Tackling Concurrency Bugs with TLA+

Hillel Wayne

@hillelogram

hillelwayne.com

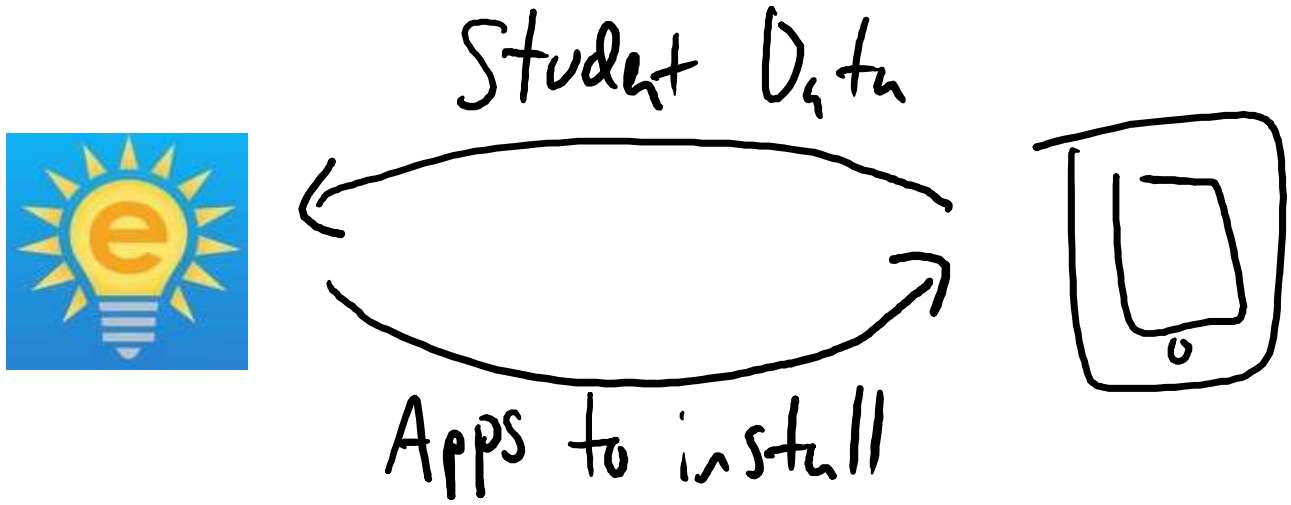

#tlaplus

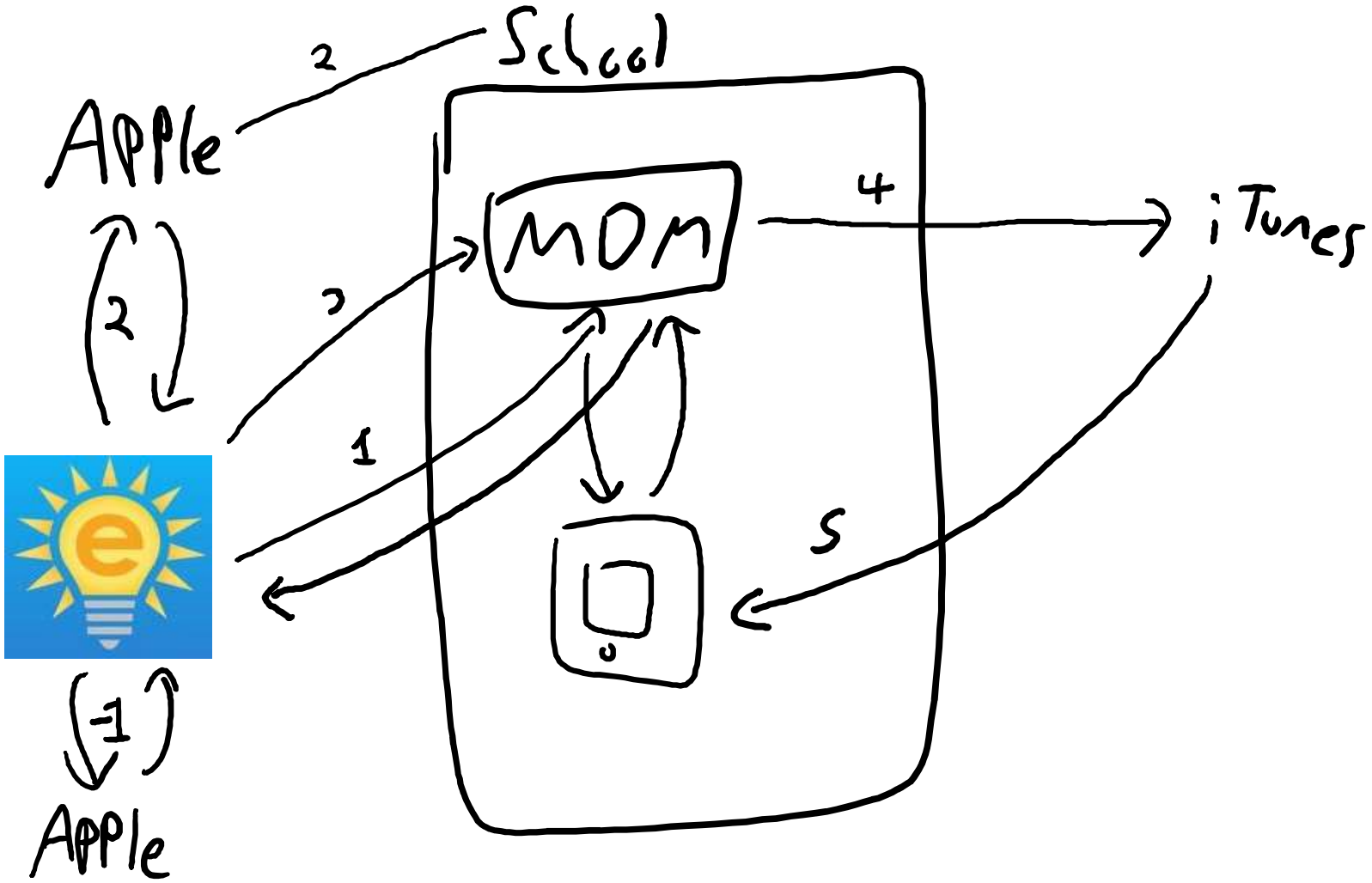
