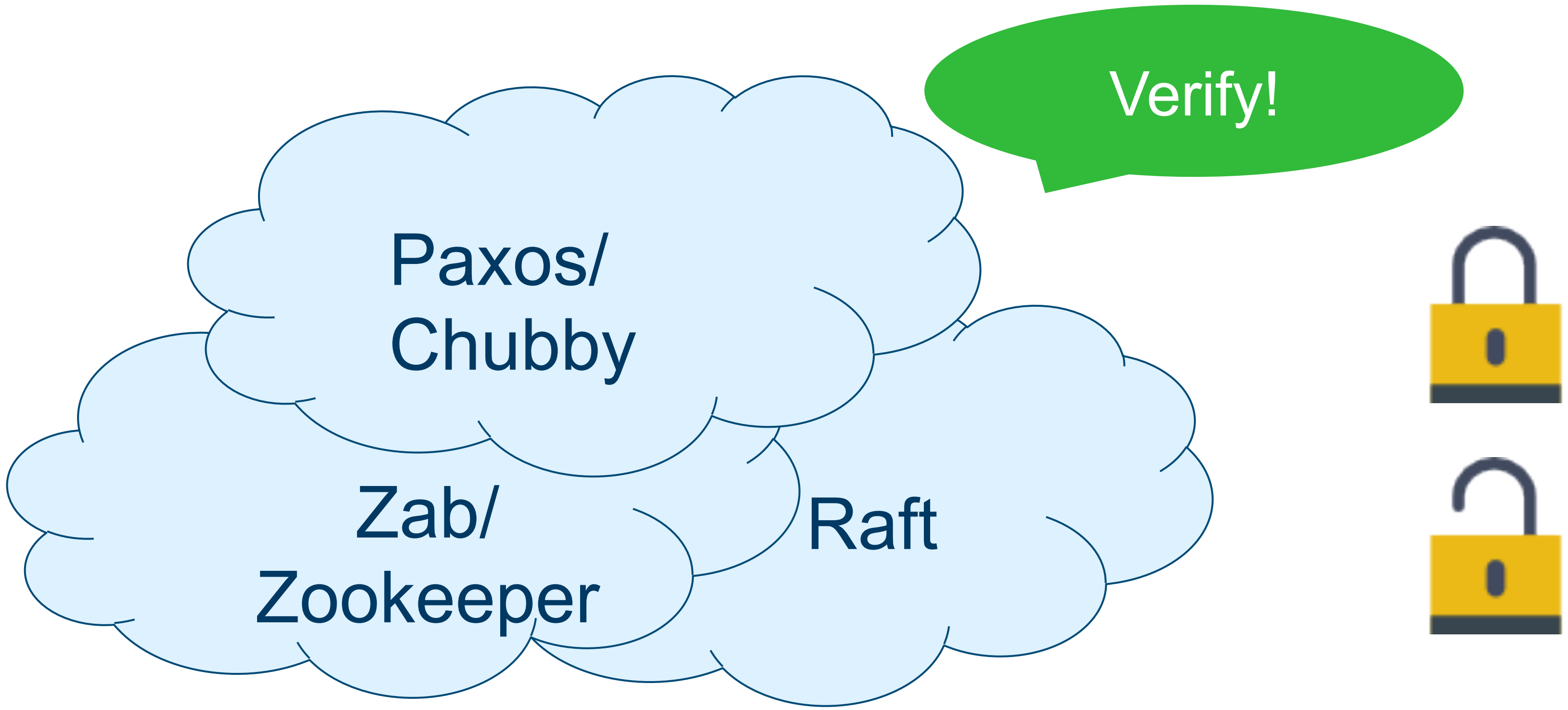
Cardinalities and Universal Quantifiers for Verifying Parameterized Systems

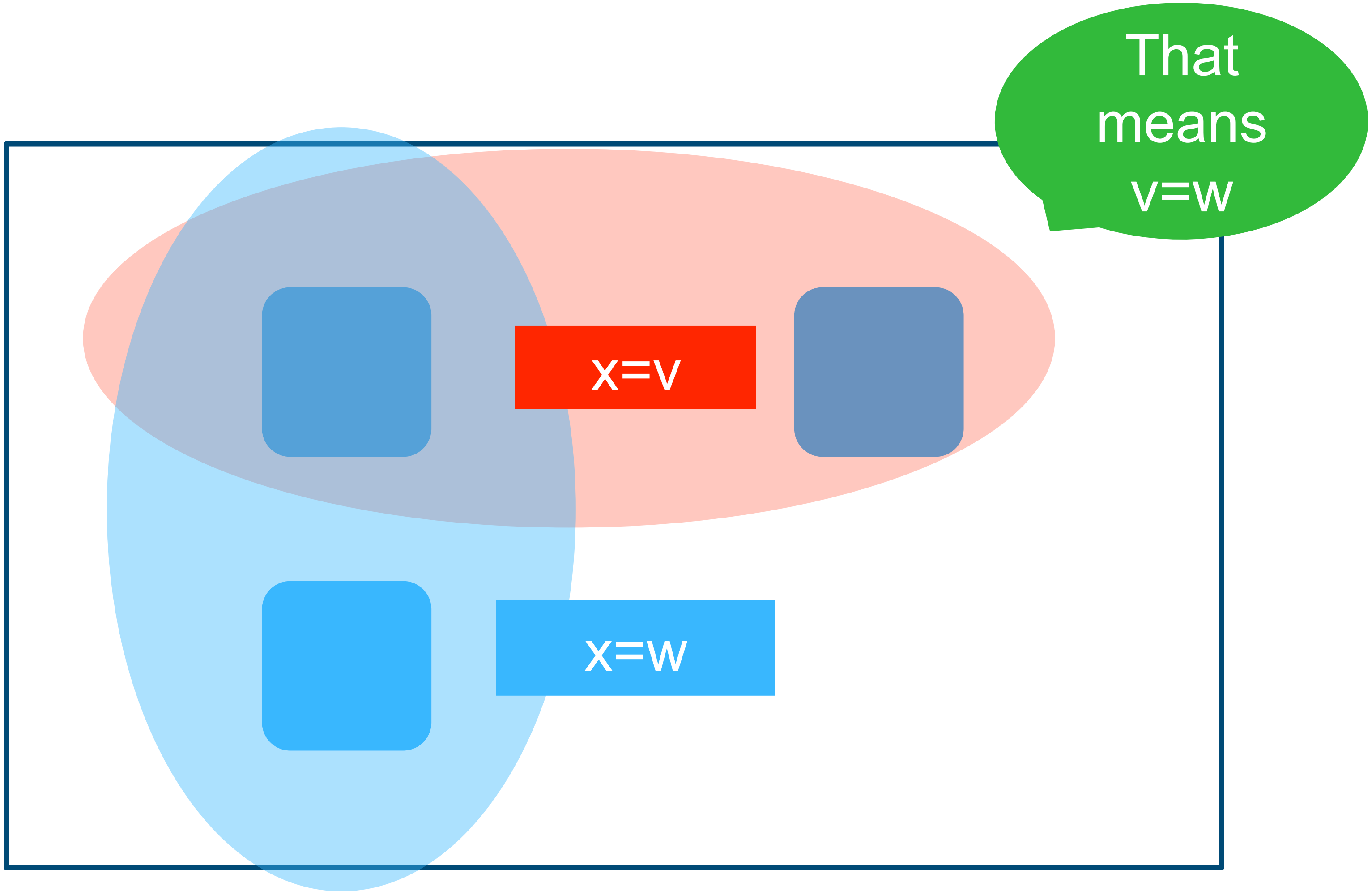
Klaus v. Gleissenthall, UC San Diego and TU Munich

