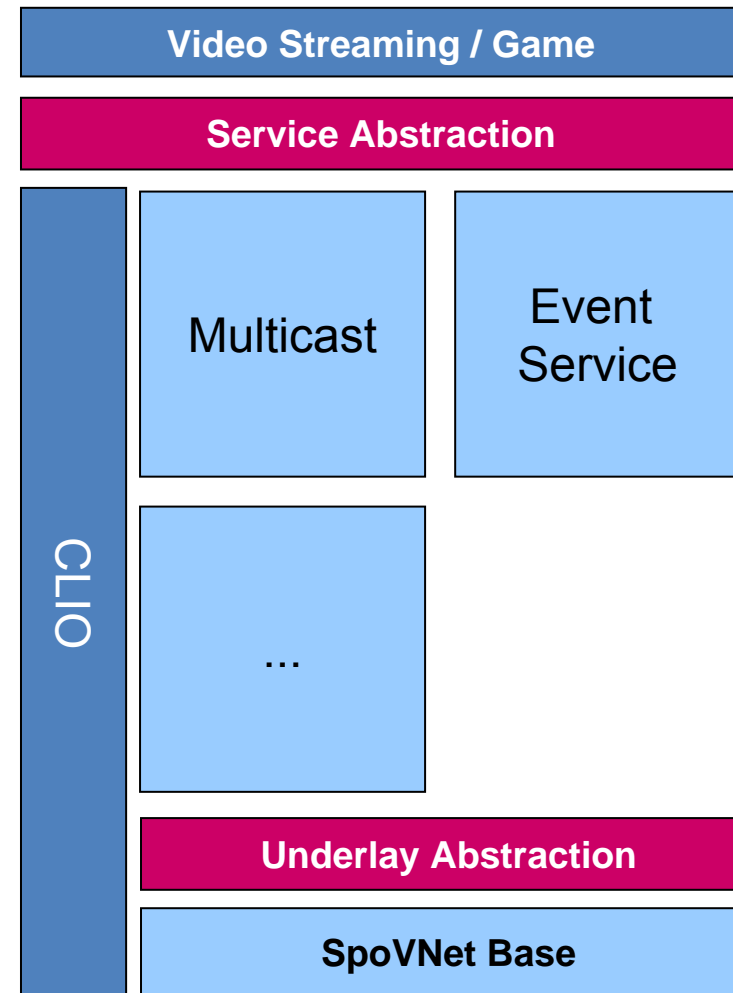
- Two exemplary services
 - Multicast
 - group management
 - dissemination to all group members
 - Event service
 - selective event dissemination
 - correlation of events
- Characteristics addressed in SpoVNet
 - Application can specify requirements in terms of QoS
 - e.g. latency, bandwidth, ...
 - data related quality
 - Security
 - Efficient adaptation
 - based on overlay organization and cross layer optimization
 - follows application requirements and network and device properties

Outline



How does SpoVNet address requirements of NGI applications?

- Introduce applications + services
- Demonstrate how SpoVNET services match application requirements
- Show the value of SpoVNet's architecture in supporting such services
- Conclusion

