

Object Group Support

Implementing Message Groups in FlexiNet

Dave Otway



Java Multicasting experiments

- MulticastSocket extends DatagramSocket
 - they have different IP addresses (unicast + multicast), but share the same UDP port
 - packets sent from a MulticastSocket arrive with the sender's DatagramSocket IP address
- packets sent to 2 MulticastSockets with the same address on the same machine arrive at both
- packets sent to 2 DatagramSockets with the same address on the same machine arrive at only one

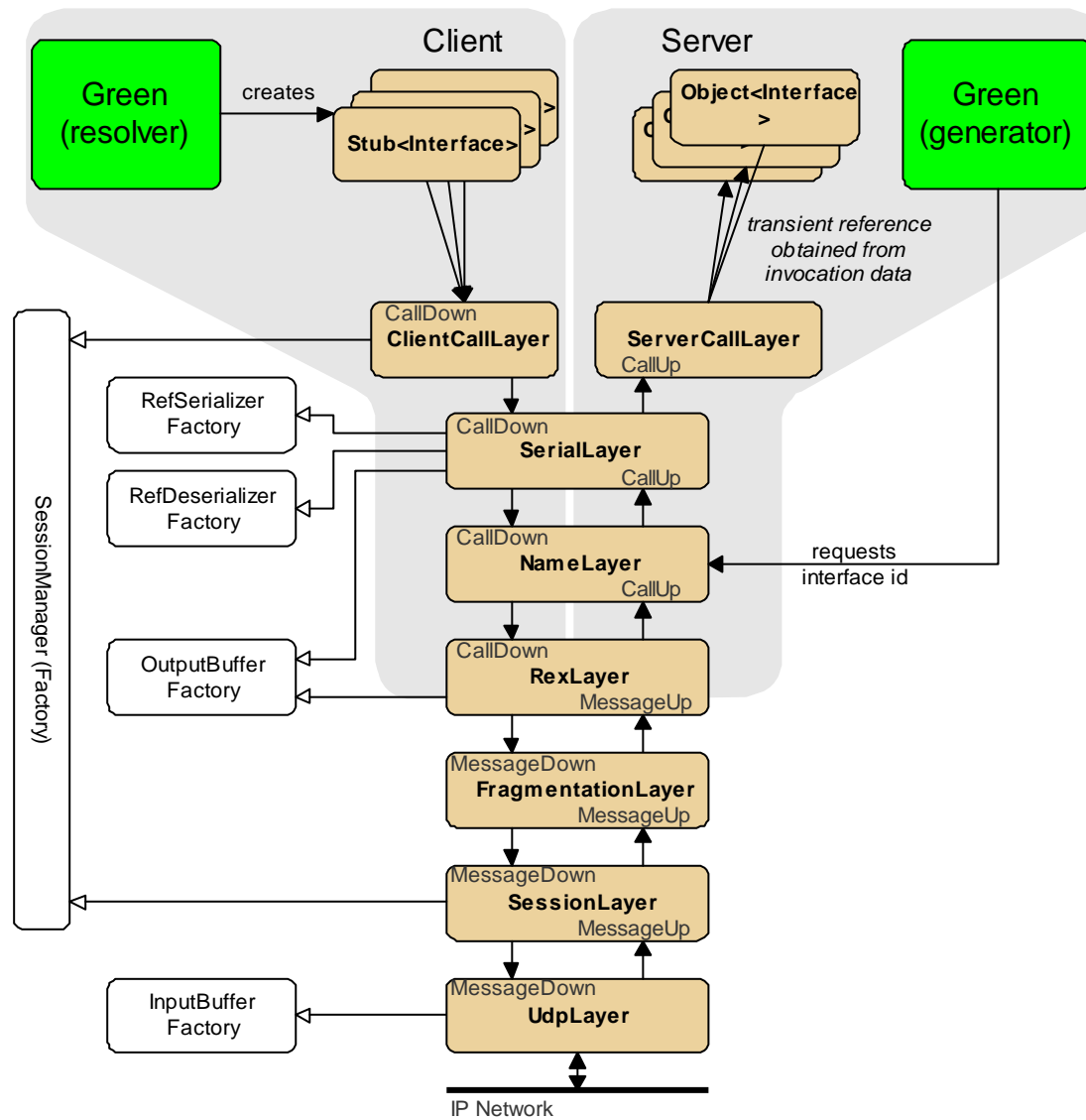


Consequence

- if more than one member of a group per machine:
 - each member must have a different port number
 - must store port numbers in the membership list
 - must include sender's port number in the packets
- current implementation assumes all members have the same port number
 - doesn't store it in membership list
 - doesn't include it in packets



