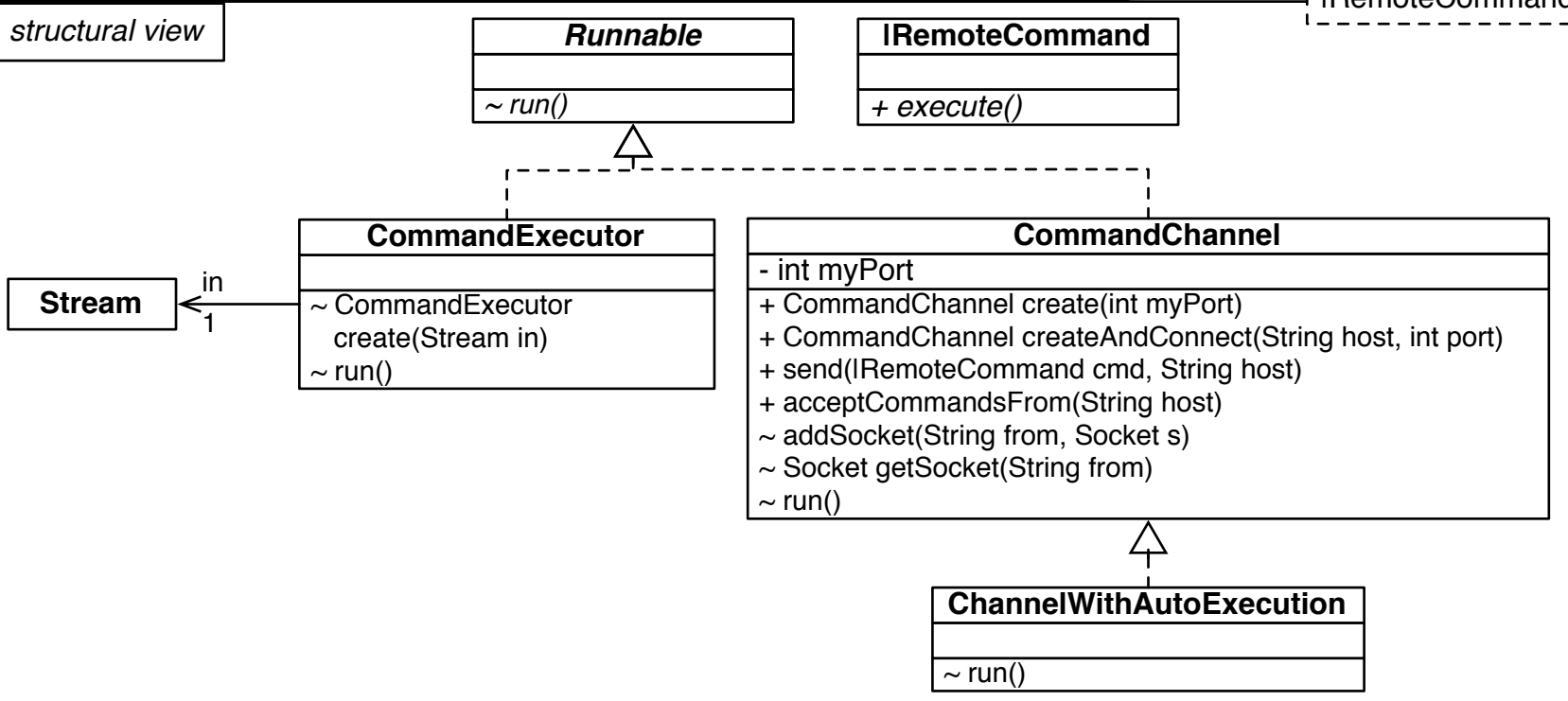


aspect NetworkCommand depends on Command, SocketCommunication, Map

IRemoteCommand

structural view



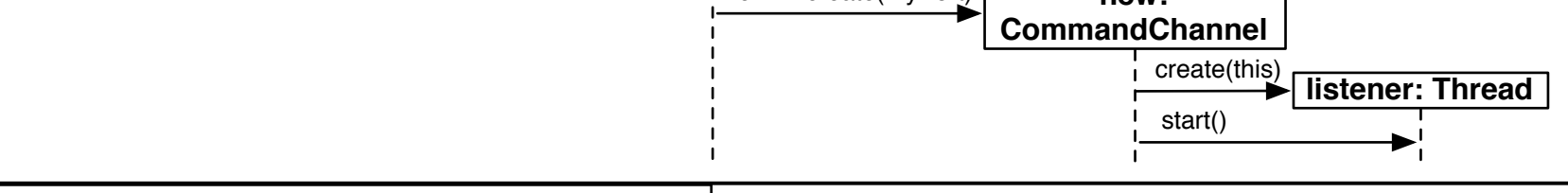
Distribution Roles:

- Listener:** depends on SocketCommunication.Listener
CommandChannel.create
- Initiator:** depends on SocketCommunication.Initiator
CommandChannel.createAndConnect
- Executor:** requires Listener or Initiator, depends on SocketCommunication.Receiver
acceptCommandsFrom, CommandExecutor, Stream, IRemoteCommand.lexecute
- AutoExecutor:** depends on SocketCommunication.Listener, SocketCommunication.Receiver