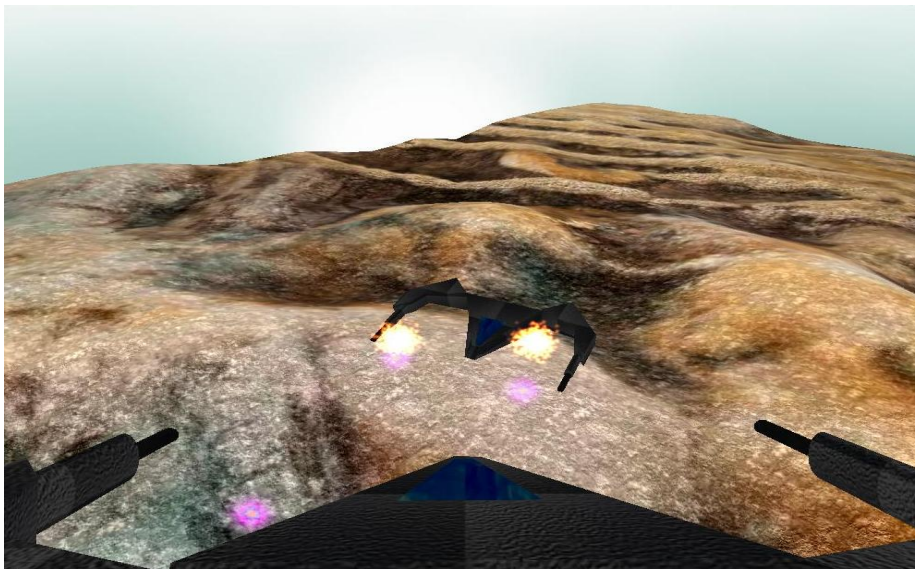


Multiplayer Online Game



Planet Π4:

- *Massively Multiplayer Online Game* (MMOG)
- highly demanding with respect to real-time capabilities
- Approach: Peer-to-peer architecture



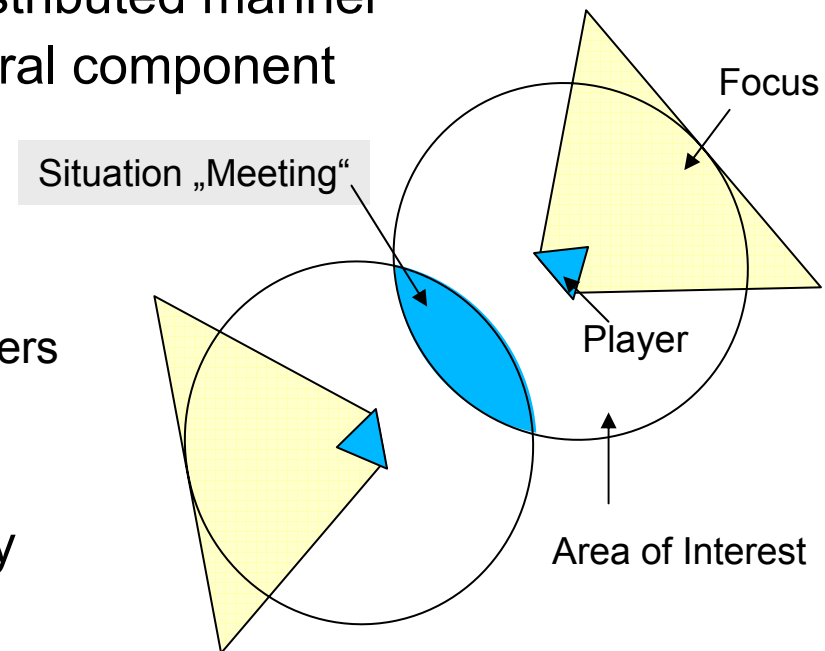
Supports

- Large number of players
- fast movement,
- collision detection of objects

MMOG Communication



- Peer-to-Peer architecture
 - Game state is managed in a distributed manner
 - Situation detection without central component
 - Low-latency communication
- Main Challenges
 - Responsiveness
 - Equal relative latency for all players
 - Consistency
 - Equal view for all players
 - Reliability, Security, Persistency
- Solution's required properties
 - Scalable for Massively Multiplayer Online Game
 - Capable of the Responsiveness-Consistency trade-off despite dynamicity



Multiplayer Online Game



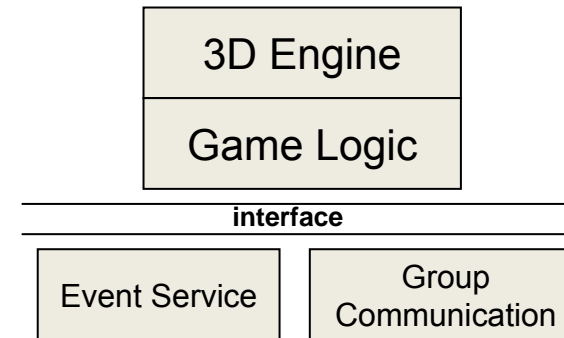
Solution:

- Reduce communication to small groups, e.g. by Area of Interest (AoI)
- signal QoS requirements, e.g. responsiveness
- Use SpoVNet Services for group selection and group communication

Group selection via the Event Service

1. Define complex events
 - movements of avatars
 - meetings and departures
 - e.g. $\text{dist}(x,y) < 50 \Rightarrow$ meeting
2. Subscribe to events of interest
 - e.g. players in my area of interest
3. Publish continuously game coordinates

Game architecture:



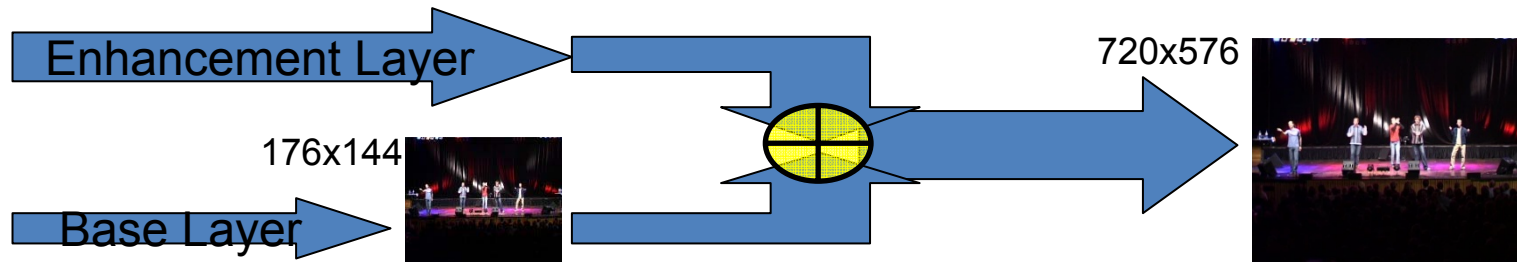
Group Communication: MCPO

1. state exchange within an area of interest
2. Manage multicast group for an area of interest
 - responsiveness determines QoS requirements

Video Streaming



- Goal: Adaptive to the characteristics of end devices and to their network path.
- Approach:
 - Adapt the video in size (spatial), in framerate (temporal) and in quality (SignalNoiseRatio)
 - Based on the new H.264 Scalable Video Coding (SVC)
 - Adjustment of enhancement layers based on network and device properties.



Adaptation of Video Streams



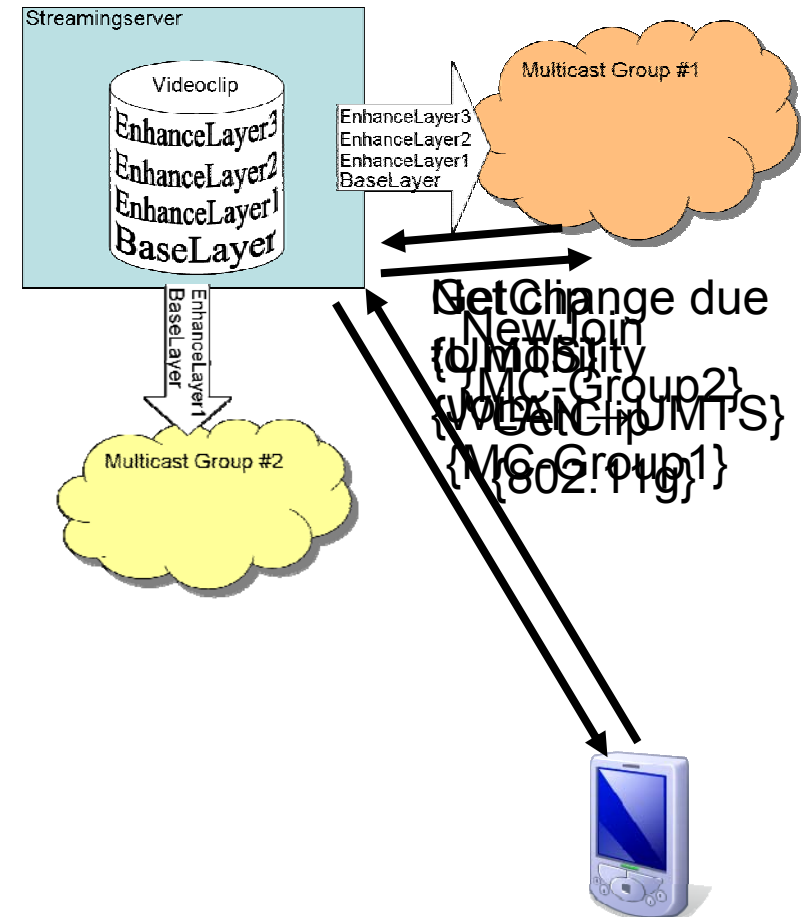
Two points to adapt the video stream:

