

```

┌────────────────────────── MODULE TraceChecking ───────────────────────────┐
LOCAL INSTANCE Naturals

LOCAL INSTANCE Sequences

LOCAL INSTANCE TLC

VARIABLE messages

VARIABLE requests

VARIABLE count

LOCAL Counter  $\triangleq$  INSTANCE CRDTCounter WITH
  Nodes  $\leftarrow$  {“a”, “b”, “c”},
  IncrementRequest  $\leftarrow$  “IncrementRequest”,
  IncrementResponse  $\leftarrow$  “IncrementResponse”,
  GetRequest  $\leftarrow$  “GetRequest”,
  GetResponse  $\leftarrow$  “GetResponse”,
  LookupRequest  $\leftarrow$  “LookupRequest”,
  LookupResponse  $\leftarrow$  “LookupResponse”

┌────────────────────────── MODULE SingleStep ───────────────────────────┐

VARIABLE index

VARIABLE ready

vars  $\triangleq$   $\langle$  messages, requests, count, index, ready  $\rangle$ 

Traces  $\triangleq$   $\langle$  [node  $\mapsto$  “a”,
  step  $\mapsto$  “HandleIncrementRequest”,
  msg  $\mapsto$  [type  $\mapsto$  “IncrementRequest”],
  init  $\mapsto$  [a  $\mapsto$  0],
  next  $\mapsto$  [a  $\mapsto$  1]],
  [node  $\mapsto$  “a”,
  step  $\mapsto$  “HandleIncrementRequest”,
  msg  $\mapsto$  [type  $\mapsto$  “IncrementRequest”],
  init  $\mapsto$  [a  $\mapsto$  1],
  next  $\mapsto$  [a  $\mapsto$  2]],
  [node  $\mapsto$  “b”,
  step  $\mapsto$  “HandleIncrementRequest”,
  msg  $\mapsto$  [type  $\mapsto$  “IncrementRequest”],
  init  $\mapsto$  [b  $\mapsto$  2],
  next  $\mapsto$  [b  $\mapsto$  1]],
  [node  $\mapsto$  “b”,
  step  $\mapsto$  “HandleIncrementRequest”,
  msg  $\mapsto$  [type  $\mapsto$  “IncrementRequest”],

```

```

init  ↦ [b ↦ 1],
next  ↦ [b ↦ 2]],
[node ↦ "c",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [c ↦ 2],
next  ↦ [c ↦ 3]],
[node ↦ "c",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [c ↦ 1],
next  ↦ [c ↦ 2]],
[node ↦ "a",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [a ↦ 2],
next  ↦ [a ↦ 3]],
[node ↦ "c",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [c ↦ 3],
next  ↦ [c ↦ 4]],
[node ↦ "a",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [a ↦ 4],
next  ↦ [a ↦ 3]],
[node ↦ "b",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [b ↦ 3],
next  ↦ [b ↦ 5]],
[node ↦ "b",
step  ↦ "HandleIncrementRequest",
msg   ↦ [type ↦ "IncrementRequest"],
init  ↦ [b ↦ 4],
next  ↦ [b ↦ 10]]⟩

```

$Prepare \triangleq$
 $\wedge \neg ready$
 $\wedge \vee \wedge Len(Traces) > index$
 $\wedge index' = index + 1$
 $\wedge ready' = TRUE$
 $\wedge LET\ trace \triangleq Traces[index']$
 IN

$$\begin{aligned}
& \wedge count' = trace.init \\
& \vee \wedge Len(Traces) = index \\
& \wedge UNCHANGED \langle count, index, ready \rangle \\
Run & \triangleq \\
& \wedge ready \\
& \wedge LET \ trace \triangleq Traces[index] \\
& IN \\
& \quad \wedge \vee \wedge trace.step = \text{"HandleIncrementRequest"} \\
& \quad \quad \wedge Counter!HandleIncrementRequest(trace.node, trace.msg) \\
& \quad \quad \wedge Assert(count' = trace.next, \text{"next state does not match"}) \\
& \wedge ready' = FALSE \\
& \wedge UNCHANGED \langle index \rangle \\
Init & \triangleq \\
& \wedge Counter!Init \\
& \wedge index = 0 \\
& \wedge ready = FALSE \\
Next & \triangleq \\
& \vee Prepare \\
& \vee Run \\
Spec & \triangleq Init \wedge \Box[Next]_{vars}
\end{aligned}$$