ChannelWithAutoExecution.create, CommandExecutor, Stream, IRemoteCommand.lexecute
- Commander:** requires Listener or Initiator, depends on SocketCommunication.Sender
send, IRemoteCommand.create

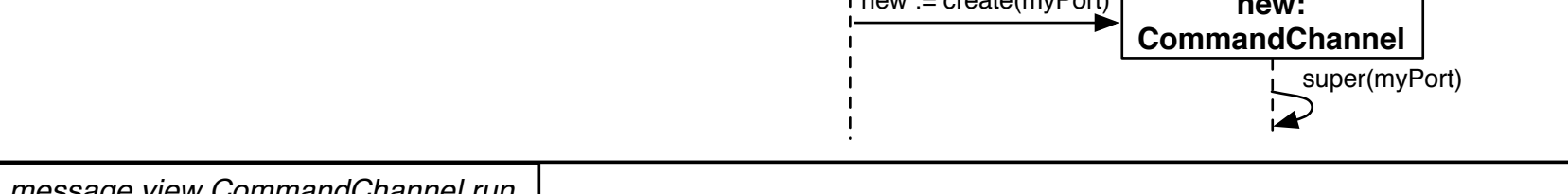
Instantiations:

- SocketCommunication: **ISendable** → **IRemoteCommand**; **InputStream** → **Stream**
- Command: **ICommand** → **IRemoteCommand**; **lexecute** → **execute**
- Map: **IData** → **CommandChannel**; **IKey** → **String**; **IValue** → **Socket**; **add** → **addSocket**; **getValue** → **getSocket**

