

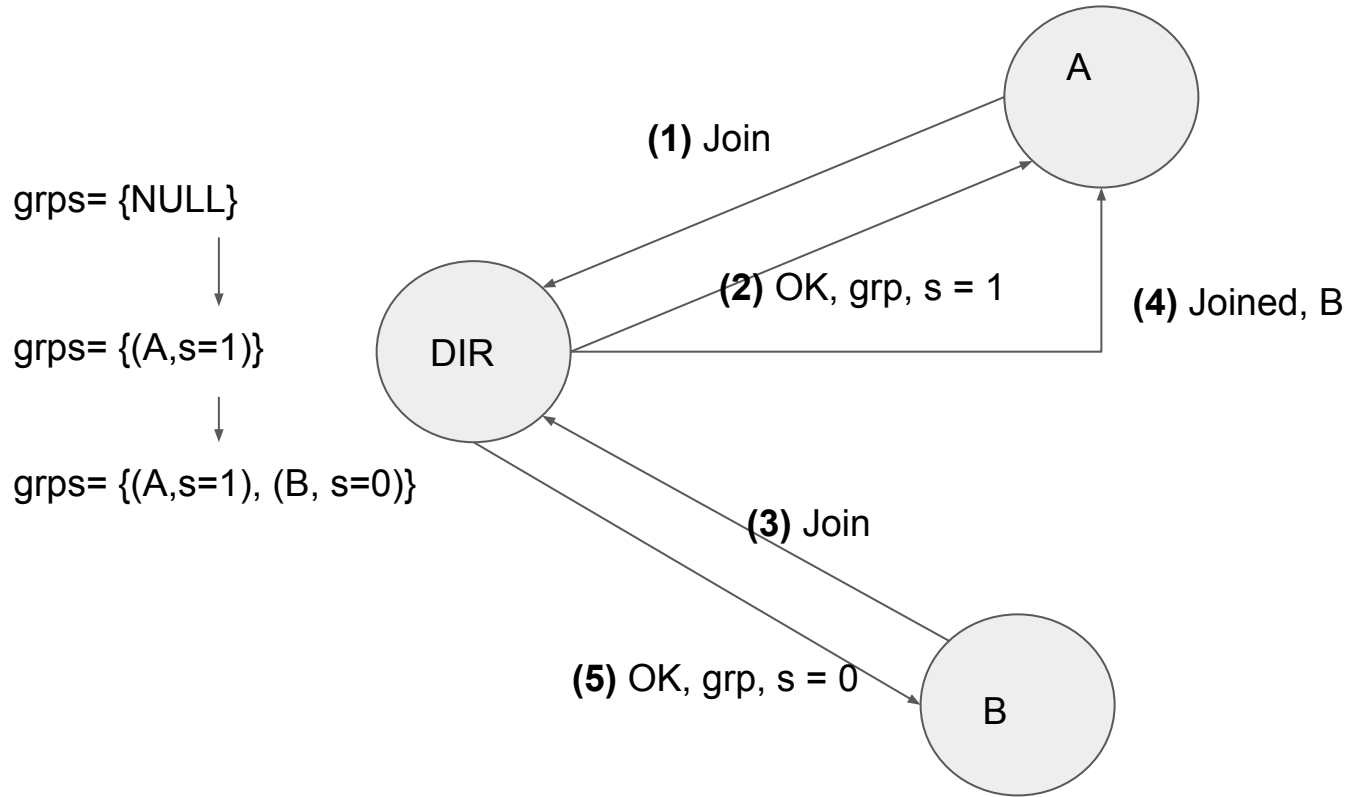
# Κατανεμημένα συστήματα

2η εργασία: Ομαδική επικοινωνία  
(Ημ.παράδοσης:31/3/2018)

Νικητάκης Παναγιώτης(1717)  
Ράντου Καλλιόπη(2004)

- Τρέχοντας το πρόγραμμα του directory(dir.py), παίρνουμε το ip και το port του στα οποία θα γίνει η TCP σύνδεση των εφαρμογών.
- Η κάθε εφαρμογή που έρχεται, συνδέεται πρώτα με το directory.
- Το directory ενημερώνει την ήδη υπάρχουσα ομάδα ή δημιουργεί καινούρια και στέλνει μήνυμα επιβεβαίωσης στην τρέχουσα εφαρμογή.
- Η εφαρμογή συνδέεται στο συγκεκριμένο group και δημιουργεί udp-multicast σύνδεση.
- Μπορεί είτε να στείλει, είτε να παραλάβει μηνύματα μέσω αυτού του socket.
- Όταν φύγει από την ομάδα που ανήκει, ενημερώνει πρώτα το directory.
- Το directory ενημερώνει την ήδη υπάρχουσα ομάδα και στέλνει μήνυμα επιβεβαίωσης στην τρέχουσα εφαρμογή.
- Η εφαρμογή αποχωρεί, αφού τερματίσουν τα thread.