- **Server centric: Collect feedback**
 - from the end device and
 - from its network path
- **Along the path adaptation**
 - performed by intelligent network device bridging fast and a slower network
 - adapt the stream if the packets use a hierarchical payload

Interactions with SpoVNET services



- Cross Layer Information (CLIO) determines
 - end device, its network connection, and its network path.
- Event Service
 - aggregation and delivery of information along the network path.
- Multicast Overlay (MCP-O)
 - delivers the resulting video stream to multiple end devices.

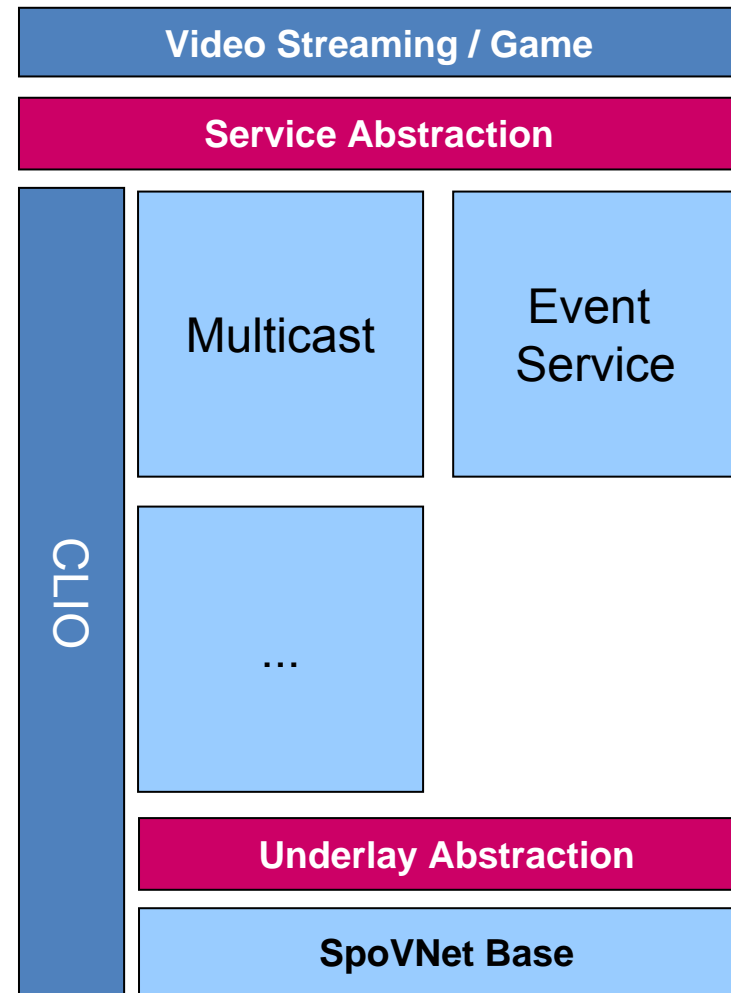


Outline



How does SpoVNet address requirements of NGI applications?