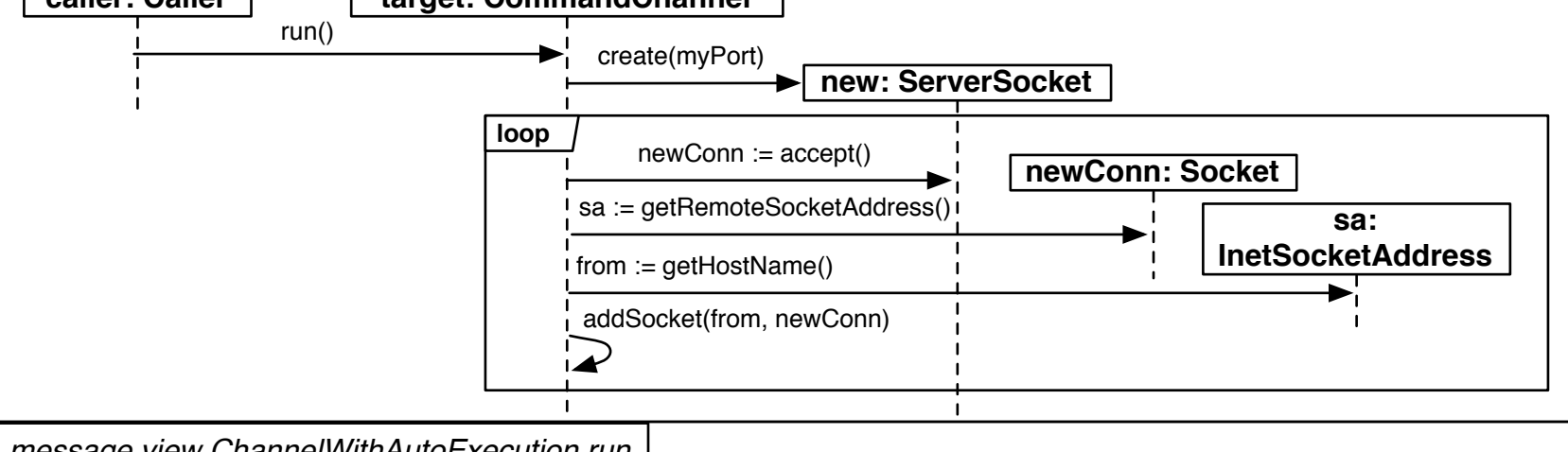
message view CommandChannel.create



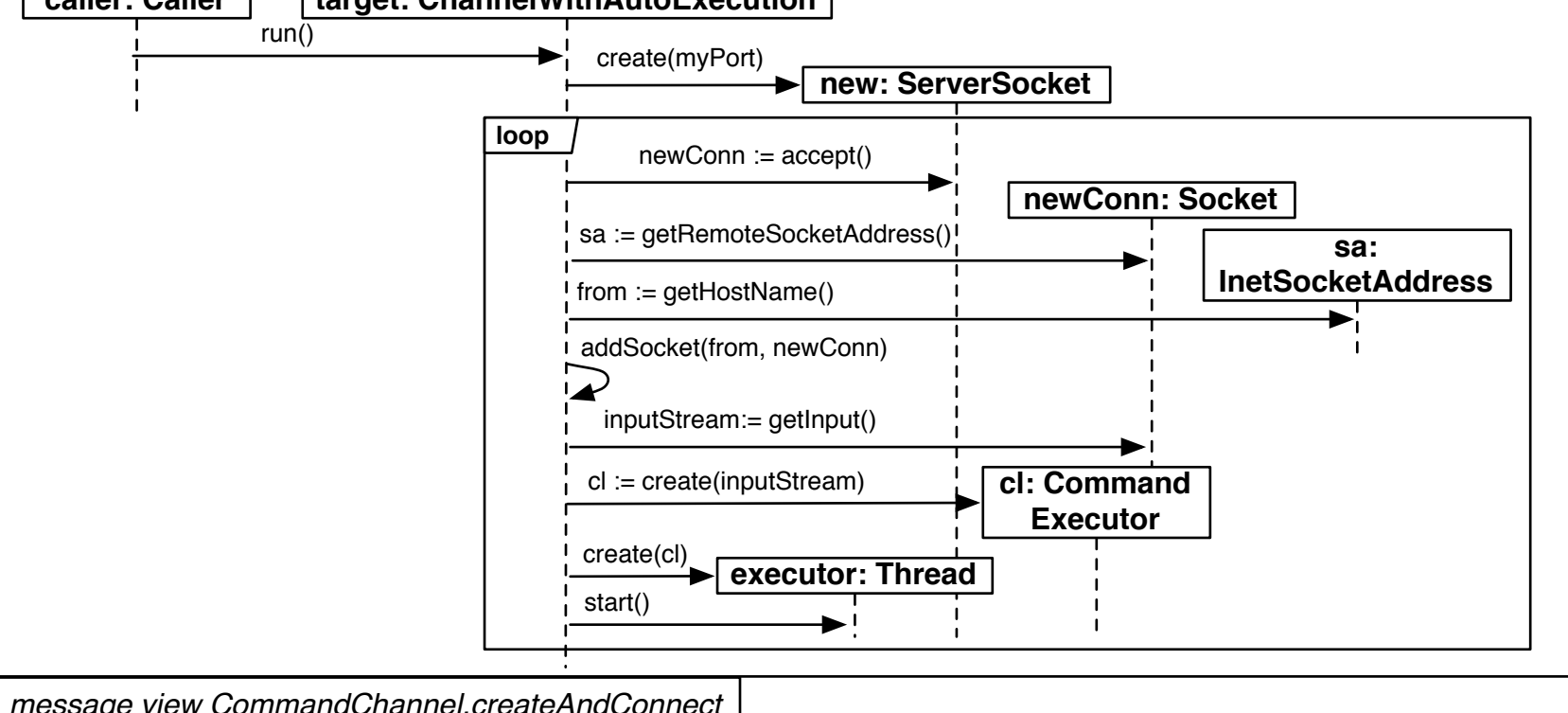
message view ChannelWithAutoExecution.create