REX Stack

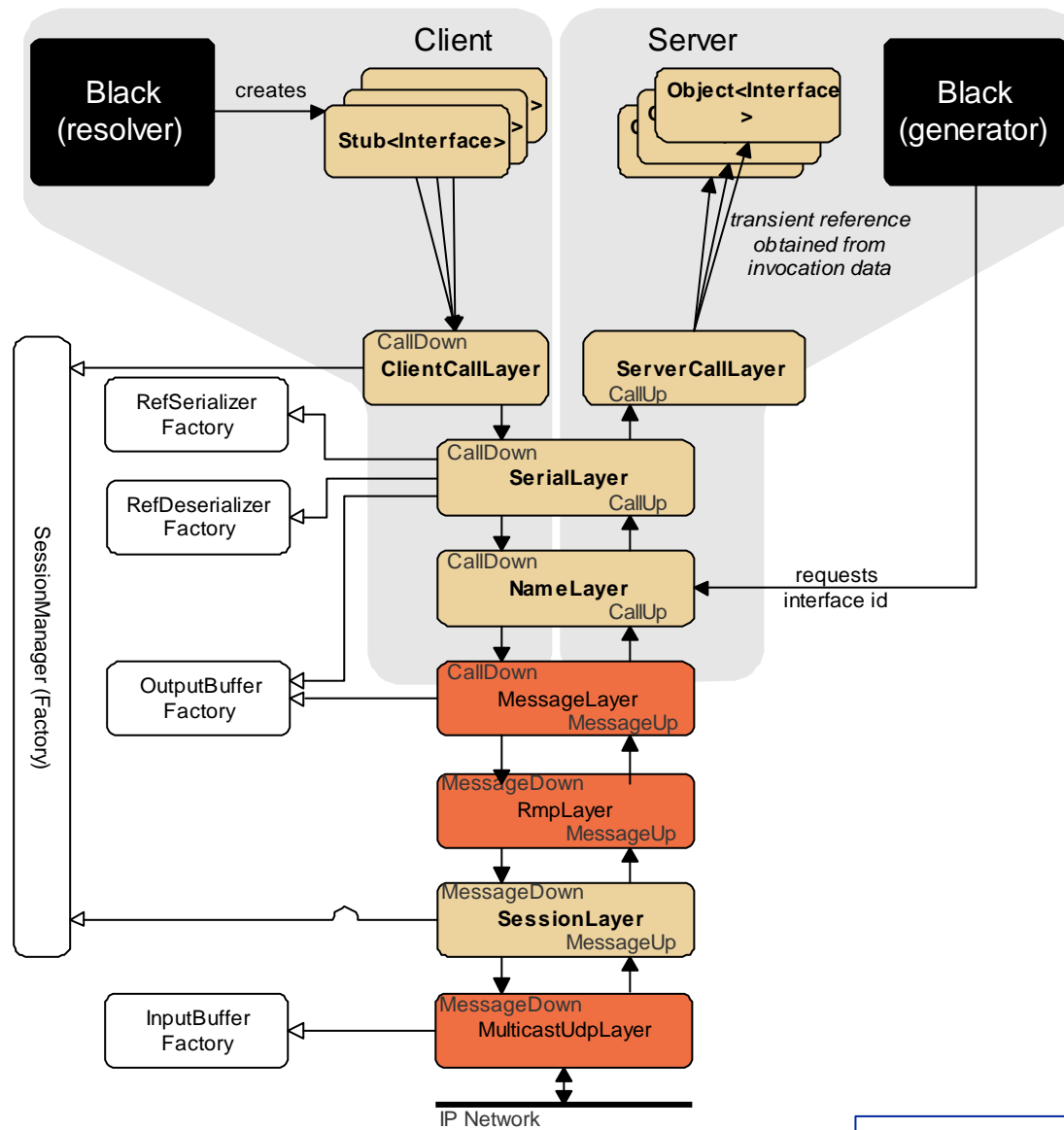


Multicast REX experiments

- extended UdpLayer to MulticastUdpLayer
 - servers worked OK
 - client received multiple replies from unknown addresses
 - client invoked itself
- suppressed address checking in SessionManager & developed Black binder by editing Green
 - client successfully invoked 2 servers and itself
- confident that most of REX stack could be reused



