

```

1  |----- MODULE CAbsJupiter -----|
   | Centralized version of AbsJupiter. |
5  | EXTENDS SetStateSpace |
6  |-----|
7  | VARIABLES |
8  |   copss,   copss[r]: the state space (i.e., a set) of Cop maintained at replica r ∈ Replica |
9  |   hb,      happens-before relation among operations |
10 |   sv       global serial view (established at the server) |
12 | vars ≜ ⟨intVars, ctxVars, copss⟩ |
13 |-----|
14 | TypeOK ≜ |
15 |   ∧ TypeOKInt |
16 |   ∧ TypeOKCtx |
17 |   ∧ copss ∈ [Replica → SUBSET Cop] |
18 |   ∧ hb ⊆ Oid × Oid |
19 |   ∧ sv ∈ Seq(Oid) |
20 |-----|
21 | Init ≜ |
22 |   ∧ InitInt |
23 |   ∧ InitCtx |
24 |   ∧ copss = [r ∈ Replica ↦ {}] |
25 |   ∧ hb = {} |
26 |   ∧ sv = ⟨⟩ |
27 |-----|
28 | NextCop(r, cop, ss, ctx) ≜ Return the next fcop ∈ Cop against which cop is to be transformed. |
29 |   LET foid ≜ CHOOSE oid ∈ ctx : the first oid in ctx according to serial[r] |
30 |           ∀ id ∈ ctx \ {oid} : TRUE tb(oid, id, serial[r]) |
31 |   IN   CHOOSE fcop ∈ ss : THEOREM : Existence of fcop |
32 |           fcop.oid = foid ∧ fcop.ctx = cop.ctx |
34 | Perform(r, cop) ≜ |
35 |   LET xform ≜ xForm(NextCop, r, cop, copss[r]) | [xcop, xss] |
36 |   IN   ∧ copss' = [copss EXCEPT ![r] = xform.xss] |
37 |           ∧ SetNewAop(r, xform.xcop.op) |
39 | ClientPerform(c, cop) ≜ Perform(c, cop) |
41 | ServerPerform(cop) ≜ |
42 |   ∧ Perform(Server, cop) |
43 |   ∧ Comm!SSendSame(ClientOf(cop), cop) |
44 |-----|
45 | DoOp(c, op) ≜ |
46 |   LET cop ≜ [op ↦ op, oid ↦ [c ↦ c, seq ↦ cseq[c]] |
47 |   IN   ∧ ClientPerform(c, cop) |
48 |           ∧ Comm!CSend(cop) |

```

```

50  $Do(c) \triangleq$ 
51    $\wedge DoInt(DoOp, c)$ 
52    $\wedge DoCtx(c)$ 

54  $Rev(c) \triangleq$ 
55    $\wedge RevInt(ClientPerform, c)$ 
56    $\wedge RevCtx(c)$ 

58  $SRev \triangleq$ 
59    $\wedge SRevInt(ServerPerform)$ 
60    $\wedge SRevCtx$ 

61 |-----|
62  $Next \triangleq$ 
63    $\vee \exists c \in Client : Do(c) \vee Rev(c)$ 
64    $\vee SRev$ 

66  $Fairness \triangleq$ 
67    $WF_{vars}(SRev \vee \exists c \in Client : Rev(c))$ 

69  $Spec \triangleq Init \wedge \Box [Next]_{vars} \wedge Fairness$ 
70 |-----|
71  $QC \triangleq$  Quiescent Consistency
72    $Comm!EmptyChannel \Rightarrow Cardinality(Range(state)) = 1$ 
73 THEOREM  $Spec \Rightarrow \Box QC$ 

75  $SEC \triangleq$  Strong Eventual Consistency
76    $\forall r1, r2 \in Replica :$ 
77      $ds[r1] = ds[r2] \Rightarrow state[r1] = state[r2]$ 
78 THEOREM  $Spec \Rightarrow \Box SEC$ 

80  $Compactness \triangleq$  Compactness of state space
81    $Comm!EmptyChannel \Rightarrow Cardinality(Range(copss)) = 1$ 
82 THEOREM  $Spec \Rightarrow \Box Compactness$ 
83 |-----|

\ * Modification History
\ * Last modified Mon Feb 25 21:29:21 CST 2019 by hengxin
\ * Created Sun Feb 24 20:32:34 CST 2019 by hengxin

```