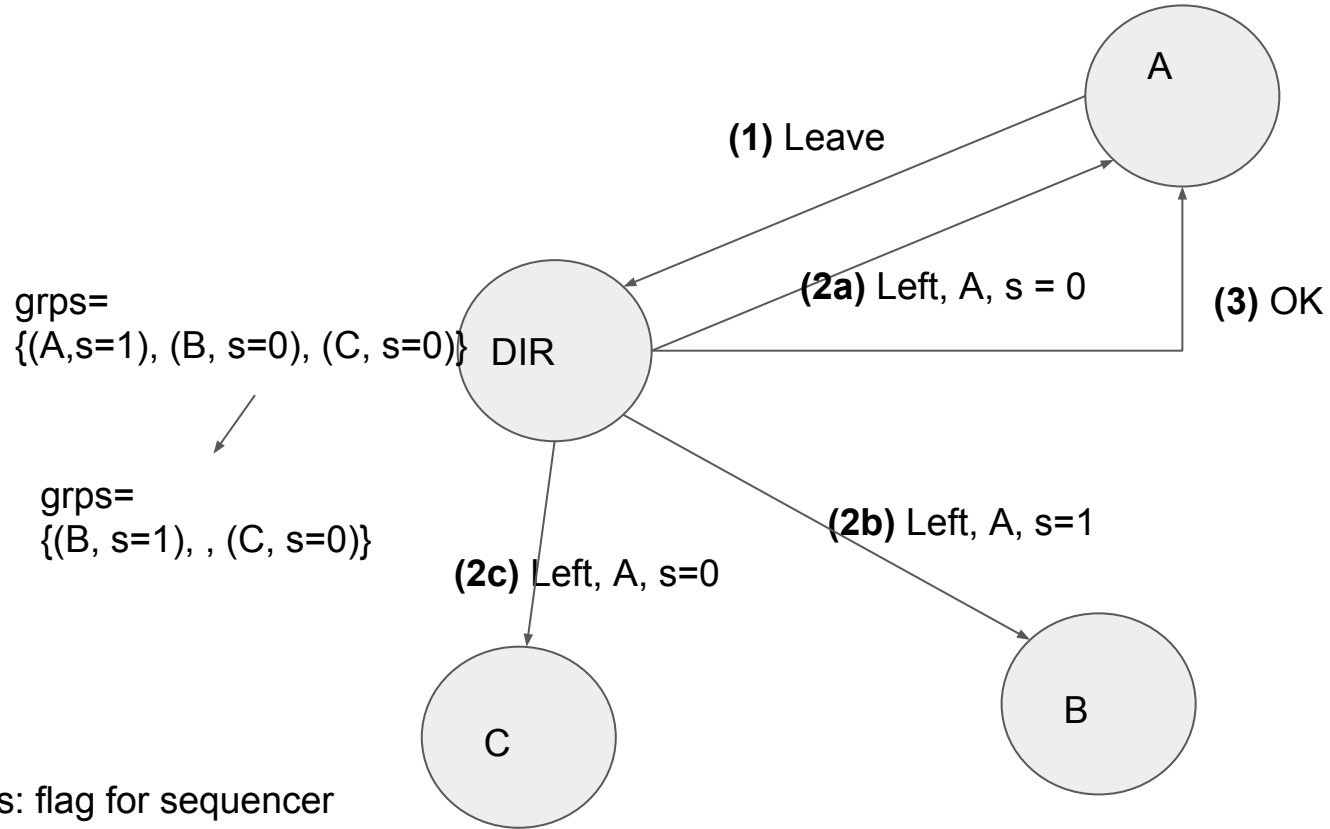
## Team management (Protocol): grp\_join()



