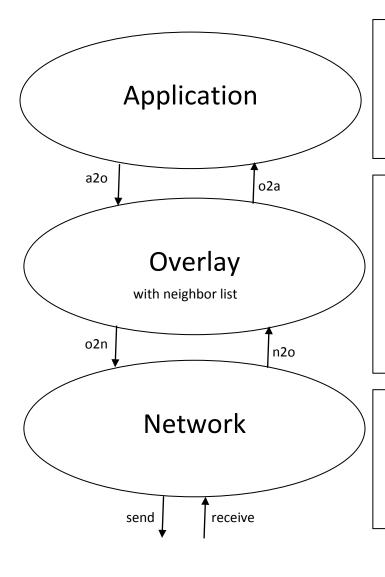# Software Specification „Folder Sync"

3-Layer-Architecture

Application

- periodically sending file list to neighbors
- sending request for file x from neighbor y
- sending file x to neighbor y

a2o

o2a

Overlay

with neighbor list

- receive ping and react with pong for peer exploration
- {build-up connection}
- {terminate connection}
- refresh neighbor list (delete neighbors with expired timeouts)
- analyze messages from application and outreach them to the network

o2n

n2o

Network

- send messages
- receive messages
- file transfer

send   receive

Message specification:

Ping

incoming ping (n2o) := ("ping", pingID, ttl, hops, senderUsername, senderIP, senderPort)

outgoing ping (o2n) := ("ping", pingID, ttl, hops, ownUsername, ownIP, targetIP, targetPort)

Pong

incoming pong (n2o) := ("pong", id, [(username1, ip1), (username2, ip2), ...])

outgoing pong (o2n) := ("pong", id, [(username1, ipP1), (username2, ip2), ...], targetIP, targetPort)

RefFL (refresh filelist)

incoming refFL (n2o) := ("refFL", fileList, senderUsername, senderIP, senderPort)

incoming refFL (o2a) := ("refFL", fileList, senderUsername, urgentFlag)

outgoing refFL (a2o) := ("refFL", fileList)

outgoing refFL (o2n) := ("refFL", fileList, ownUsername, targetIP, targetPort)

reqFile (request file)

incoming reqFile (n2o) := ("reqFile", fileName, fileHash, senderIP, senderPort)

incoming reqFile (o2a) := ("reqFile", fileName, fileHash, senderUsername)

outgoing reqFile (a2o) := ("reqFile", fileName, fileHash, targetUsername)

outgoing reqFile (o2n) := ("reqFile", fileName, fileHash, targetIP, targetPort)

answerReq (answer request)

incoming answerReq (n2o) := ("answerReq", fileName, senderIP, senderPort)

incoming answerReq (o2a) := ("answerReq", fileName, senderUsername, senderPort)

outgoing answerReq (a2o) := ("answerReq", filePath, targetUsername, targetPort)

outgoing answerReq (o2n) := ("answerReq", filePath, targetIP, targetPort)