

# Client Operation for RELOAD

[vidyan@qualcomm.com](mailto:vidyan@qualcomm.com)

draft-vidya-p2psip-eclients-00

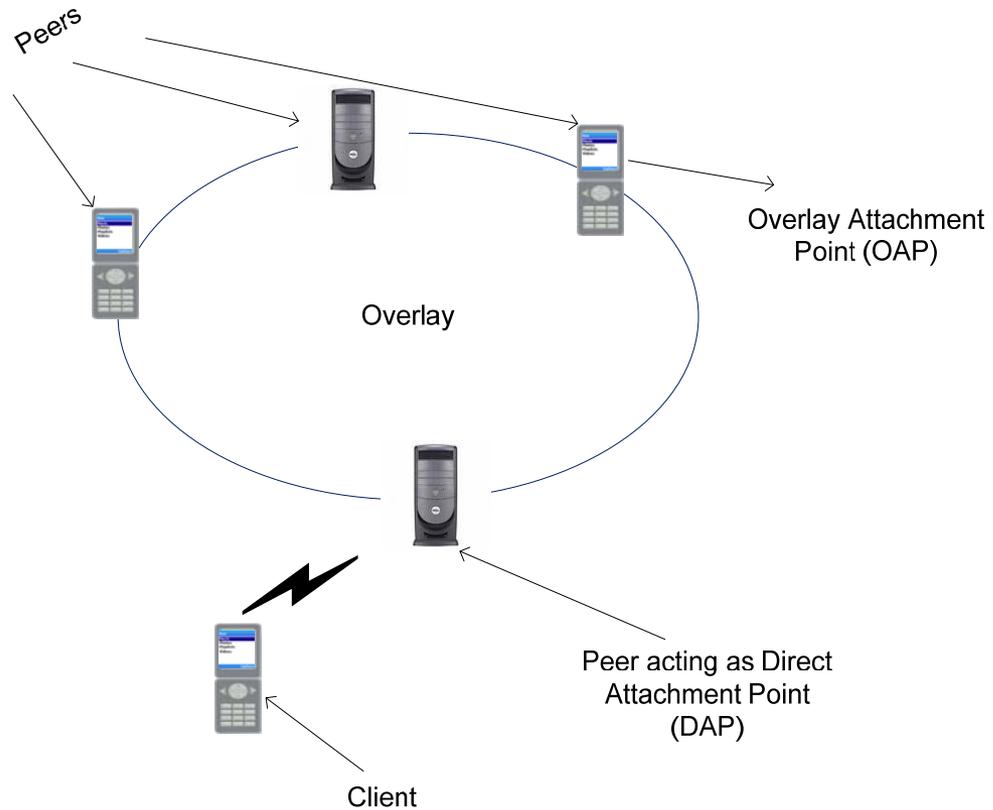
# Background

- ▶ RELOAD supports the notion of clients that may participate in p2p overlays without taking part in routing or storage aspects of the overlay
  - A client may provide and receive services to/from the overlay
- ▶ The current design allows the client to attach to the overlay through the peer that owns the identity of the client in the identity space of the overlay
  - Even this basic operation requires more clarity in the base draft
- ▶ However, the client may not always be able to connect to the peer that owns its identity
  - NATs may prevent such connectivity
- ▶ Further, a proximity based connection may be more advantageous to a client
- ▶ RELOAD talks about attaching to an arbitrary peer, but requires the client to handle reachability using a destination list

# Using Destination List

- ▶ A destination list may be used for any unsolicited requests to be routed to the client
  - Note that all responses can be routed properly using symmetric routing
- ▶ Applications that need to include reachability information now should specify a destination list as part of the service contact information
- ▶ Potential issues
  - A node id or a destination list needs to be provided depending on the current state of attachment
  - Any time the client changes point of attachment or status (say, becomes a peer), the service information must be updated to indicate change of contact information
    - Depending on the volume of data stored in the overlay by the client, this can be bad
    - In general, an overlay level change affecting service level information is undesirable

# Client, DAP and OAP



# eClient Operation

- ▶ It is desirable to allow the client to use a node that is topologically close to it to reach the overlay (we call this node a Direct Attachment Point or DAP)
  - E.g., a phone may use a laptop connected over Bluetooth as its DAP
  - This also allows a client to use the same node as DAP for multiple overlays it is attached to, where possible
- ▶ The node that owns the identity of the client in the overlay is termed the OAP
- ▶ Packets sent from the client can be directly routed on the overlay from the DAP without any extra hops
- ▶ Requests sent to the client will end up at the OAP by default
  - Indirection based approach allows OAP to route to the client's DAP

# Join Process

- ▶ Client discovers the DAP via out-of-band means
- ▶ Client sends a Join Request message to the DAP
- ▶ DAP routes the join to the OAP (regular join routing)
- ▶ OAP notices DAP ID and creates state corresponding to the client's ID with the DAP as the next hop
- ▶ OAP sends Join Response back to the client through the DAP
- ▶ OAP sets up a direct connection to the DAP
  - Routing via the overlay is an option if a direct connection is not possible
- ▶ Optionally, authorization may be needed to use a node as a DAP

# Message Routing

- ▶ Client is always reachable natively on the overlay
- ▶ All messages meant for the client will reach the OAP
- ▶ OAP will find DAP as the next hop in its DAP table
- ▶ OAP should send the message to the DAP
- ▶ DAP delivers message to client
- ▶ OAP does not incur an additional cost due to the connection with DAP
  - It is in lieu of the direct connection with the client itself

# Failures, Handoffs

- ▶ When the client attaches to a different DAP or OAP itself or becomes a peer on the overlay:
  - The client must re-join the overlay
  - No impact to any service level data stored on the overlay
- ▶ DAP or OAP leaves/failures need to be handled
  - Procedures defined in the draft – some scenarios result in client having to re-join; some other scenarios are more seamless