pno	statusc	idlec
O))	valc	n

	statusp	idlep
pc-1	valp	k
	tmp	0

	statusp	idlep	
3c-2	valp	1	
	tmp	0	

_	statusp	idlep
2C-1	valp	m
_	tmp	0

	statusc	idlec
၅	valc	n

transition: modval

k replaced by new if pc-1 is idlep new is a (arbitrary) new value

	statusp	idlep
pc-1	valp	new
	tmp	0

	statusp	idlep	
3c-2	valp	1	•
_	tmp	0	

	statusp	idlep
n-oc	valp	m
_	tmp	0

pno	statusc	busy
ပိ	valc	n

transition: gotvalue

	statusp	gotvalue
pc-1	valp	new
	tmp	n

	statusp	idlep	
3c-2	valp	1	
_	tmp	0	

	statusp	idlep
n-oc	valp	m
_	tmp	0

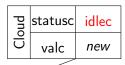
pno	statusc	busy
ပိ	valc	new

 $\widehat{\text{transition:}}$ update assuming $new \geq n$

	statusp	update
pc-1	valp	new
	tmp	new

	statusp	idlep	
oc-2	valp	1	
_	tmp	0	

	statusp	idlep
n-oc	valp	m
	tmp	0



transition: gotoidle

	statusp	idlep
pc-1	valp	new
_	tmp	0

	statusp	idlep
oc-2	valp	1
_	tmp	0

	statusp	idlep
n-oc	valp	m
	tmp	0

Transitions

- GetValue: if PC and Cloud is idle, fetch Cloud value
- ▶ Update: update Cloud/PC according to larger value
- ► Gotoldle: both PC and Cloud go back to idle

We need to define the value of each observer after applying each of the transitions above.

GOAL

If PC is in updated state, then the values of Cloud and PC agree