

```

1  |----- MODULE CJupiter -----|
   | Specification of our own CJupiter protocol; see Wei@OPODIS'2018. |
5  | EXTENDS StateSpace, JupiterSerial |
6  |-----|
7  VARIABLES
8      css      css[r]: the n-ary ordered state space at replica  $r \in Replica$ 
10 vars  $\triangleq \langle intVars, ctxVars, serialVars, css \rangle$ 
11 |-----|
12 TypeOK  $\triangleq$ 
13      $\wedge$  TypeOKInt
14      $\wedge$  TypeOKCtx
15      $\wedge$  TypeOKSerial
16      $\wedge \forall r \in Replica : IsSS(css[r])$ 
17 |-----|
18 Init  $\triangleq$ 
19      $\wedge$  InitInt
20      $\wedge$  InitCtx
21      $\wedge$  InitSerial
22      $\wedge css = [r \in Replica \mapsto EmptySS]$ 
23 |-----|
   | Iteratively transform cop with a path in the css at replica  $r \in Replica$ , following the first edges. |
28 xForm( $r, cop$ )  $\triangleq$ 
29     LET rcss  $\triangleq$  css[r]
30      $u \triangleq Locate(cop, rcss)$ 
31      $v \triangleq u \cup \{cop.oid\}$ 
32     RECURSIVE xFormHelper( $-, -, -, -$ )
33     xFormHelper( $uh, vh, coph, xcss$ )  $\triangleq$   $xcss$ : eXtra css created during transformation
34     IF  $uh = ds[r]$  THEN [ $xcss \mapsto xcss, xcop \mapsto coph$ ]
35     ELSE LET fedge  $\triangleq$  the first edge
36         CHOOSE  $e \in rcss.edge$  :
37              $\wedge e.from = uh$ 
38              $\wedge \forall uhe \in rcss.edge \setminus \{e\} :$ 
39                  $(uhe.from = uh) \Rightarrow tb(e.cop.oid, uhe.cop.oid, serial[r])$ 
40              $uprime \triangleq fedge.to$ 
41              $fcop \triangleq fedge.cop$ 
42              $coph2fcop \triangleq COT(coph, fcop)$ 
43              $fcop2coph \triangleq COT(fcop, coph)$ 
44              $vprime \triangleq vh \cup \{fcop.oid\}$ 
45         IN xFormHelper( $uprime, vprime, coph2fcop,$ 
46              $xcss \oplus [node \mapsto \{vprime\},$ 
47                  $edge \mapsto \{[from \mapsto vh, to \mapsto vprime, cop \mapsto fcop2coph],$ 
48                      $[from \mapsto uprime, to \mapsto vprime, cop \mapsto coph2fcop]\}$ )
49         IN xFormHelper( $u, v, cop, [node \mapsto \{v\}, edge \mapsto \{[from \mapsto u, to \mapsto v, cop \mapsto cop]\}$ )

```

```

51  $Perform(r, cop) \triangleq$ 
52   LET  $xform \triangleq xForm(r, cop)$   $xform: [xcss, xcop]$ 
53   IN    $\wedge css' = [css \text{ EXCEPT } ![r] = @ \oplus xform.xcss]$ 
54        $\wedge SetNewAop(r, xform.xcop.op)$ 

56  $ServerPerform(cop) \triangleq$ 
57    $\wedge Perform(Server, cop)$ 
58    $\wedge Comm!SSendSame(ClientOf(cop), cop)$   $\text{broadcast the original operation}$ 
59 |-----|

60  $DoOp(c, op) \triangleq$ 
61   LET  $cop \triangleq [op \mapsto op, oid \mapsto [c \mapsto c, seq \mapsto cseq[c]], ctx \mapsto ds[c]]$ 
62   IN    $\wedge Perform(c, cop)$ 
63        $\wedge Comm!CSend(cop)$ 

65  $Do(c) \triangleq$ 
66    $\wedge DoInt(DoOp, c)$ 
67    $\wedge DoCtx(c)$ 
68    $\wedge DoSerial(c)$ 

70  $Rev(c) \triangleq$ 
71    $\wedge RevInt(Perform, c)$ 
72    $\wedge RevCtx(c)$ 
73    $\wedge RevSerial(c)$ 

75  $SRev \triangleq$ 
76    $\wedge SRevInt(ServerPerform)$ 
77    $\wedge SRevCtx$ 
78    $\wedge SRevSerial$ 
79 |-----|

80  $Next \triangleq$ 
81    $\vee \exists c \in Client : Do(c) \vee Rev(c)$ 
82    $\vee SRev$ 

84  $Fairness \triangleq$   $\text{There is no requirement that the clients ever generate operations.}$ 
85    $WF_{vars}(SRev \vee \exists c \in Client : Rev(c))$ 

87  $Spec \triangleq Init \wedge \Box[Next]_{vars} \wedge Fairness$ 
88 |-----|

89  $Compactness \triangleq$   $\text{Compactness of CJupiter: the CSSes at all replicas are the same.}$ 
90    $Comm!EmptyChannel \Rightarrow Cardinality(Range(css)) = 1$ 

92 THEOREM  $Spec \Rightarrow Compactness$ 
93 |-----|

\ * Modification History
\ * Last modified Wed Jan 02 21:01:46 CST 2019 by hengxin
\ * Created Sat Sep 01 11:08:00 CST 2018 by hengxin

```