Hillel Wayne

ESPARK LEARNING

Speaker site
@hillelogram
hwayne

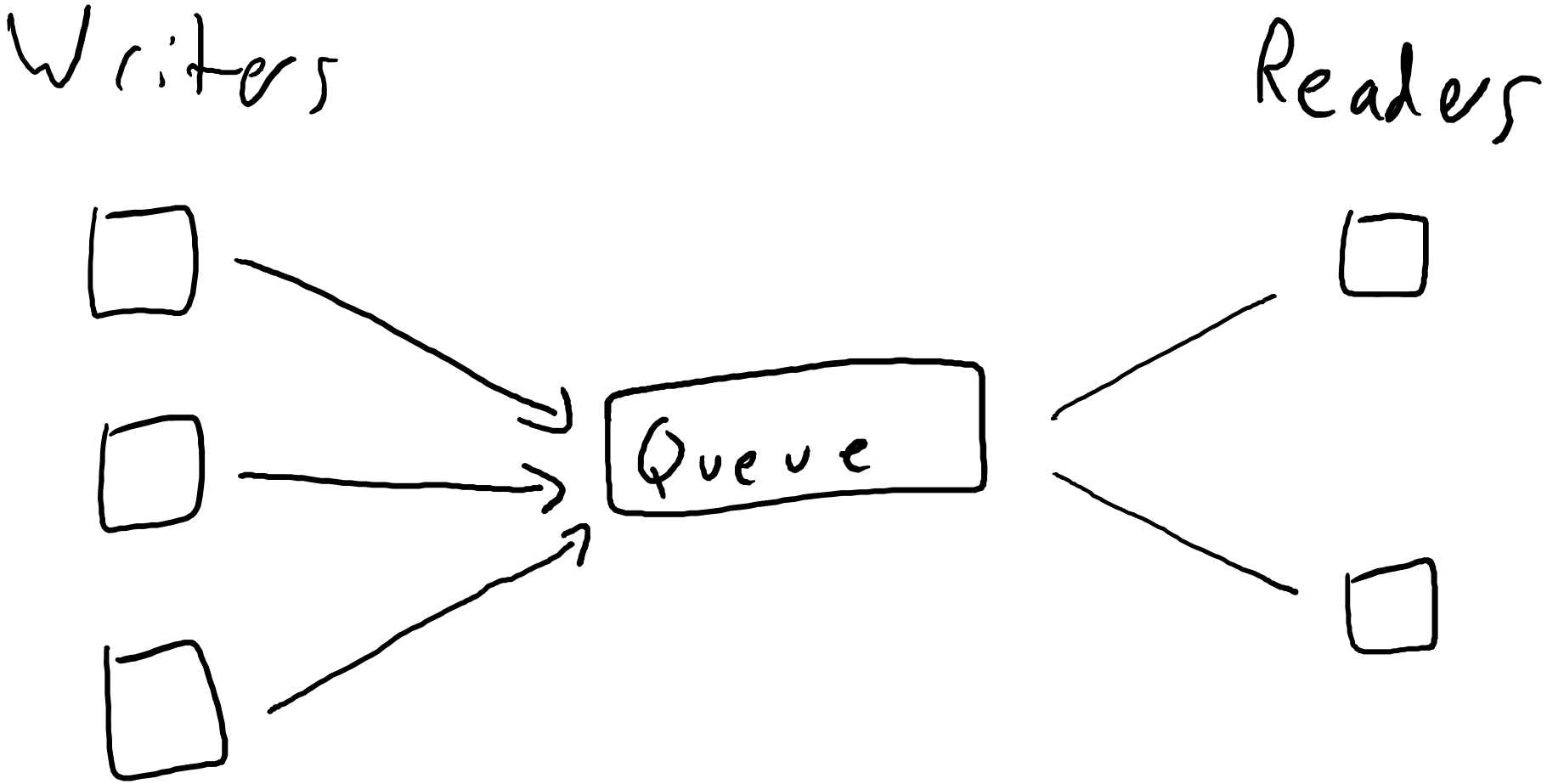
Hillel is a senior software engineer at eSpark Learning. He enjoys technical writing and is the author of Learn TLA+ (https://learntla.com). In his free time he juggles and makes chocolates. He _probably_ brought enough for everyone.