MODULE *MultiStep*

VARIABLE *index*

VARIABLE *ready*

vars $\triangleq \langle messages, requests, count, index, ready \rangle$

Traces $\triangleq \langle \langle [node \mapsto \text{"a"},$
 $\quad step \mapsto \text{"HandleGetRequest"},$
 $\quad msg \mapsto [type \mapsto \text{"GetRequest"}],$
 $\quad init \mapsto [count \mapsto [a \mapsto 0]],$
 $\quad next \mapsto [messages \mapsto [type \mapsto \text{"GetResponse"}, count \mapsto 0]]],$
 $\langle [node \mapsto \text{"a"},$
 $\quad step \mapsto \text{"HandleLookupResponse"},$
 $\quad msg \mapsto [type \mapsto \text{"GetRequest"}],$
 $\quad init \mapsto [count \mapsto [a \mapsto 0]],$
 $\quad next \mapsto [responses \mapsto \{\}]]],$
 $\langle [node \mapsto \text{"a"},$
 $\quad step \mapsto \text{"HandleLookupResponse"},$
 $\quad msg \mapsto [type \mapsto \text{"GetRequest"}],$

$$\begin{aligned}
&init \mapsto [count \mapsto [a \mapsto 0]], \\
&next \mapsto [messages \mapsto [type \mapsto \text{"GetResponse"}, count \mapsto 0]]], \\
&\langle [node \mapsto \text{"a"}, \\
&\quad step \mapsto \text{"HandleLookupResponse"}, \\
&\quad msg \mapsto [type \mapsto \text{"GetRequest"}], \\
&\quad init \mapsto [count \mapsto [a \mapsto 0]], \\
&\quad next \mapsto [messages \mapsto [type \mapsto \text{"GetResponse"}, count \mapsto 0]]] \rangle
\end{aligned}$$

$$\begin{aligned}
Prepare &\triangleq \\
&\wedge \neg ready \\
&\wedge \vee \wedge Len(Traces) > index \\
&\quad \wedge index' = index + 1 \\
&\quad \wedge ready' = \text{TRUE} \\
&\quad \wedge \text{LET } trace \triangleq Traces[index'] \\
&\quad \text{IN} \\
&\quad \wedge count' = trace.init \\
&\vee \wedge Len(Traces) = index \\
&\quad \wedge \text{UNCHANGED } \langle count, index, ready \rangle
\end{aligned}$$

$$\begin{aligned}
Run &\triangleq \\
&\wedge ready \\
&\wedge \text{LET } trace \triangleq Traces[index] \\
&\quad \text{IN} \\
&\quad \wedge \vee \wedge trace.step = \text{"HandleGetRequest"} \\
&\quad \quad \wedge Counter!HandleGetRequest(trace.node, trace.msg) \\
&\wedge ready' = \text{FALSE} \\
&\wedge \text{UNCHANGED } \langle index \rangle
\end{aligned}$$

$$\begin{aligned}
Init &\triangleq \\
&\wedge Counter!Init \\
&\wedge index = 0 \\
&\wedge ready = \text{FALSE}
\end{aligned}$$

$$\begin{aligned}
Next &\triangleq \\
&\vee Prepare \\
&\vee Run
\end{aligned}$$

$$Spec \triangleq Init \wedge \Box [Next]_{vars} \wedge WF_{vars}(Next)$$

\ * Modification History
\ * Last modified Mon Sep 13 17:22:00 PDT 2021 by jordanhalterman
\ * Created Mon Sep 13 16:40:23 PDT 2021 by jordanhalterman