1st RMP Stack

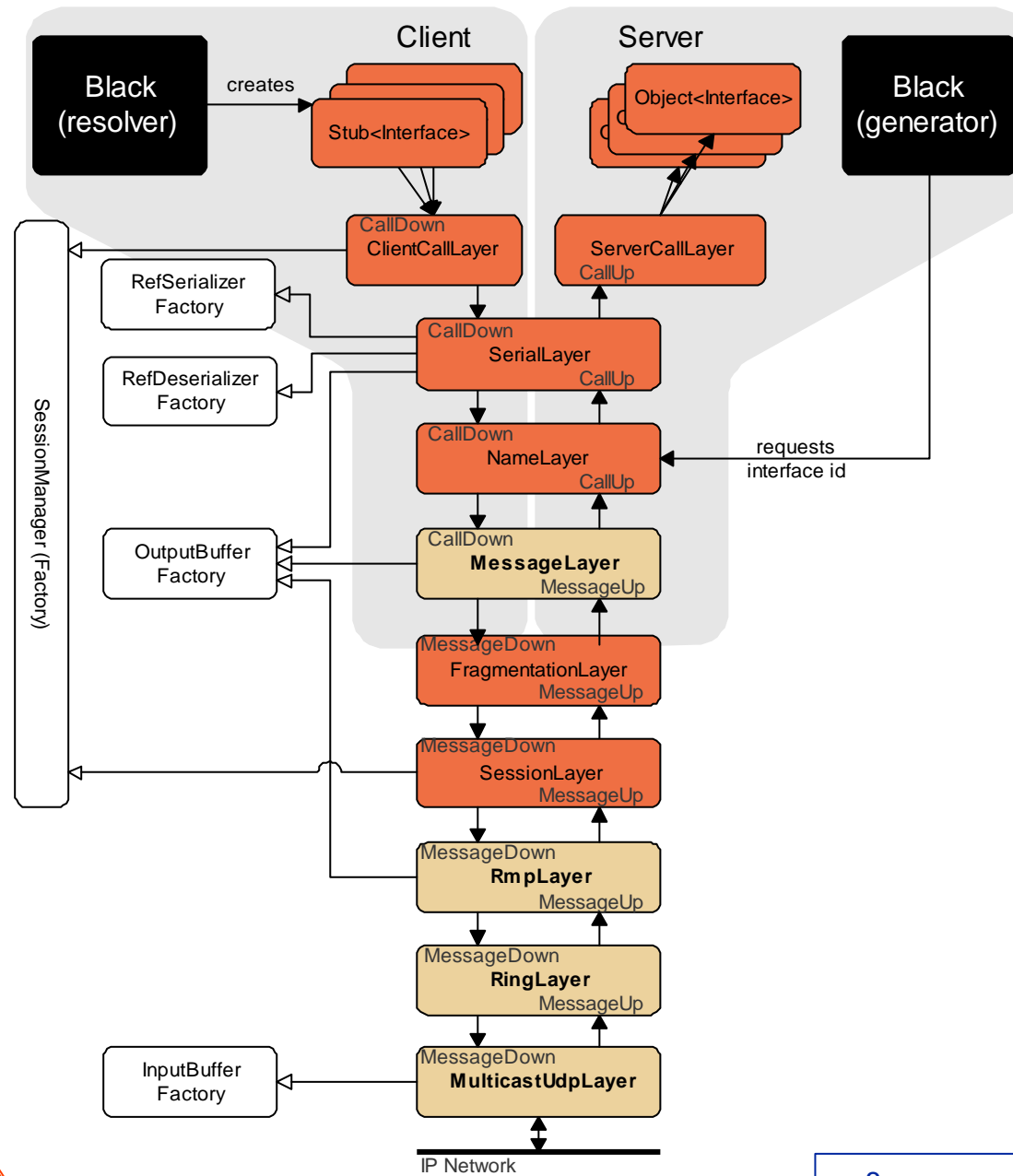


Session Experiments

- assumed that RMP membership list = REX session
- discovered (the hard way) that this is NOT true
 - sequence numbers skip places
 - addresses keep changing
- and still need the REX Session to pair request replies for non-member calls
- therefore needed another Session object (Ring)