Nikolaj Bjørner and Andrey Rybalchenko,
Microsoft Research

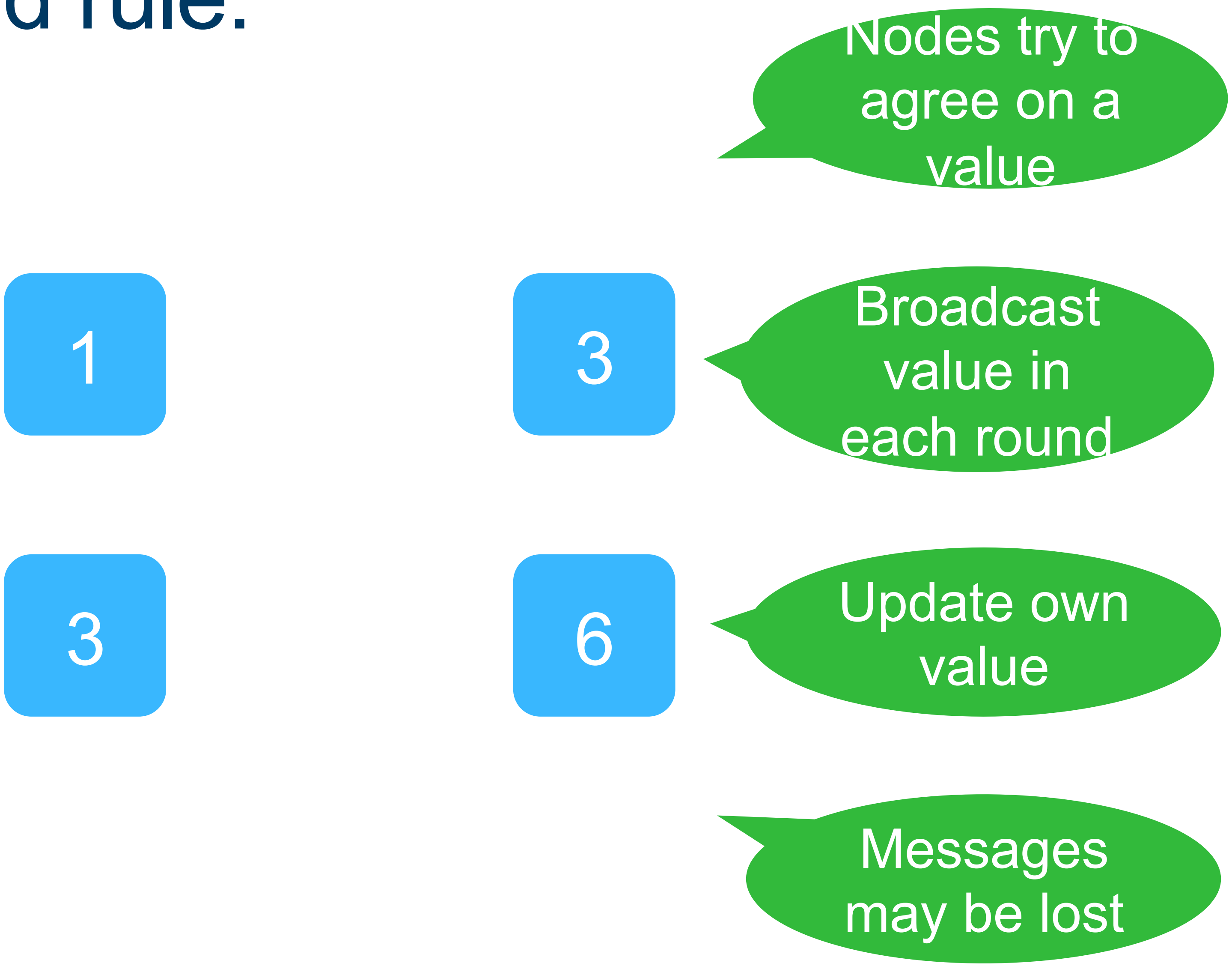
Parallel / Distributed systems



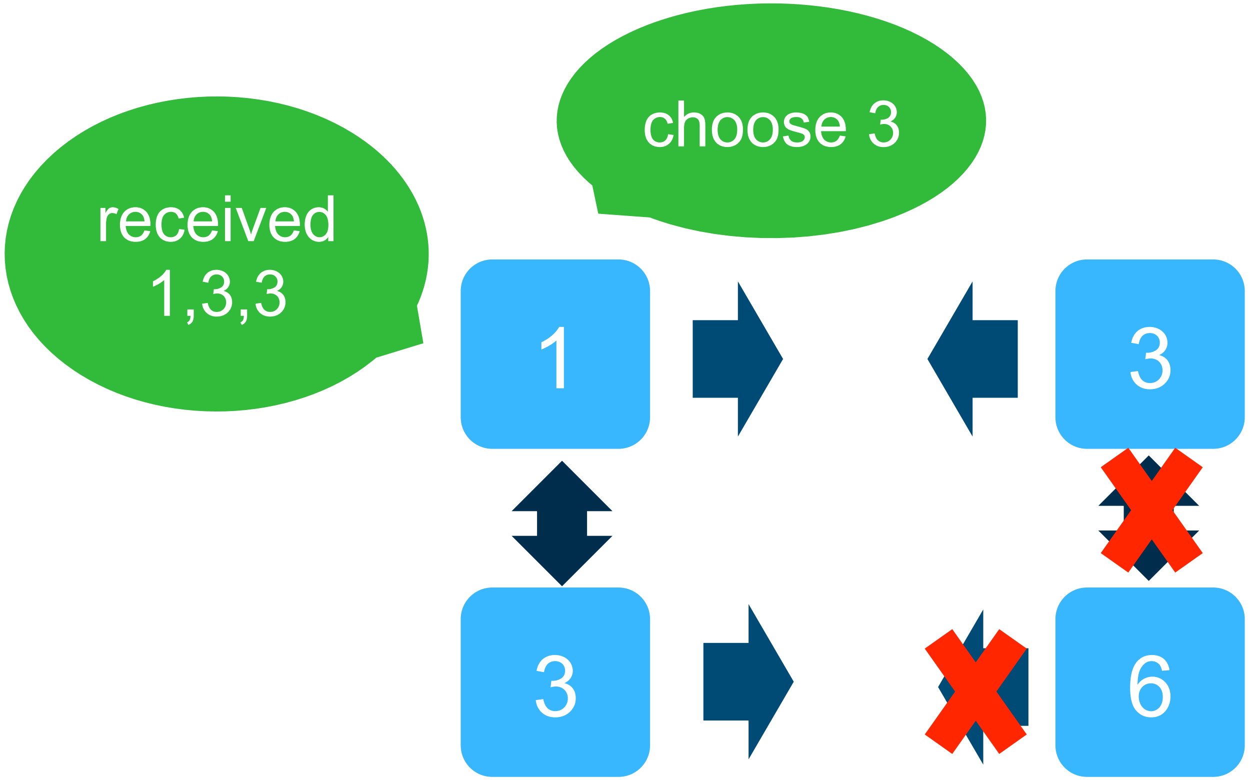
Consensus: majorities



One third rule:



One third rule:

