- MODULE VoucherLifeCycle -

This specification is of a Voucher and it's life cycle. This is based on the definiton of Vouchers in RFC 3506 with the tuple part decoupled.

Note: A new state called "phantom" was introduced to indicate the state of a voucher that is yet to be issued, once a voucher is issued it becomes a "valid" voucher. This is a one way transition and it cannot reversed.

```
CONSTANT V The set of vouchers.

VARIABLE vState, vState[v] is the state of a voucher v.

The state of the voucher life cycle machine.

vvlcState[v] is the state of the life cycle machine for the voucher v.
```

$VTypeOK \triangleq$

```
The type-correctness invariant
```

$VInit \triangleq$

The initial predicate.

We now define the actions that may be performed on the Vs, and then define the complete next-state action of the specification to be the disjunction of the possible V actions.

```
Redeem(v) \triangleq \\ \land vState[v] = \text{"valid"} \\ \land vlcState[v] = \text{"working"} \\ \land vState' = [vState \text{ EXCEPT }![v] = \text{"redeemed"}] \\ \land vlcState' = [vlcState \text{ EXCEPT }![v] = \text{"done"}] \\ Cancel(v) \triangleq \\ \land vState[v] = \text{"valid"} \\ \land vlcState[v] = \text{"working"} \\ \land vState' = [vState \text{ EXCEPT }![v] = \text{"cancelled"}] \\ \land vlcState' = [vlcState \text{ EXCEPT }![v] = \text{"done"}] \\ VNext \triangleq \exists v \in V : Issue(v) \lor Redeem(v) \lor Transfer(v) \lor Cancel(v) \\ \end{cases}
```

The next-state action.

$VConsistent \triangleq$

A state predicate asserting that a V started at a valid start state and has reached a valid final state at the end of the life cycle. V can be "valid" only when the state of the machine is "working". It is an invariant of the specification.

$VSpec \stackrel{\Delta}{=} VInit \wedge \Box [VNext]_{\langle vState, \, vlcState \rangle}$

The complete specification of the protocol written as a temporal formula.

THEOREM $VSpec \Rightarrow \Box(VTypeOK \land VConsistent)$

This theorem asserts the truth of the temporal formula whose meaning is that the state predicate $VTypeOK \land VConsistent$ is an invariant of the specification VSpec. Invariance of this conjunction is equivalent to invariance of both of the formulas VTypeOK and VConsistent.

- * Last modified $Tue\ Jun\ 12\ 13:25:29\ IST\ 2018$ by Fox
- * Created Fri Mar 16 11:56:25 SGT 2018 by Fox