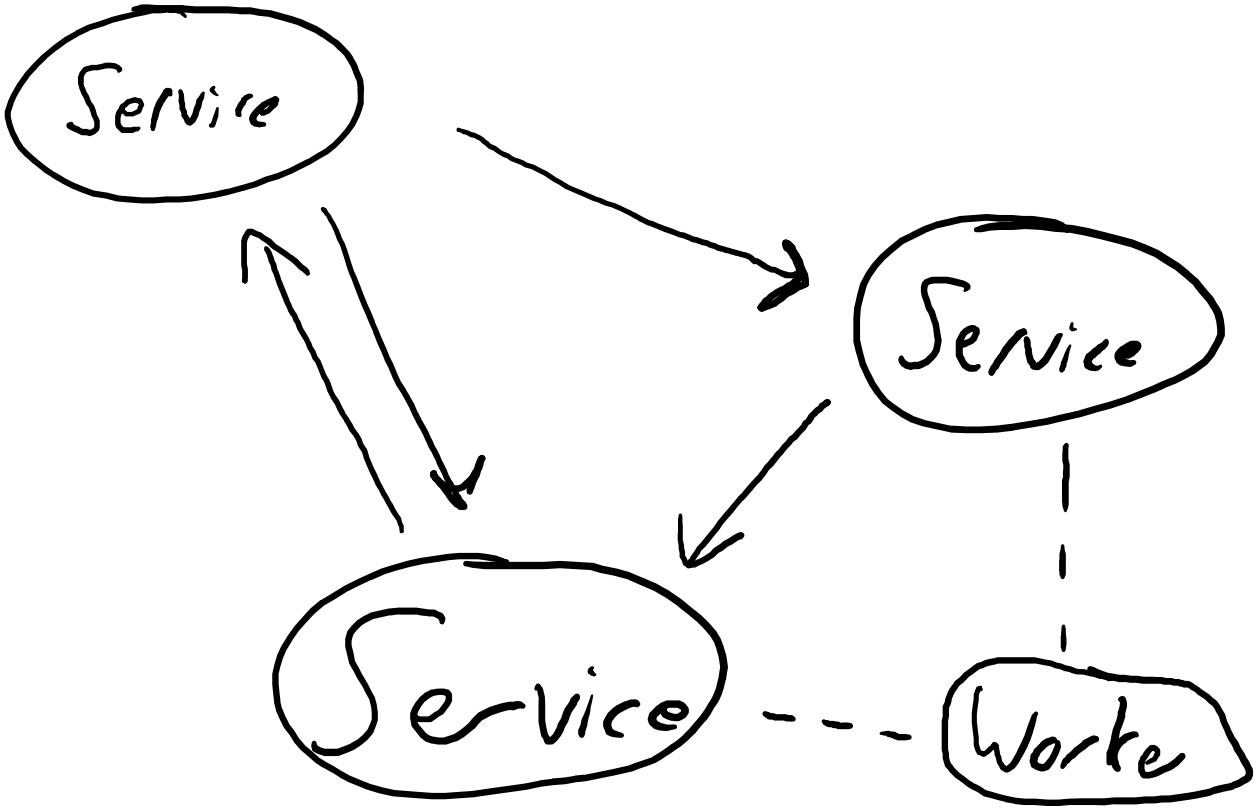

Student Data
Apps to install


School
Apple
2
MOM
4
iTunes
3
2
1
5
-1
Apple

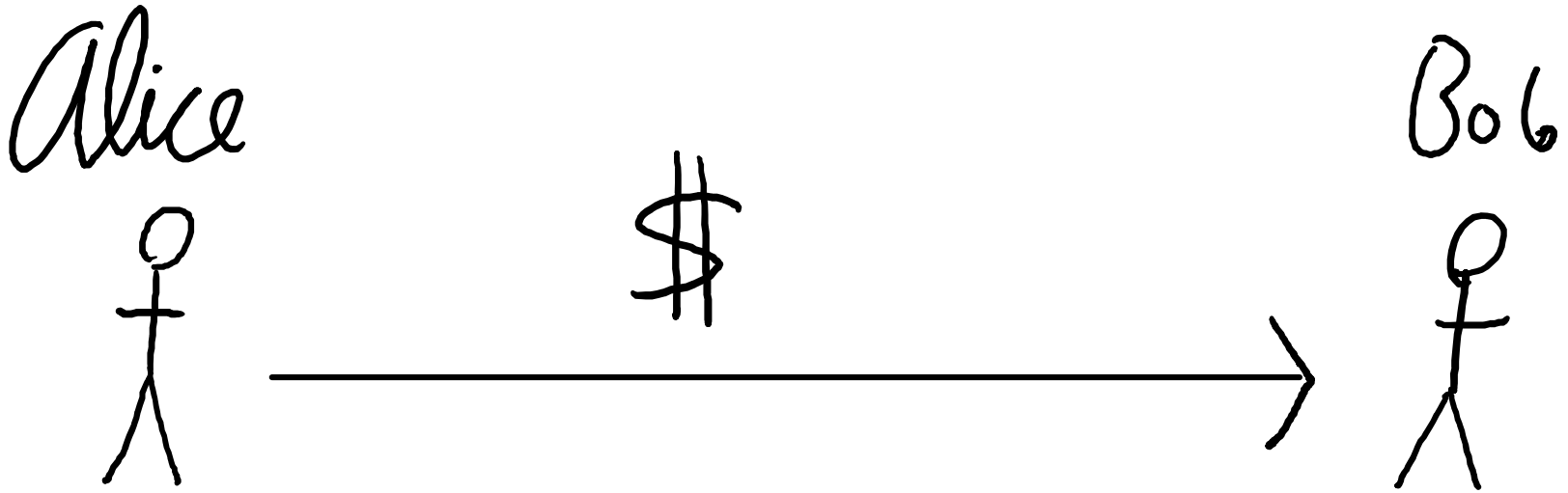
Concurrency

- Multiple Systems
- Running Independently
- With some global state
- **Nondeterministic**