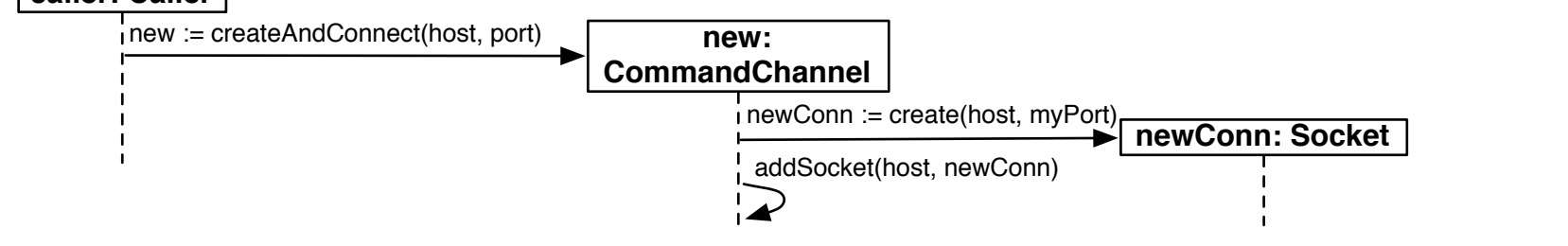
message view CommandChannel.run



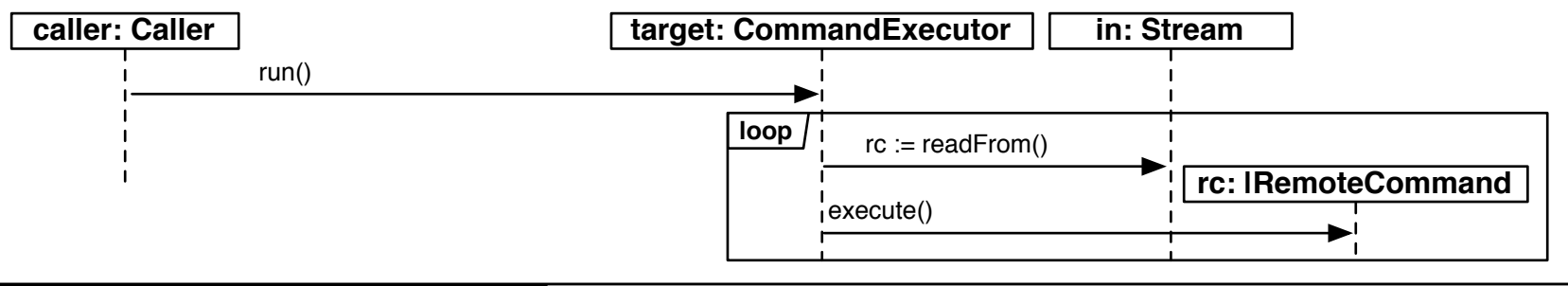
message view ChannelWithAutoExecution.run



