```
1  ---------- MODULE CSComm ----------
   Specification of communication in a Client-Server system model.
5  EXTENDS SequenceUtils
6  ----------
7  CONSTANTS
8      Client,     the set of clients
9      Server,     the (unique) server
10     Msg         the set of possible messages
11 ----------
12 VARIABLES
13     cincoming,  cincoming[c]: incoming channel at client c ∈ Client
14     sincoming   incoming channel at the Server
15 ----------
16 TypeOK ≜
17     ∧  cincoming ∈ [Client → Seq(Msg)]
18     ∧  sincoming ∈ Seq(Msg)
19 ----------
20 Init ≜
21     ∧ cincoming = [c ∈ Client ↦ ⟨⟩]
22     ∧ sincoming = ⟨⟩

24 EmptyChannel ≜ Init
25 ----------
   A client sends a message msg to the Server.
29 CSend(msg) ≜
30     ∧ sincoming′ = Append(sincoming, msg)
31     ∧ UNCHANGED cincoming
   Client c receives a message from the Server.
35 CRev(c) ≜
36       ∧ cincoming[c] ≠ ⟨⟩
37       ∧ cincoming′ = [cincoming EXCEPT ![c] = Tail(@)]   consume a message
38       ∧ UNCHANGED sincoming
39 ----------
   SRev/SSend below is often used as a subaction. No UNCHANGED in their definitions.

   The Server receives a message.
47 SRev ≜
48     ∧ sincoming ≠ ⟨⟩
49     ∧ sincoming′ = Tail(sincoming)  consume a message
   The Server sends a message cmsg to each client other than c ∈ Client.
53 SSend(c, cmsg) ≜
54     ∧ cincoming′ = [cl ∈ Client ↦
55                       IF cl = c
56                        THEN cincoming[cl]
```

```
57                 ELSE  Append(cincoming[cl], cmsg[cl])]
```

The *Server* broadcasts the same message *msg* to all Clients other than $c \in Client$.

```
61  SSendSame(c, msg) ≜
62      ∧ SSend(c, [cl ∈ Client ↦ msg])
63  ⌊
```

* Modification History
* *Last* modified *Tue Dec* 04 20:49:02 *CST* 2018 by *hengxin*
* Created Sun *Jun* 24 10:25:34 *CST* 2018 by *hengxin*