- Introduce applications + services
- Demonstrate how SpoVNET services match application requirements
- Show the value of SpoVNet's architecture in supporting such services
- Conclusion



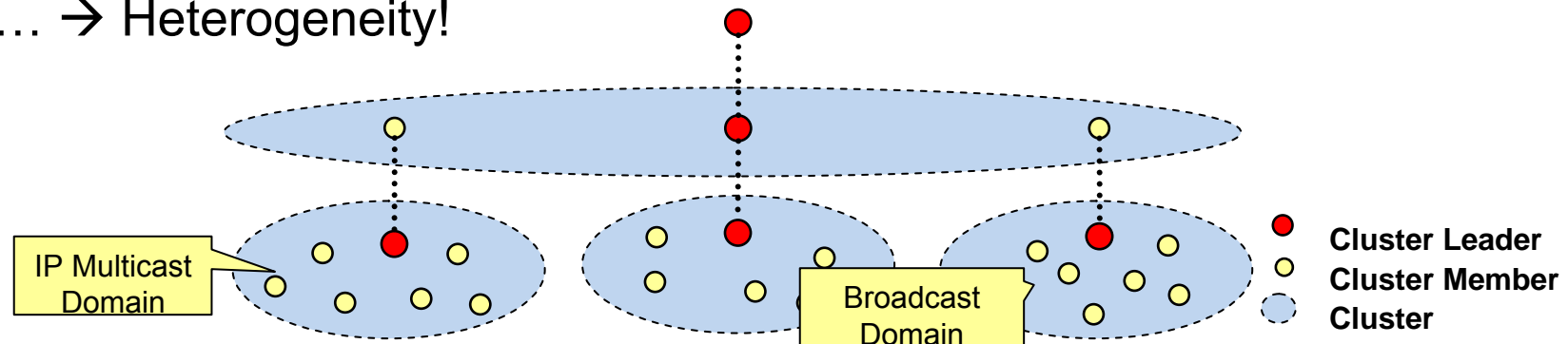
MCP-O: QoS-Enabled Multicast



- Achieve scalability through **hierarchical clustering**
→ (Limit protocol overhead)
- Logical structure determines data dissemination
- Use **Service Metric** to affect building of structure:

$$d : K \times K \rightarrow \mathbb{R}_{\geq 0}$$
$$d : (x, y) \mapsto \sum_i a_i f_i(x, y)$$

- Weights a_i determined through application requirements
- $f_i(x, y)$ e.g. latency, same WiFi domain, same IP Multicast domain, ... → Heterogeneity!