Final RMP Stack

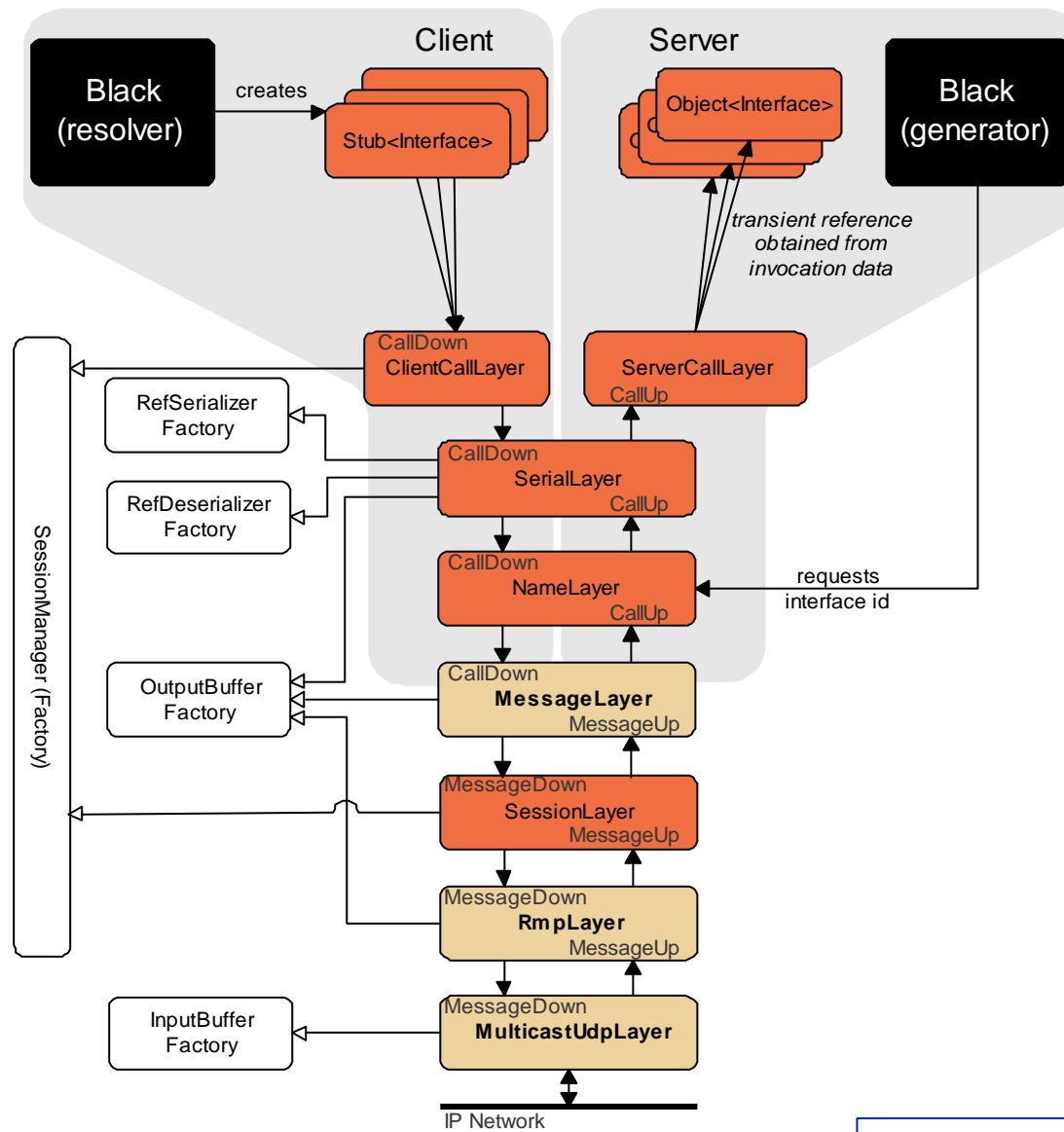


REX -> RMP

- Stub<interface>, Object<interface>, ClientCallLayer, ServerCallLayer, SerialLayer, NameLayer, FragmentationLayer & SessionLayer are reused
- Black is a minor edit of Green
- MessageLayer is a cut down RexLayer
- MulticastUdpLayer is a minor extension to UdpLayer
- RmpLayer & RingLayer are new



Current RMP Stack



Status

- Basic protocol, Join & NAKs working

To Do List

- Leave
- Reformation
- Buffer recycling
- Queue pruning
- Thread continuation
- Delivery serialization
- Token timeout
- Resilience
- Multi-NAKs
- Multi-ACKs
- Multiple rings
- Non member invocations