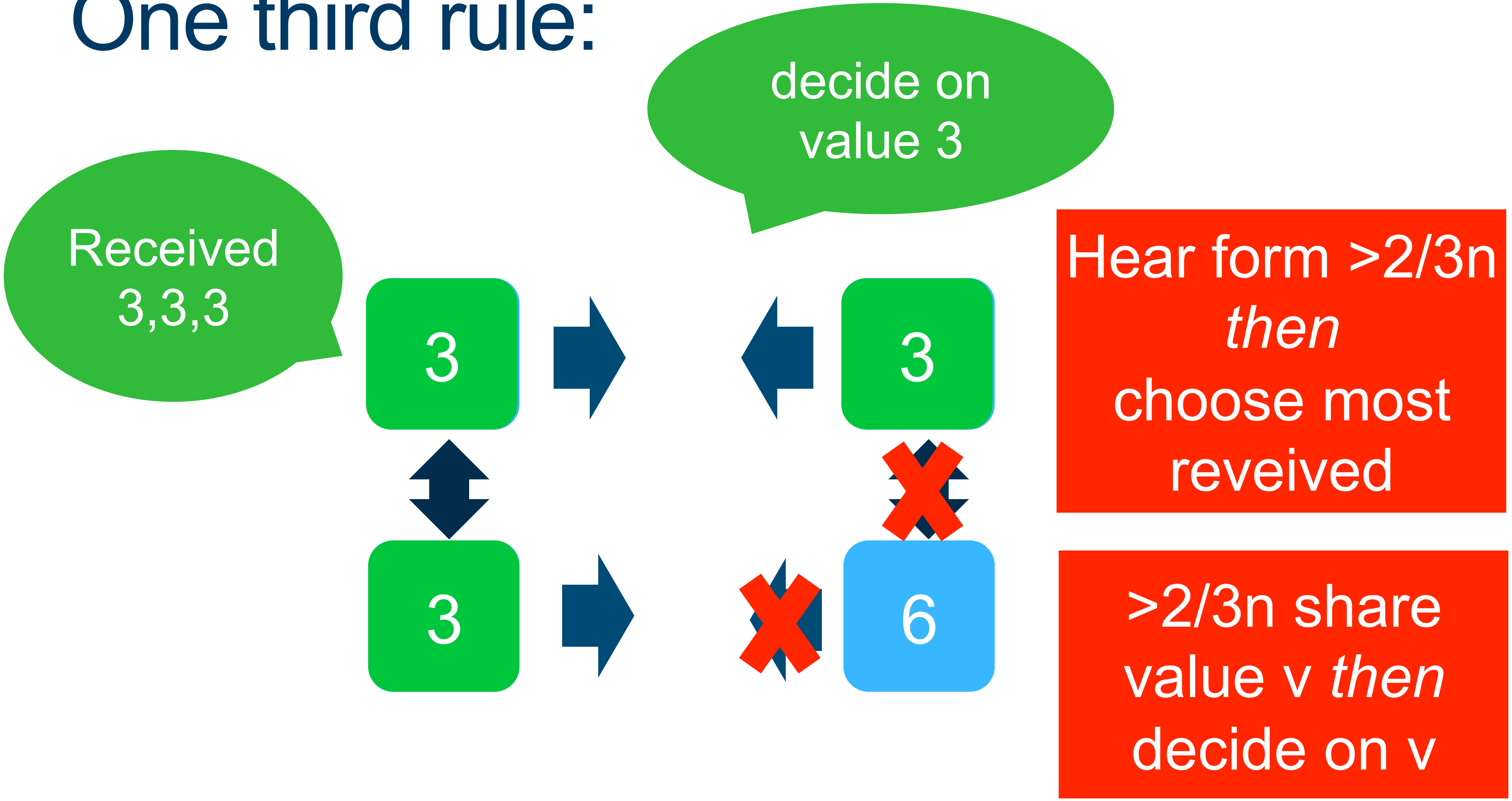


n= # of nodes

Hear form >2/3n *then* choose *most reveived*

>2/3n share value v *then decide on v*

One third rule:



One third rule: property

We want to verify

forall p,p': decide(p,v) *and* decide(p',w) *then* v=w

Agreement

One third rule: invariant

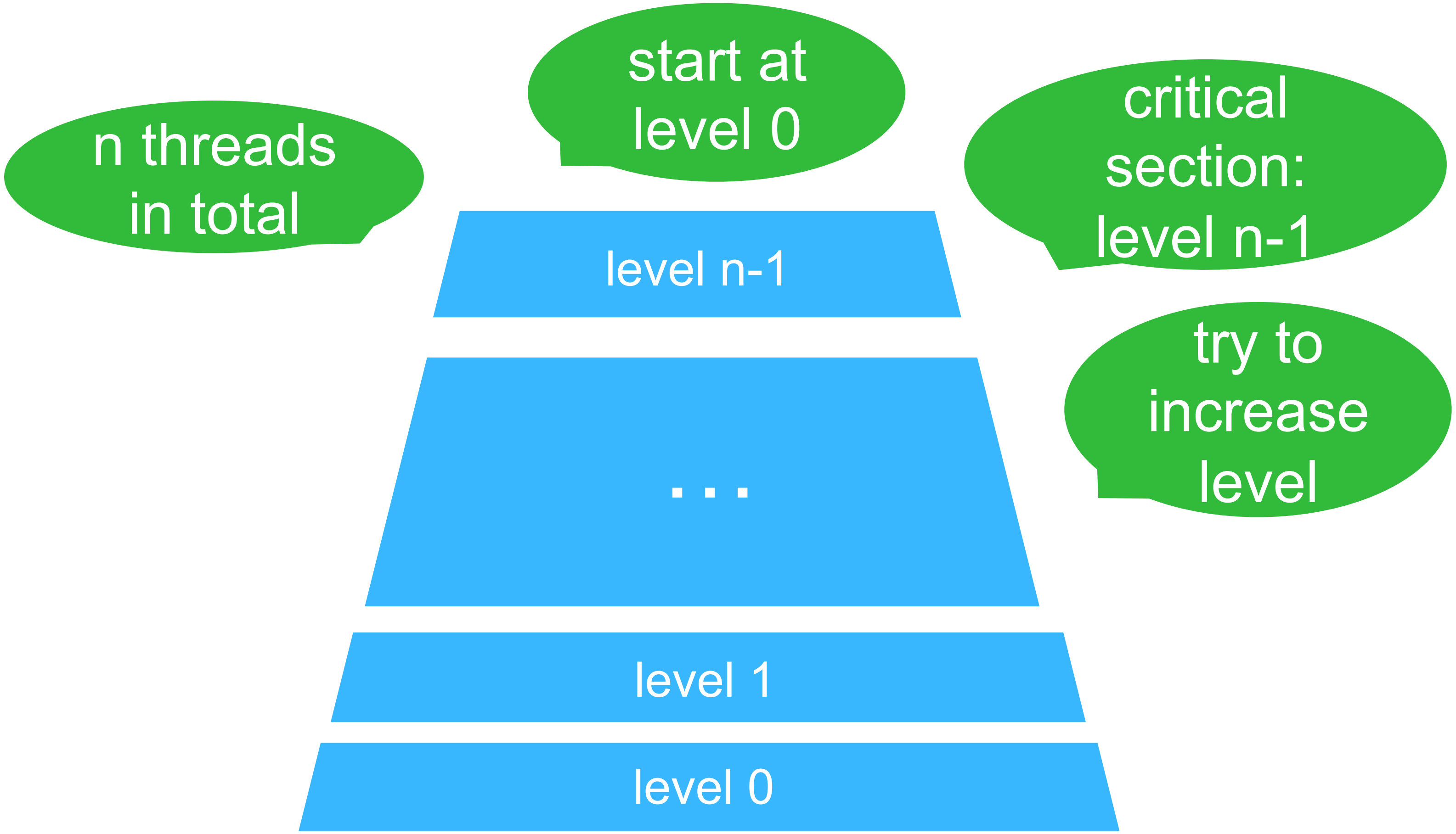
Quantification: number of processes not known statically