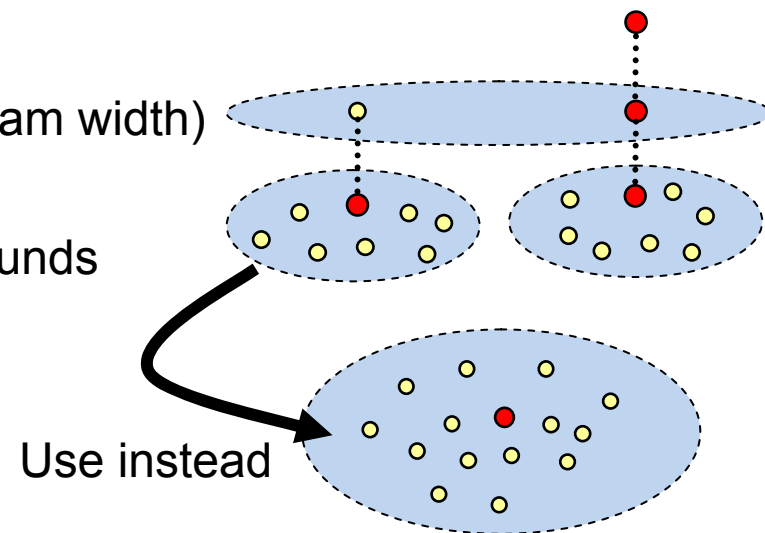
Adaptivity of MCP-O Hierarchy



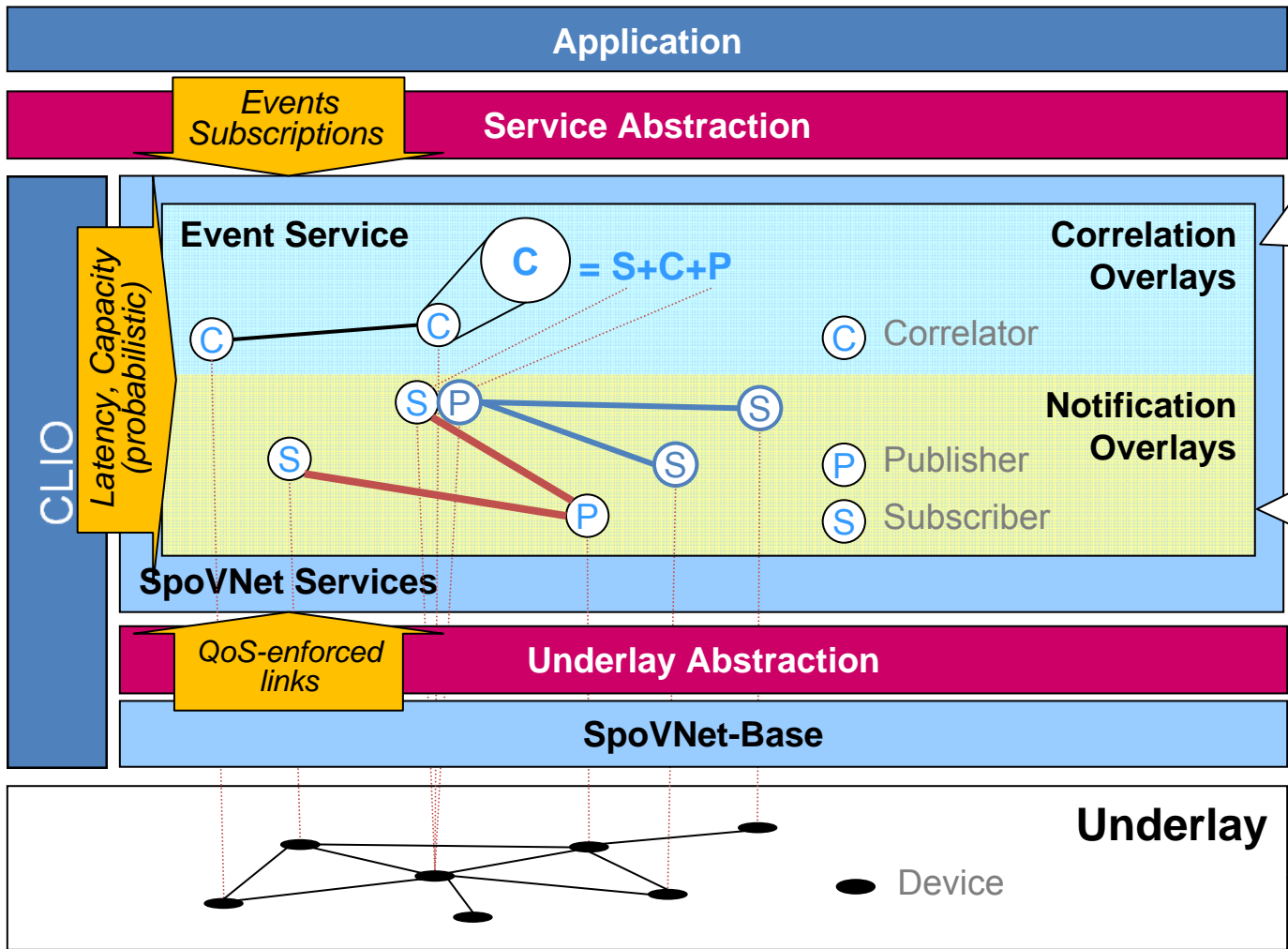
- Hierarchical approach not always the best choice
 - Good scalability, but cutting back e.g. latencies ☺
- Example: Latency-sensitive app with few members
 1. Application describes its demands and data stream width (Priority on Latency)
 2. CLIO provides upstream bandwidth per member
 3. Adapt cluster size K

