MODULE *MCPrimSecRep*

EXTENDS *PrimSecRep*, *TLC*

CONSTANTS *MaxReq*,
MaxVer

$mcvars \triangleq vars$

$MCInit \triangleq Init$

Constraints: channel sizes should be limited and version number should not grow infinitely.

$MaxChannelConstr \triangleq$ imposes a limit on the size of the channels at any time

$\forall\, r \in Rep : \land Len(channel[r]) < MaxReq$

$MaxVersionNumber \triangleq$ impose a limit on the version numbers

$\forall\, o \in Object : master.objects[o].version < MaxVer$

Reduce the state space by taking advantage of symmetry

$Perms \triangleq Permutations(Val) \cup Permutations(Rep) \cup Permutations(Object)$

Reset the version number of object o, to create the impression of infinite evolvement. The action is enabled only when all replicas know the same version number.

$\_ResetVersion(o) \triangleq$

$\land\ master.objects[o].version \notin \{0, 1\}$ object exists and it's not at the first version already

$\land \forall\, r \in Rep \cap (\{master.objects[o].prim\} \cup master.objects[o].sec) :$

For all replicas that store the object

$\land\ cache[r][o].version = master.chunks[o].version$

$\land\ master' = [master \text{ EXCEPT } !.objects[o].version = 1]$

$\land\ cache' = [r \in Rep \mapsto$
IF $r \in Rep \cap (\{master.objects[o].prim\} \cup master.objects[o].sec)$
THEN $[cache[r] \text{ EXCEPT } !.objects[o].version = 1]$
ELSE $cache[r]]$

$\land$ UNCHANGED $\langle data, channel, stat \rangle$

$ResetVersion \triangleq$

$\exists\, o \in Object : \_ResetVersion(o)$

$MCNext \triangleq Next \lor ResetVersion$

$MCSpec \triangleq \land \Box[MCNext]_{mcvars}$