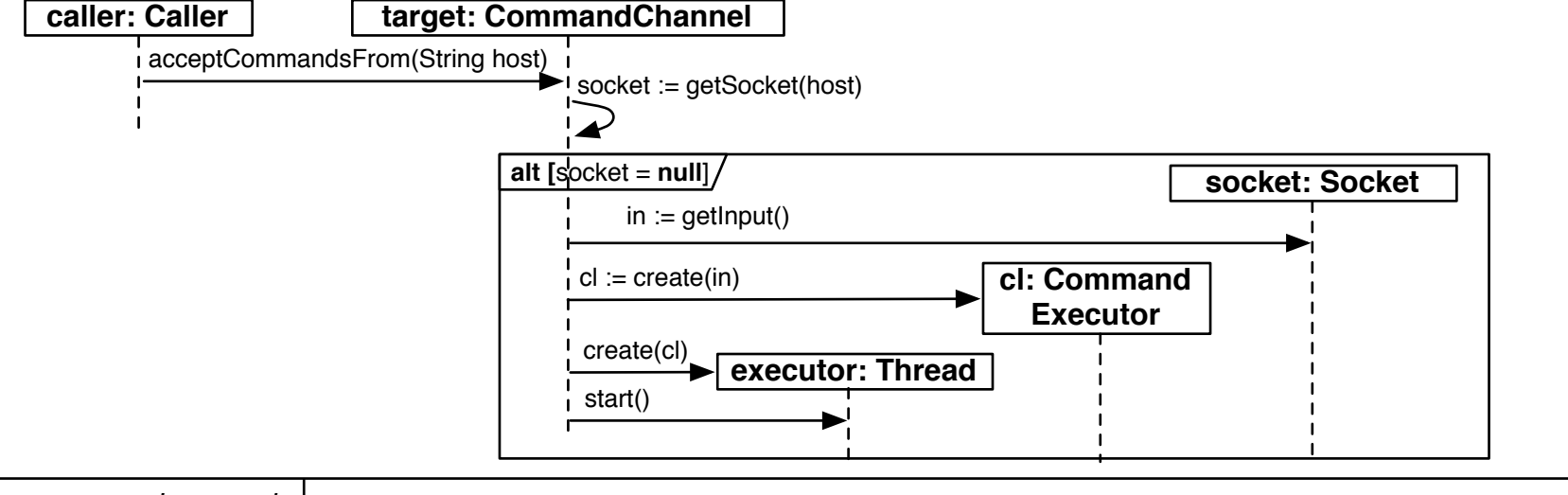
message view CommandChannel.createAndConnect