Agreement by: no two majorities

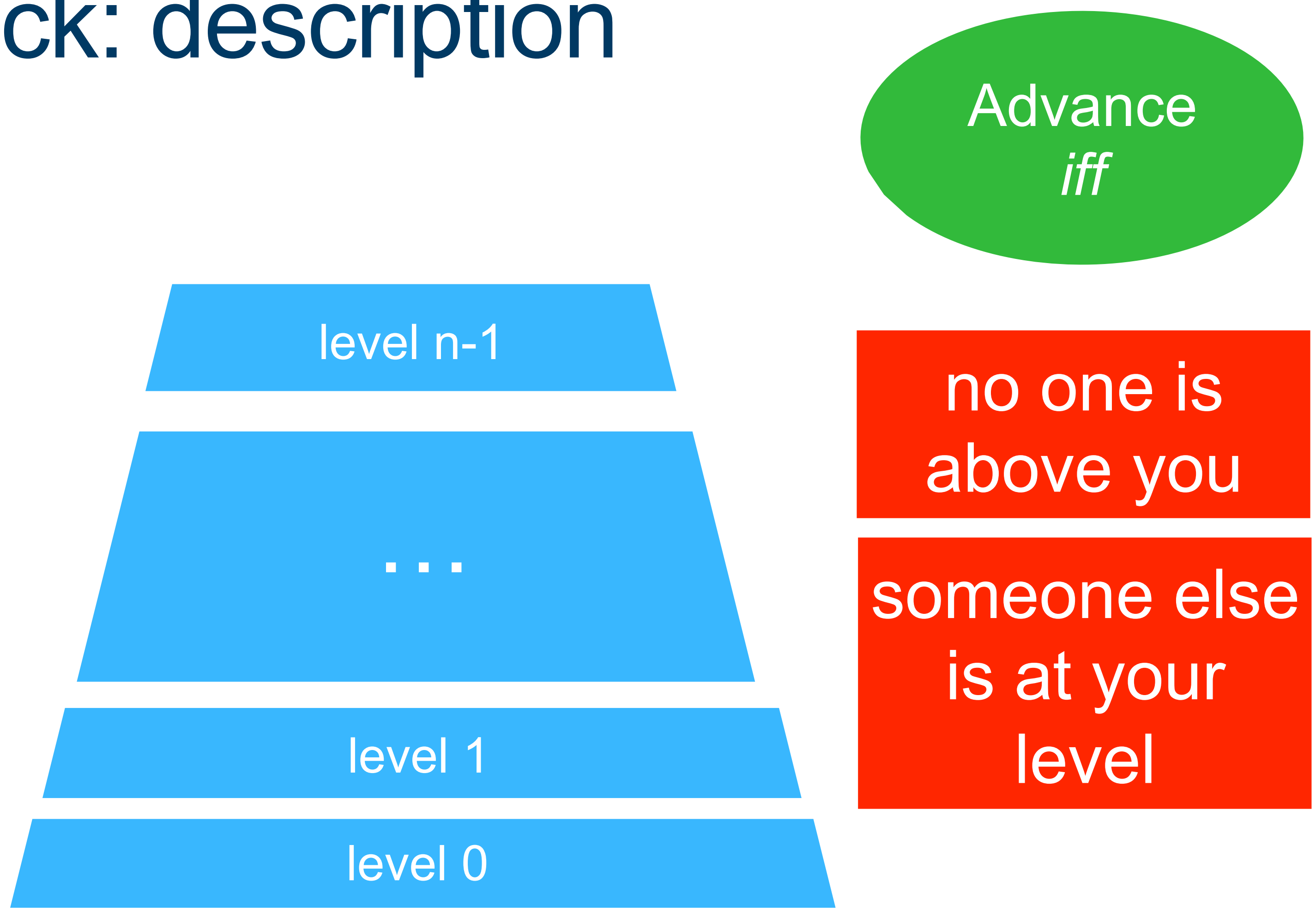
forall p: decide(p,v) *then*
#{t | candidate(t)=v } >2/3n

