

```

1  |----- MODULE Peer -----|
2  EXTENDS Sequences

4  CONSTANTS Id, Id of the peer, given by the Fabric.tla
5             Key, Value, Version, Set of keys, values, and its version
6             Arg, arguments, implicitly includes a function name as the first argument
7             Transaction, Transactions, given by the Fabric.tla
8             Commit(-, -, -) commit action

10 VARIABLES msgs, Message buffer, given by the Fabric.tla
11            ledger Ledger of the peer

13 LOCAL vars  $\triangleq \langle msgs, ledger \rangle$ 

15 typedefs

17 LOCAL Send(m)  $\triangleq msgs' = msgs \cup \{m\}$ 

19 State  $\triangleq [Key \rightarrow (Value \times Version)]$  State DB
20 Chain  $\triangleq Seq(Transaction)$  Blockchain
21 RWSet  $\triangleq [read : SUBSET (Key \times Version), write : SUBSET (Key \times Value)]$ 
22 SC  $\triangleq [(State \times Arg) \rightarrow RWSet]$  At this point, we assume it's deterministic
23 Ledger  $\triangleq [chain : Chain, state : State, sc : SC]$ 

25 ASSUME  $\forall stateBefore, stateAfter, rwset :$ 
26       Commit(stateBefore, rwset, stateAfter)  $\in$  BOOLEAN

28 actions

30 Simulate(tx)  $\triangleq$ 
31   LET
32     arg  $\triangleq tx.arg$ 
33     rwset  $\triangleq ledger.sc[ledger.state, arg]$ 
34   IN Send([type  $\mapsto$  , rwset  $\mapsto$  rwset])

Spec template

40 Init  $\triangleq msgs = \{\} \wedge ledger = [chain : \langle \rangle, state : \{\}, sc \in SC]$ 
41 Next  $\triangleq \exists m \in msgs, tx \in Transaction :$ 
42     m.type = "invoke"  $\wedge m.tx = tx \wedge Simulate(tx)$ 

44 Spec  $\triangleq Init \wedge \Box [Next]_{vars}$ 

46 |-----|
    * Modification History
    * Last modified Fri Apr 05 02:21:56 JST 2019 by shinsa
    * Created Thu Apr 04 15:06:48 JST 2019 by shinsa

```