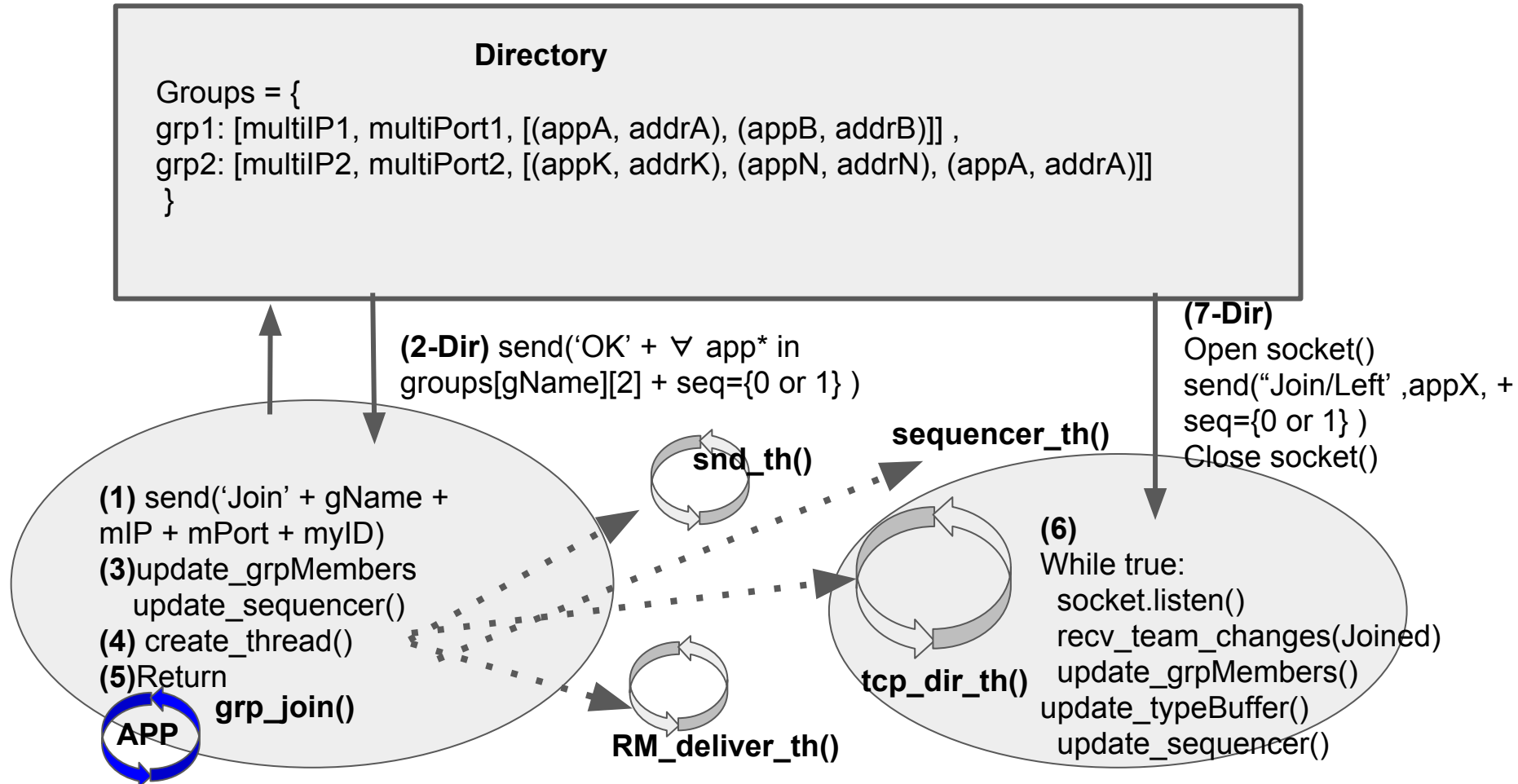
s: flag for sequencer

S TIMH = 0(initialize)