Count # of nodes with same candidate

Filter lock: description



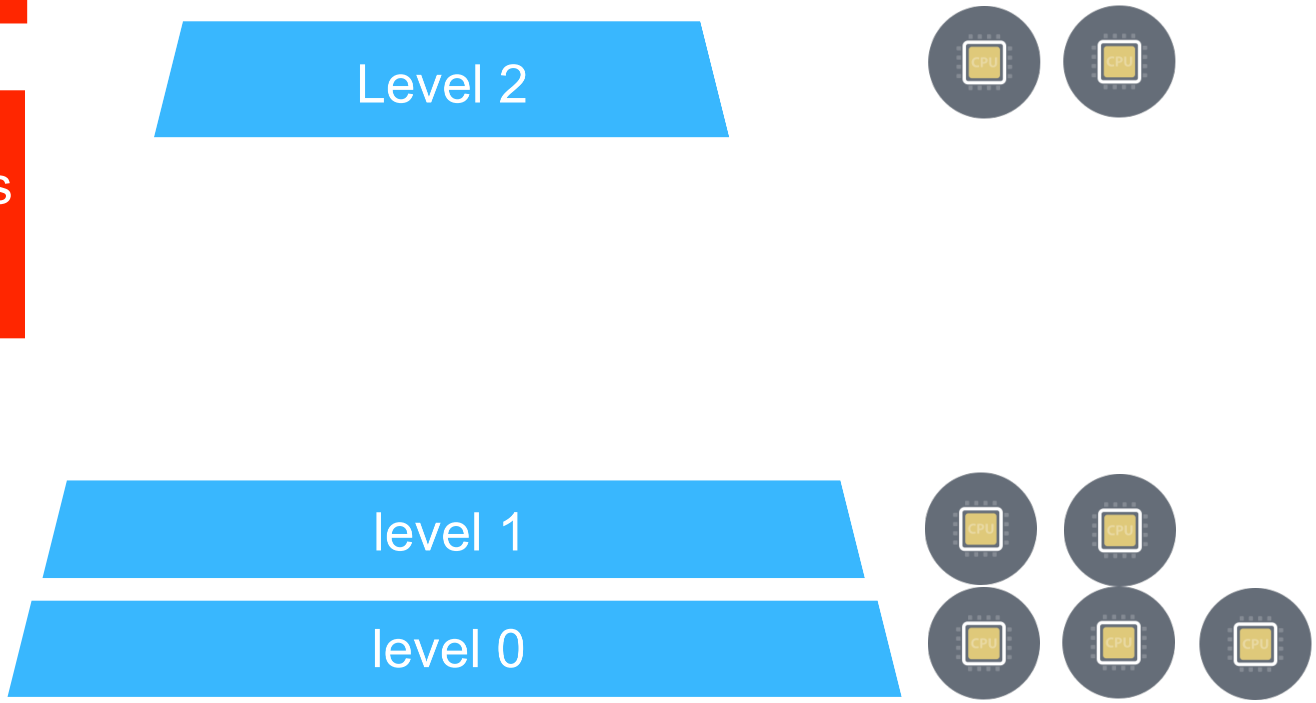
Filter lock: description



Filter lock: description

no one is above you

someone else is at your level



Filter Lock: property

Mutual exclusion

$\#\{ t \mid lv(t) = n\text{-}1 \} \leq 1$

Filter Lock: invariant

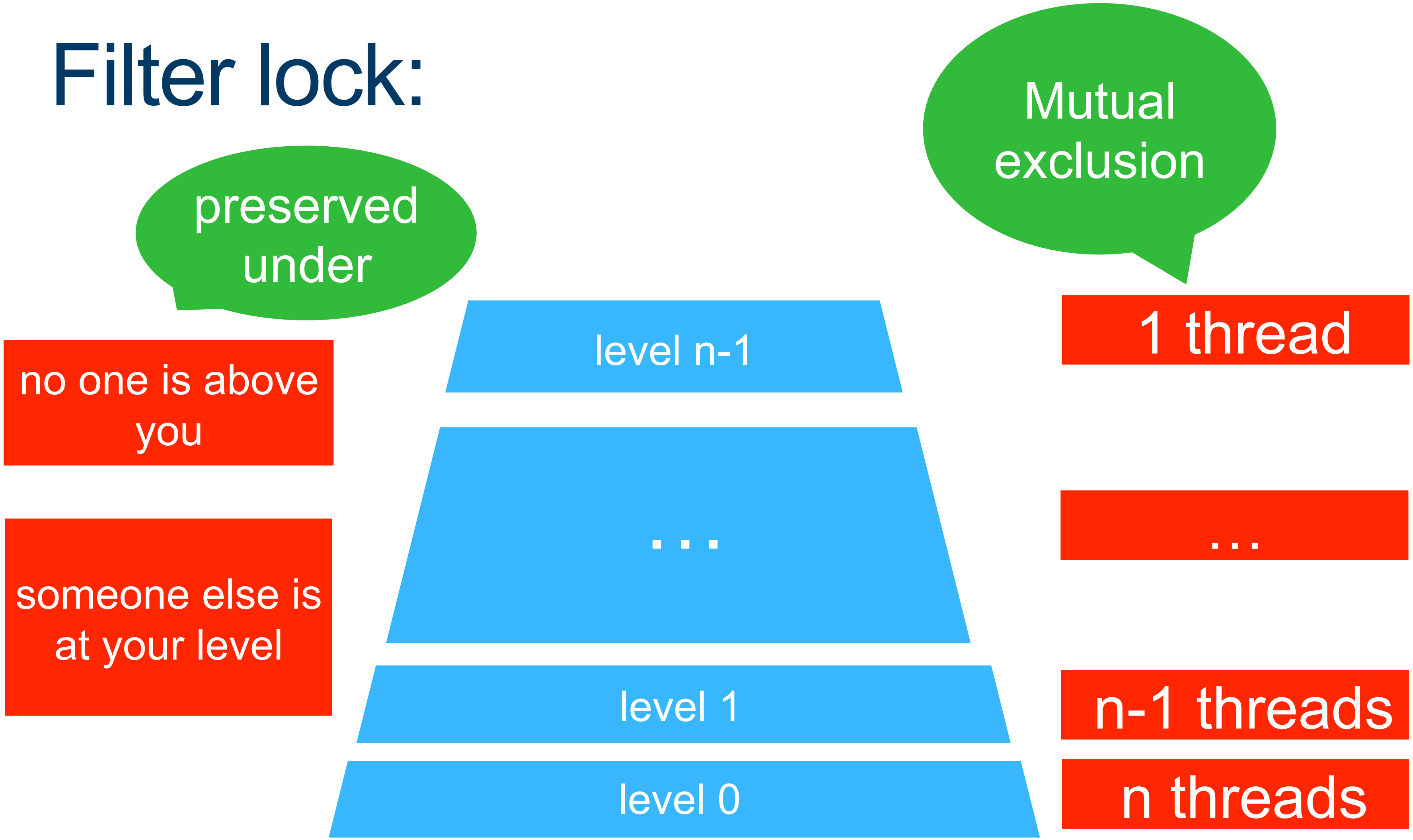
Quantify over level

count processes at level

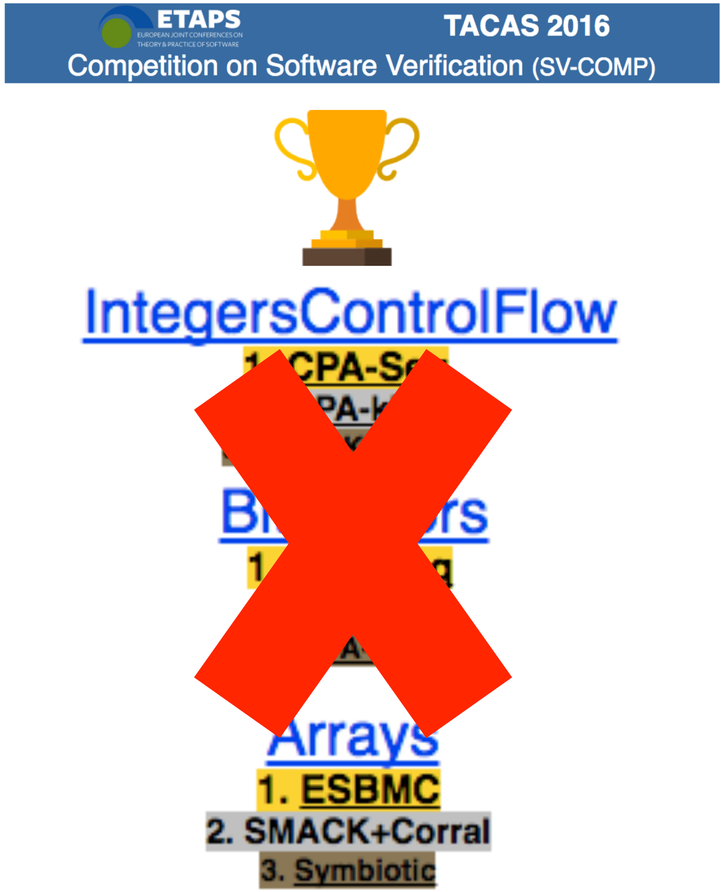
$$\forall l: 0 \le l \le n-1 \rightarrow \#\{t \mid lv(t) \ge l\} \le n-l$$

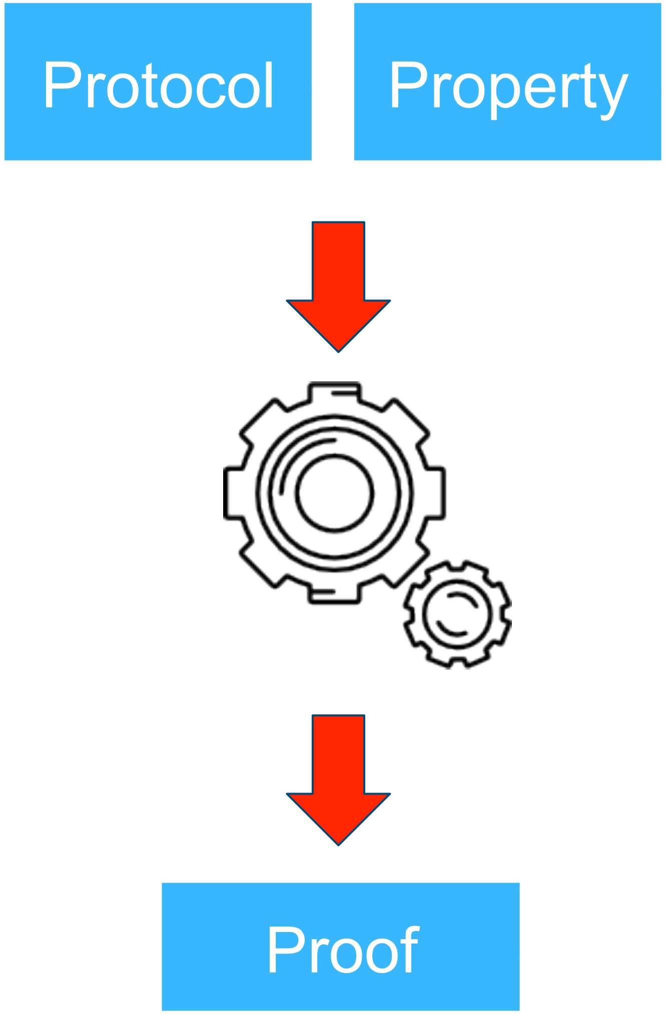
Infinitely many cardinalities

Filter lock:


preserved under
Mutual exclusion
no one is above you
someone else is at your level
level n-1
. . .
level 1
level 0
1 thread
. . .
n-1 threads
n threads

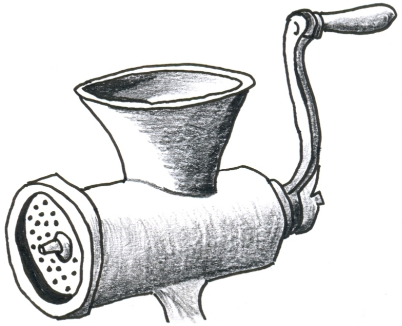
Program Verifiers:





In this talk:

Sharpie



#π

Verifies properties show before

1: What to count

2: How to do the counting

A simple example:

Some number n of threads

```
    global int a=0;
1:  a++;
2:
```

If there is a thread at location 2, then a>0

Example: in logic

Local variables as functions

initial states: $\forall t: pc(t)=1 \wedge a=0$

primed = after transition

transition relation: $pc(me)=1 \wedge pc' := pc(me \leftarrow 2) \wedge a'=a+1$

safety: $pc(me)=2 \rightarrow a>0$

Example: constraints

Horn clauses

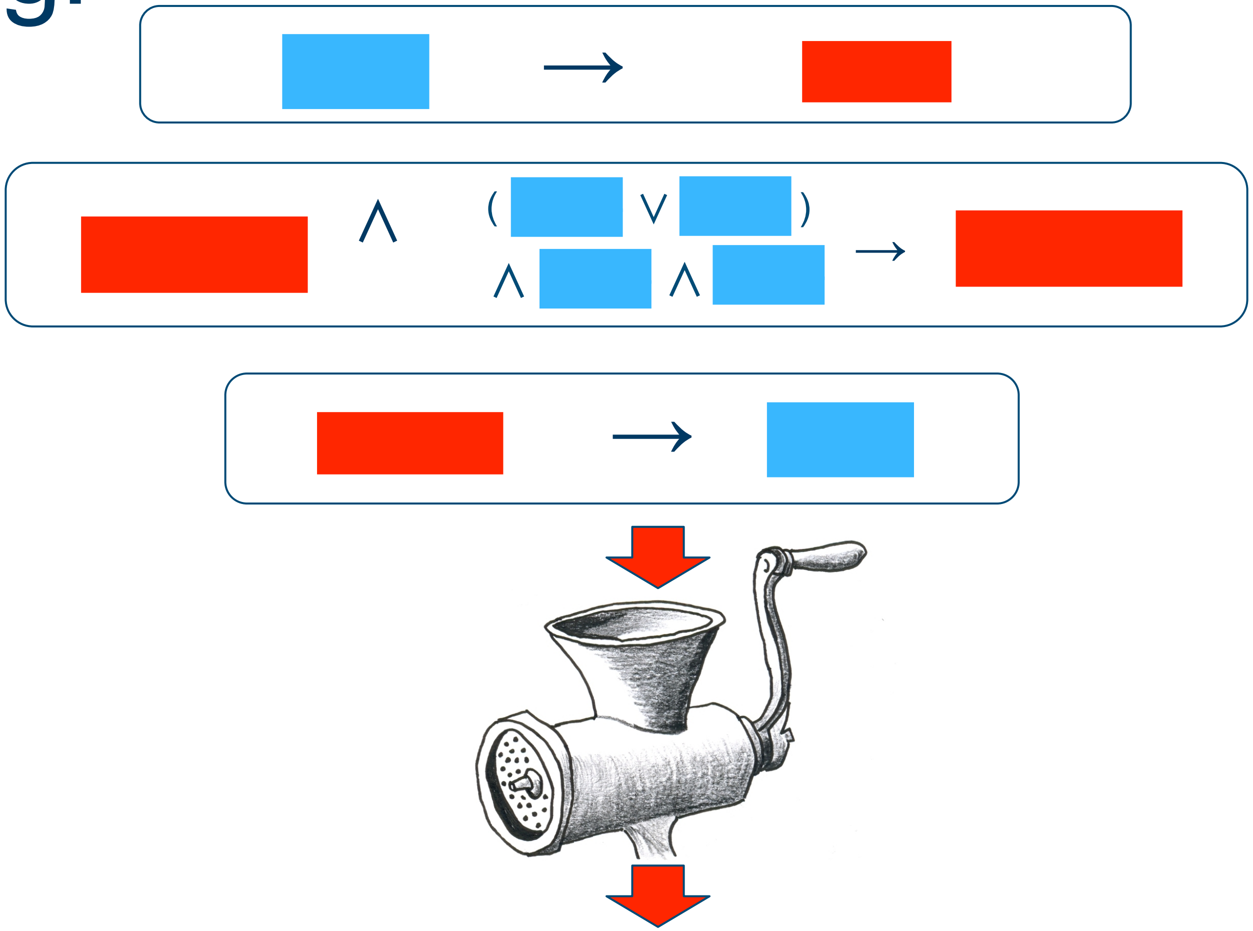
find: inv

$\forall t$: pc(t)=1 $\wedge$ a=0 $\rightarrow$ inv(a,pc)

inv(a,pc) $\wedge$ pc(me)=1 $\wedge$ pc':=pc(me$\leftarrow$2) $\wedge$ a'=a+1 $\rightarrow$ inv(a',pc')

inv(a,pc) $\rightarrow$ pc(me)=2 $\rightarrow$ a>0

Solving:



$\#\{t \mid pc(t)>1\} \leq a$

How to count: invariant checking

Checking: not supported by existing method

∀t: pc(t)=1

#{t | pc(t)>1}≤a → pc(me)=2 → a>0

How to count: invariant checking

Set is empty

$\forall t: pc(t)=1 \wedge a=0 \quad \rightarrow \quad 0 \leq 0$

Set is non-empty

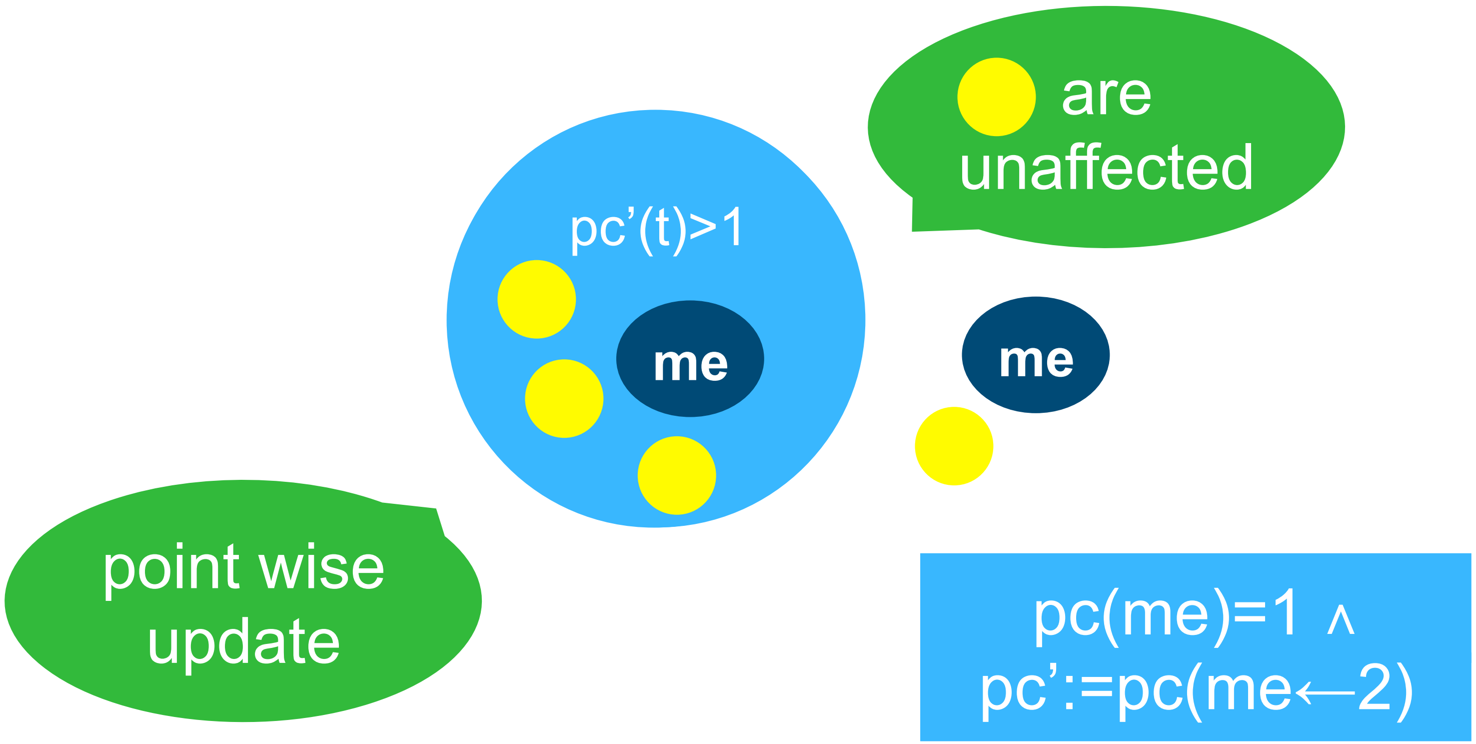
$1 \leq a \quad \rightarrow \quad pc(me)=2 \rightarrow a>0$

Example: point wise update

How to deal with upd?