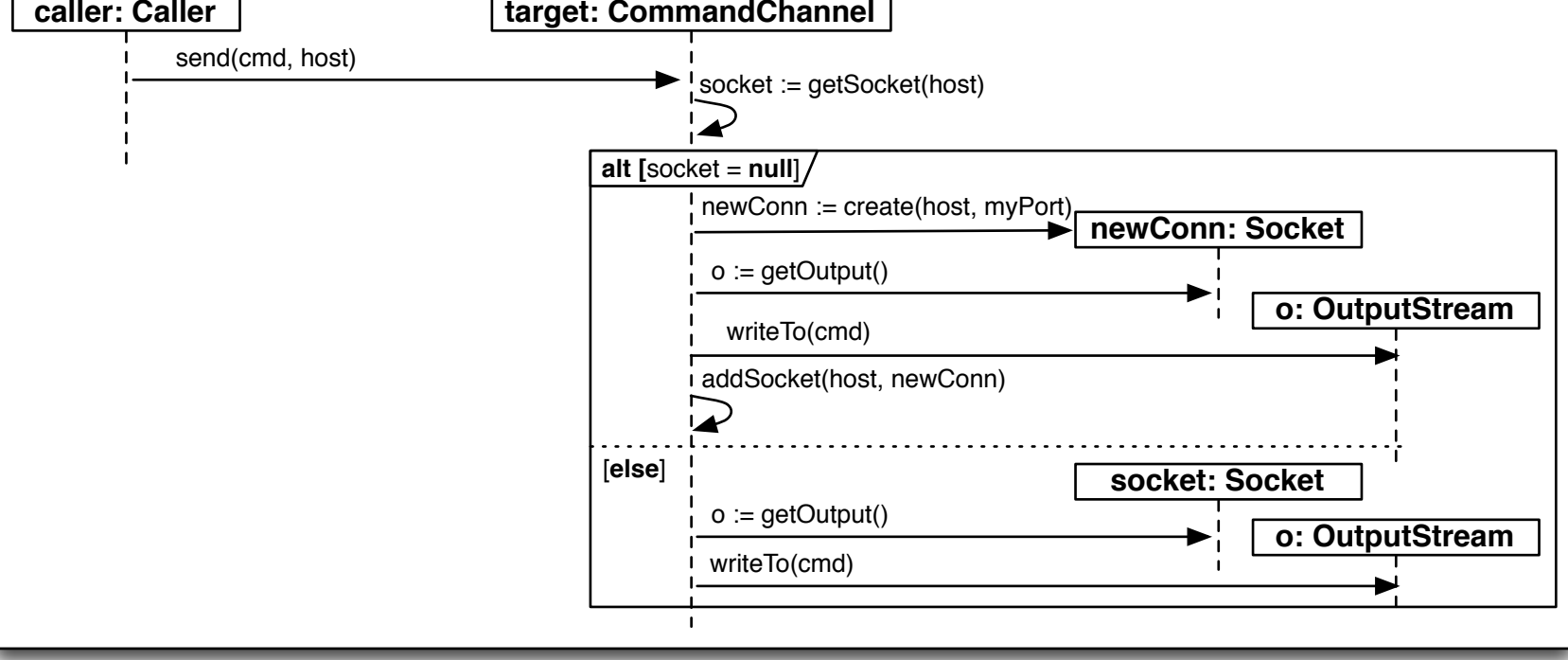
message view CommandExecutor.run



message view acceptCommandsFrom



message view send



aspect SocketCommunication

structural view

**<<interface>>
Serializable**

ISendable

ServerSocket

+ ServerSocket create(int port)
+ Socket accept()

Socket

+ Socket create(String host, int port)
+ close()
+ boolean isConnected()
+ SocketReceiver getReceiver()
+ SocketSender getSender()

myReceiver
0..1

SocketReceiver

+ ISendable receive()

mySender
0..1

SocketSender

+ send(ISendable s)

ISendable

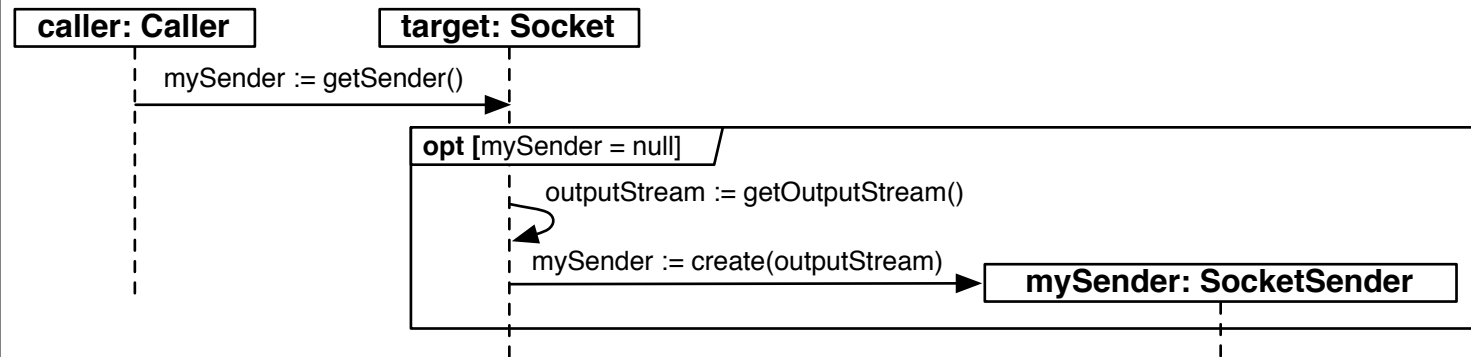
Distribution Roles:

Listener: ServerSocket, create, accept, Socket
Initiator: Socket, Socket.create, Socket.close
Sender: requires Listener or Initiator
Receiver: Socket, getSender, SocketSender, send, ISendable
Receiver: requires Listener or Initiator
Socket, getReceiver, SocketReceiver, receive, ISendable

Implementation:

Serializable: java.io.Serializable
 ServerSocket: java.net.ServerSocket
 Socket: java.net.Socket
 SocketSender: java.io.ObjectOutputStream
 send: writeObject
 SocketReceiver: java.io.ObjectInputStream
 receive: readObject

message view getSender



message view getReceiver