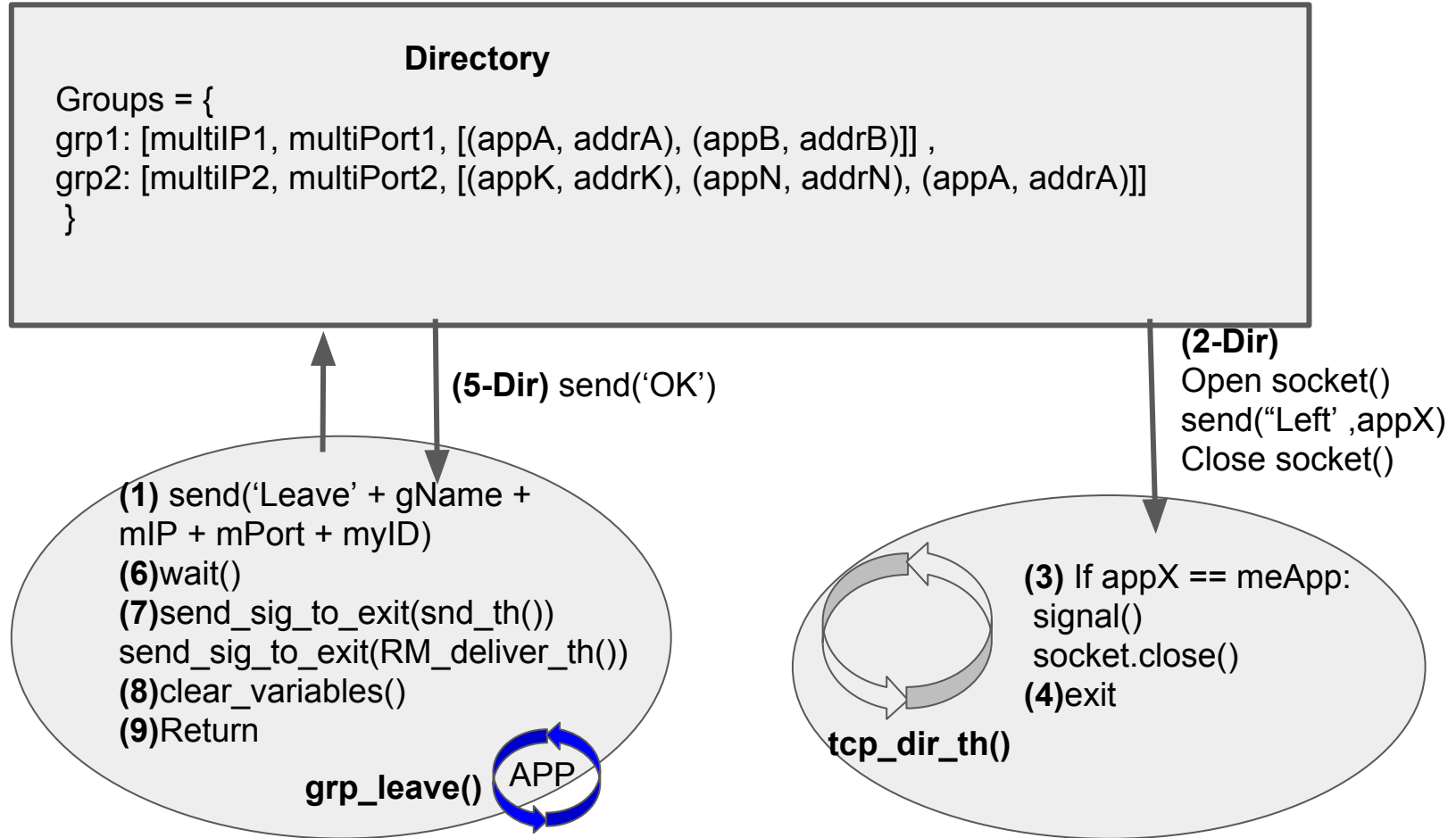
## Team management (Protocol): grp\_leave()



## Team management (Implementation): grp\_join()



## Team management (Implementation) : grp\_leave()



# Total order Multicast (Implementation)

(1) seqNo++

(2) qSender.put(msg, len, seqNo)

(3) return

grp\_send()

APP

sock = socket(multicast)

While True:

qMes.get()

packet =

"msg"+myID+seqNo+'D'

RM\_send(packet, sock)

snd\_th()

**RM\_send:**

While acks[] != grp :

sock.send(packet)

acks[] += sock.recv()

return

tcp\_dir\_th()

RM\_deliver\_th()

sock = socket(multicast)

While True:

packet = sock.rcv()

If packet.head == "msg":

sock.send(ACKM+myID, senderAddr)

#elegxos gia diplotypa

if packet not in msgbuffer:

msgbuffer += packet

typeBuffer.append('S')

if flagSequencer == 1:

packetNo++

p = "seq"+pID+pSeq+pNo

qSeq.put([pID, pSeq, pNo])

Else if data not in msgbuffer:

msgbuffer.packet.append(data)

else if packet.head == 'seq':

sock.send(ACKS+myID, senderAddr)

#elegxos gia diplotypa

If packet not in msgbuffer:

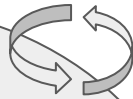
msgbuffer.append(packet)

typeBuffer.append('S')

Else:

msgbuffer.packet.append(packetNo)

### **sequencer\_thread()**



```
sock = socket(multicast)
```

```
While True:
```

```
    qSeq.get()
```

```
    packet = packID+packSeq+packNo
```

```
    RM_send(packet, sock, index)
```

### **group\_receive(socket,type,msg,length)**

```
msg = checkbuffer(index, mtype, msg, length)
```

```
Return msg
```

### **checkBuffer(socket,type,msg,length)**

```
If type == 'S'
```

```
    If S in typeBuffer: tempPos
```

```
    #get the smaller seq number
```

```
    Msg = messageBuffer[tempPos]
```

```
    Del messageBuffer[tempPos]
```

```
Elif type == 'D'
```

```
    If D in typeBuffer: tempPos
```

```
    Msg = messageBuffer[tempPos]
```

```
    Del messageBuffer[tempPos]
```

```
Else:
```

```
    msg="nomessage wth given type!"
```

```
return msg
```