```
1  ---------- MODULE JupiterInterface ----------
   This module declares the parameters and defines the operators that describe the interface of a
   family of Jupiter specs.
6  EXTENDS Integers, SequenceUtils, OT
7  ----------
8  CONSTANTS
9      Char,        the set of characters
10     Client,      the set of client replicas
11     Server,      the (unique) server replica
12     InitState    the initial state of each replica

14 ASSUME  We assume that all inserted elements are unique.
15     ∧   Range(InitState) ∩ Char = {}   due to the uniqueness requirement
16 ----------
17 VARIABLES
18     state,     state[r]: state (the list content) of replica r ∈ Replica
19     cincoming, cincoming[c]: incoming channel at the client c ∈ Client
20     sincoming, incoming channel at the Server
21     chins      a set of chars allowed to insert; this is for model checking

23 intVars ≜ ⟨state, cincoming, sincoming, chins⟩
24 ----------
25 Comm(Msg) ≜ INSTANCE CSComm

27 Replica ≜ Client ∪ {Server}

29 List ≜ Seq(Char ∪ Range(InitState))          all possible lists
30 MaxLen ≜ Cardinality(Char) + Len(InitState)  the max length of lists in any state

32 ClientNum ≜ Cardinality(Client)
33 Priority ≜ CHOOSE f ∈ [Client → 1 .. ClientNum] : Injective(f)
34 ----------
   The set of all operations. Note: The positions are indexed from 1.
38 Rd ≜ [type : {"Rd"}]
39 Del ≜ [type : {"Del"}, pos : 1 .. MaxLen]
40 Ins ≜ [type : {"Ins"}, pos : 1 .. (MaxLen + 1), ch : Char, pr : 1 .. ClientNum]  pr: priority

42 Op ≜ Ins ∪ Del   Now we don't consider Rd operations
43 ----------
44 TypeOKInt ≜
45     ∧ state ∈ [Replica → List]
46     ∧ chins ⊆ Char

48 InitInt ≜
49     ∧ state = [r ∈ Replica ↦ InitState]
50     ∧ chins = Char
```

```
52  DoIns(DoOp(_, _), c) ≜   Client c ∈ Client generates an "Ins" operation.
53      ∃ ins ∈ {op ∈ Ins :
54                    ∧ op.pos ∈ 1 .. (Len(state[c]) + 1)
55                    ∧ op.ch ∈ chins ∧ op.pr = Priority[c]} :
56          ∧ DoOp(c, ins)
57          ∧ chins′ = chins \ {ins.ch}   We assume that all inserted elements are unique.

59  DoDel(DoOp(_, _), c) ≜   Client c ∈ Client generates a "Del" operation.
60      ∃ del ∈ {op ∈ Del : op.pos ∈ 1 .. Len(state[c])} :
61          ∧ DoOp(c, del)
62          ∧ UNCHANGED chins

64  DoInt(DoOp(_, _), c) ≜   Client c ∈ Client issues an operation.
65      ∨ DoIns(DoOp, c)
66      ∨ DoDel(DoOp, c)

68  RevInt(c) ≜   Client c ∈ Client receives a message from the Server.
69      ∧ UNCHANGED chins

71  SRevInt ≜   The Server receives a message.
72      ∧ UNCHANGED chins
73  ====================================================================

    \ * Modification History
    \ * Last modified Mon Dec 31 20:27:25 CST 2018 by hengxin
    \ * Created Tue Dec 04 19:01:01 CST 2018 by hengxin
```