$\#\{t \mid pc(t)>1\} \leq a$ $\wedge$ $pc(me)=1 \wedge pc' := pc(me \leftarrow 2) \wedge a'=a+1$ $\rightarrow$ $\#\{t \mid pc'(t)>1\} \leq a'$

Example: point wise update



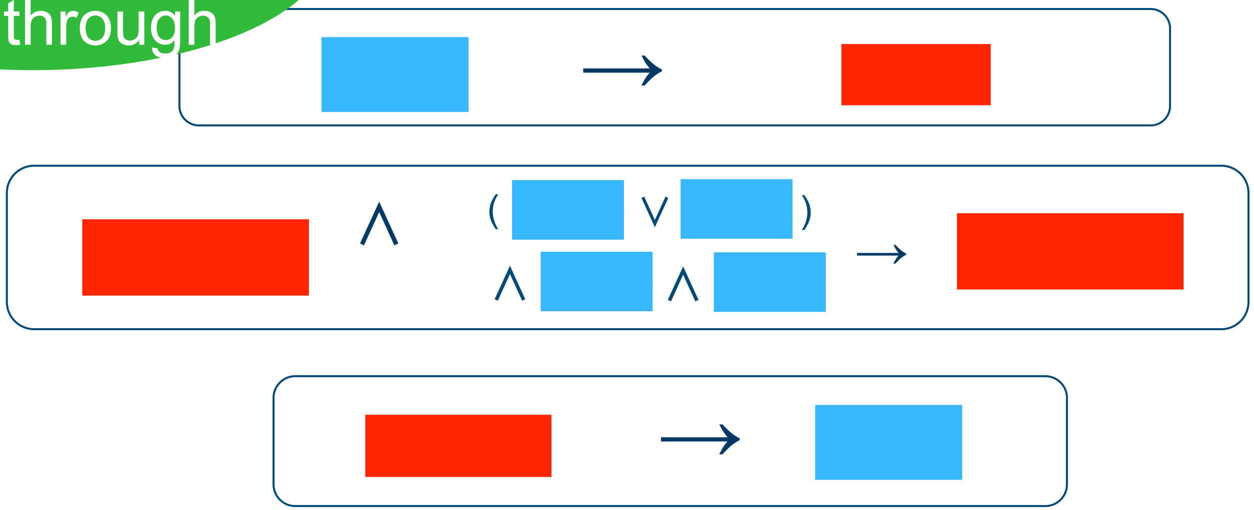
What to count

Find s and inv

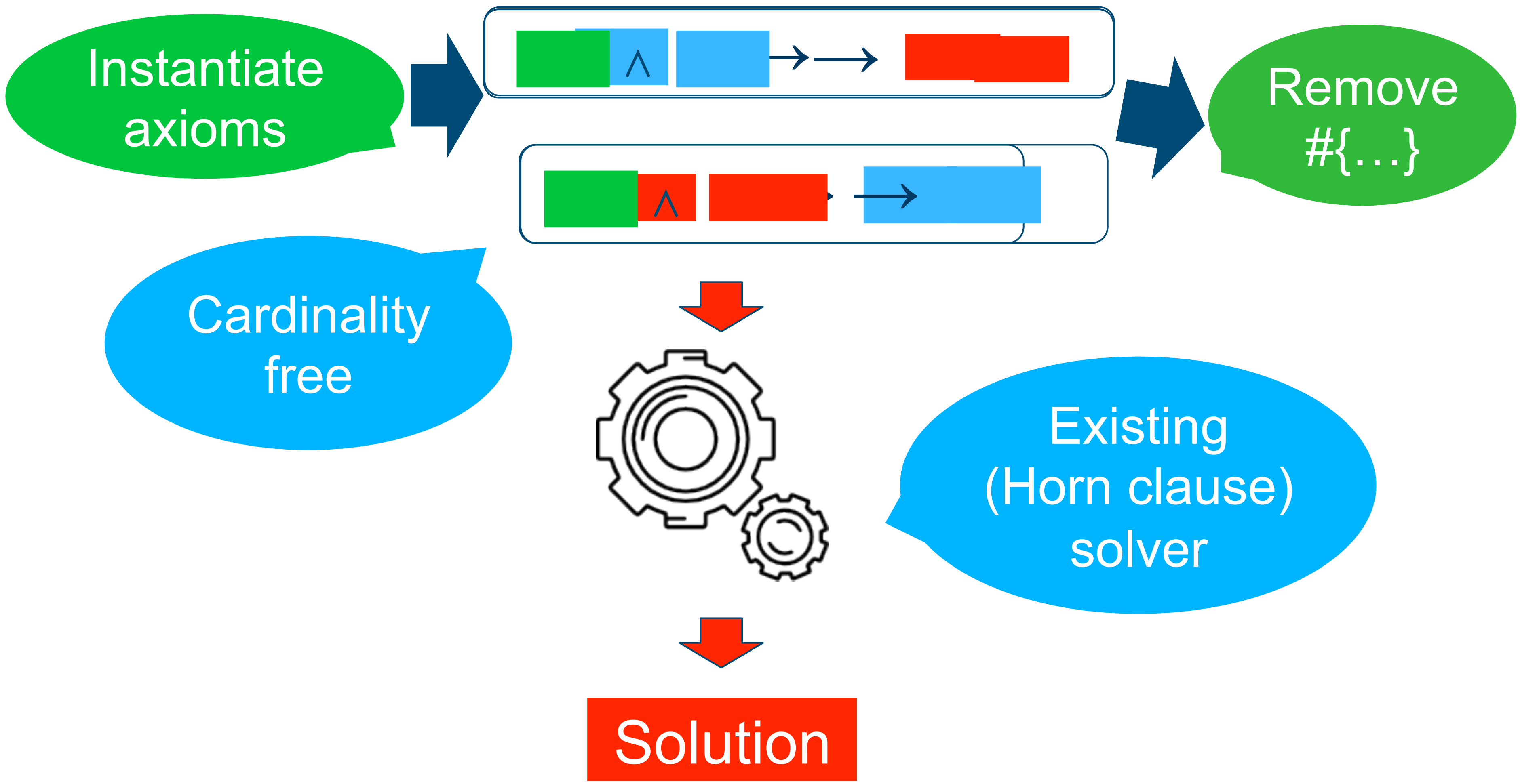
cardinality free

∃ # inv ∧ inv(a,pc,k)

#π chooses s s.t. the proof goes through



Cardinality axioms:

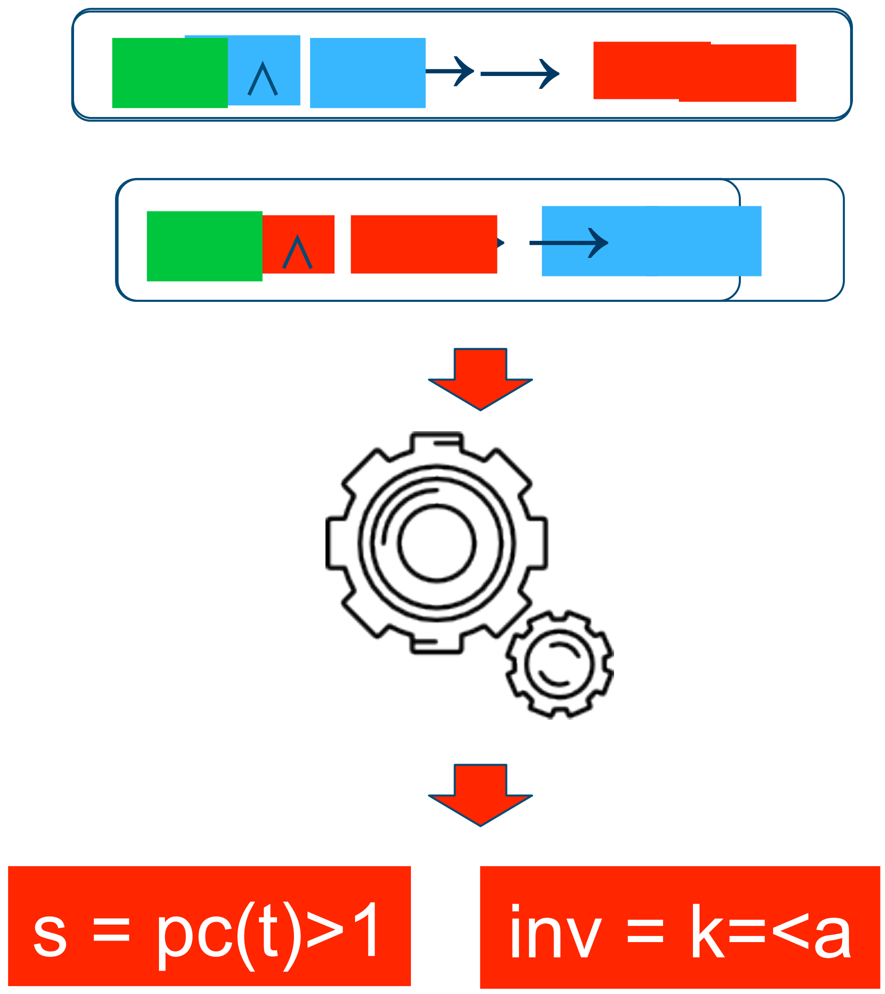


Example: finding the solution

$\#\{t \mid s(t)\}=k$

$\forall t: pc(t)=1 \wedge a=0$ $\quad\wedge\quad$ $(\forall t: \neg s(t)) \rightarrow k=0$ $\quad\rightarrow\quad$ inv(a,pc,k)

Cardinality axioms:



Evaluation:

Works on problems from the literature

cache coherence

garbage collection

Program	Card	Property	Inferred cardinalities	Time
intro [21]	✓	$(\exists t : pc(t) = 2) \rightarrow b < a$	$\#\{t \mid pc(t) = 2\}$	1.2s
bluetooth [21]	✓	$(\exists t : pc(t) = 2) \rightarrow st = 0$	$\#\{t \mid pc(t) = 2\}$	1.6s
tree traverse [21]	[illegible]	[illegible] $des + 1$	–	4.2s
cache [59]	[illegible]	[illegible] $(t) = 3\} \leq 1$	$\#\{t \mid pc(t) \geq 3\}$	0.7s
garbage collection	✓	$\#\{t \mid 2 \leq pc(t) \leq 4\} \leq 1 \land m = 1$	$\#\{t \mid 2 \leq pc(t) \leq 4\}$	10.1s

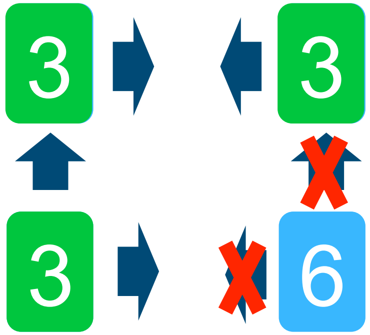
locking

consensus

[illegible]	Property	Inferred cardinalities	Time
ticket lock [21]	[illegible] $3\} \leq 1$	$\#\{t \mid m(t) \leq s \land pc(t) = 2\}$, $\#\{t \mid pc(t) = 3\}$ $\#\{t \mid m(t) = q\}$	20.9s
filter lock [31]	$\#\{t \mid lv(t) = n - 1\} \leq 1$	$\#\{t \mid lv(t) \geq q\}$	27.5s
one-third rule [14, 18]	see Section 2	$\#\{t \mid x(t) = x(q)\}$	0.8s

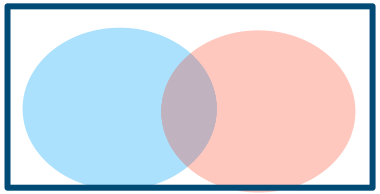
First automated cardinality proofs

Conclusion:



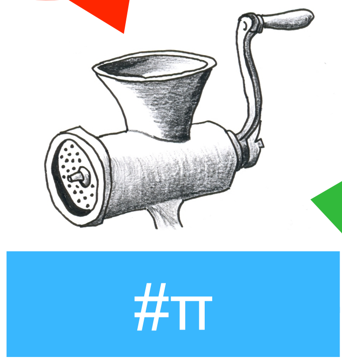
Verifying protocols requires cardinalities

Existing verifiers can't count



Sharpie

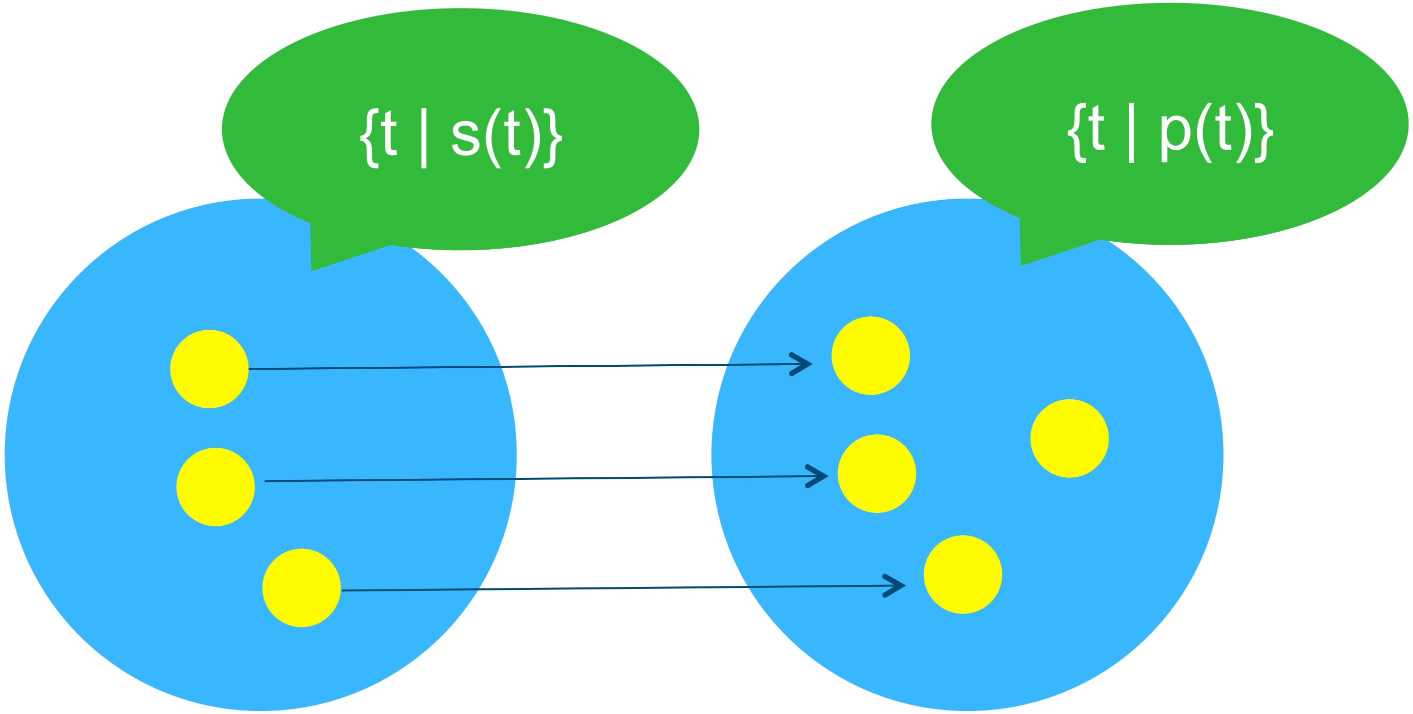
Knows how to count



Chooses what to count to complete proof

That’s it!

Axioms: inequality



$(\forall t{:}s(t) \rightarrow p(t)) \rightarrow k \leq l$

Axioms: inequality

Equality comparison

$$\frac{\#\{t \mid s(t)\}=k \qquad \#\{t \mid p(t)\}=l}{(\forall t{:}s(t)\longrightarrow p(t))\longrightarrow k\leq l}$$

Axioms: strict inequality

$$\frac{\#\{t \mid s\}=k \qquad \#\{t \mid p\}=l}{\begin{array}{l}(\forall t{:}\, s(t) \rightarrow p(t)) \wedge \\ (\exists t{:}\, \neg s(t) \wedge p(t))\end{array} \rightarrow k<l}$$

Additional witness