→ $K = (\text{lowest current upstream}) / (\text{data stream width})$

Unicast brings best achievable latency,
protocol decides reasonable overhead in bounds



Event Service



Dynamicity
 - Deployment
 - Migration
QoS
 - Data and correlation quality

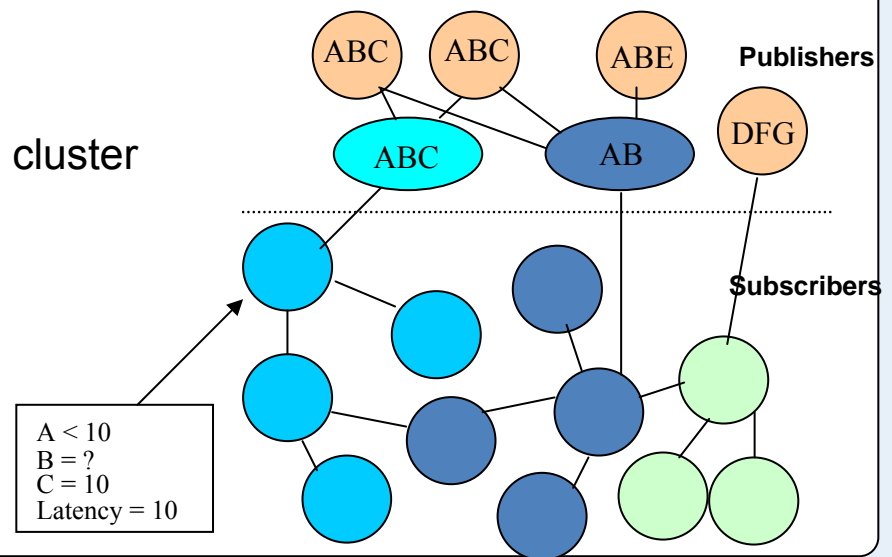
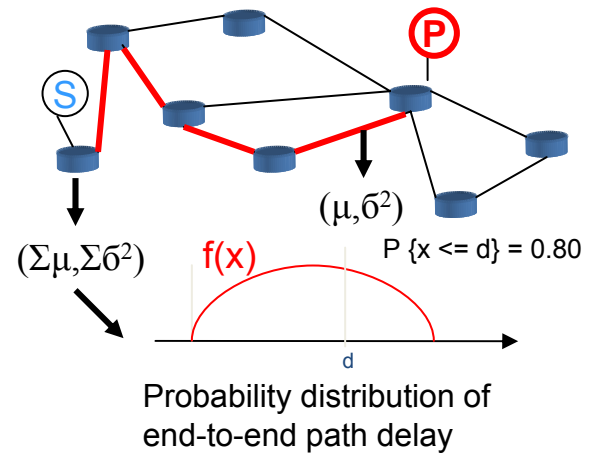
Dynamicity & QoS
 - Clustering,
 Restructuring

Heterogeneity
 - Networks
 - Devices

Event Notification



- Publish/Subscribe paradigm
 - Publication is a named set of attributes e.g. {A,B,C}
 - Subscription is a filter e.g. {A = 10 ^ B = 5}
- QoS metrics
 - Latency : probabilistic promises
 - Capacity: max amount of data/time that a node can forward
- Hybrid approach
 - Interest based subscriber clustering
 - Content-based forwarding within each cluster
- Overlay adaptation
 - Publisher centric
 - Partitioning of event space
 - Subscription centric
 - Fulfilment of QoS requirements.



Conclusions



- SpoVNet provides an extensible set of services
 - facilitate the deployment and development of applications
- Services match many requirements of NGI applications
 - scalability
 - support for QoS
 - effective adaptation
- SpoVNet architecture of value for the integration of services
 - Some services may become part of the NGI
 - Seamless transition of services
 - When possible services use reservation protocols
 - Otherwise QoS typically reflects what is currently possible

Questions