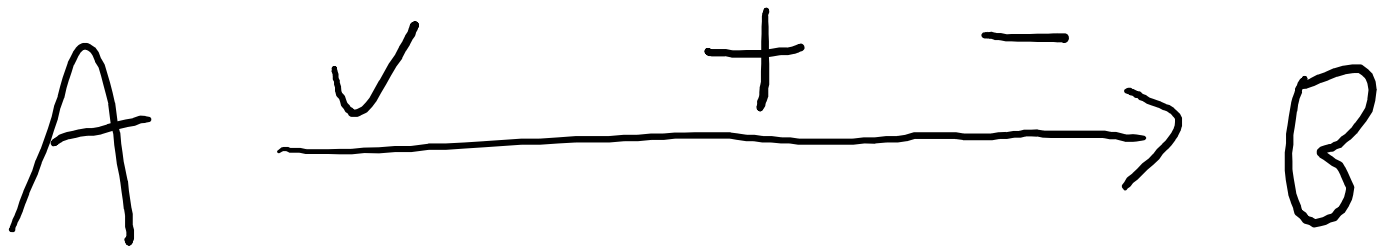

Writers
Readers
Queue


Service
Service
Service
Worke

Example



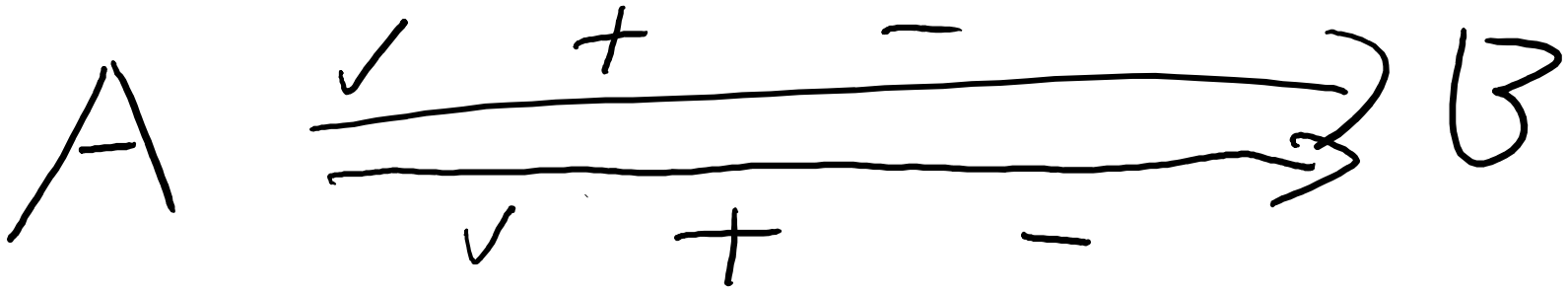
```
def transfer(amount, to)
  raise if self.balance < amount
  to.balance += amount
  self.balance -= amount
end
```



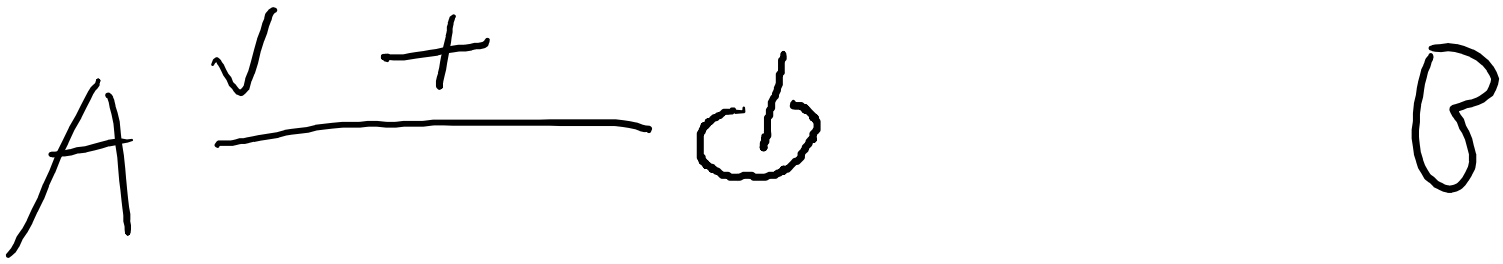
Carnival of Horrors

@hillelogram

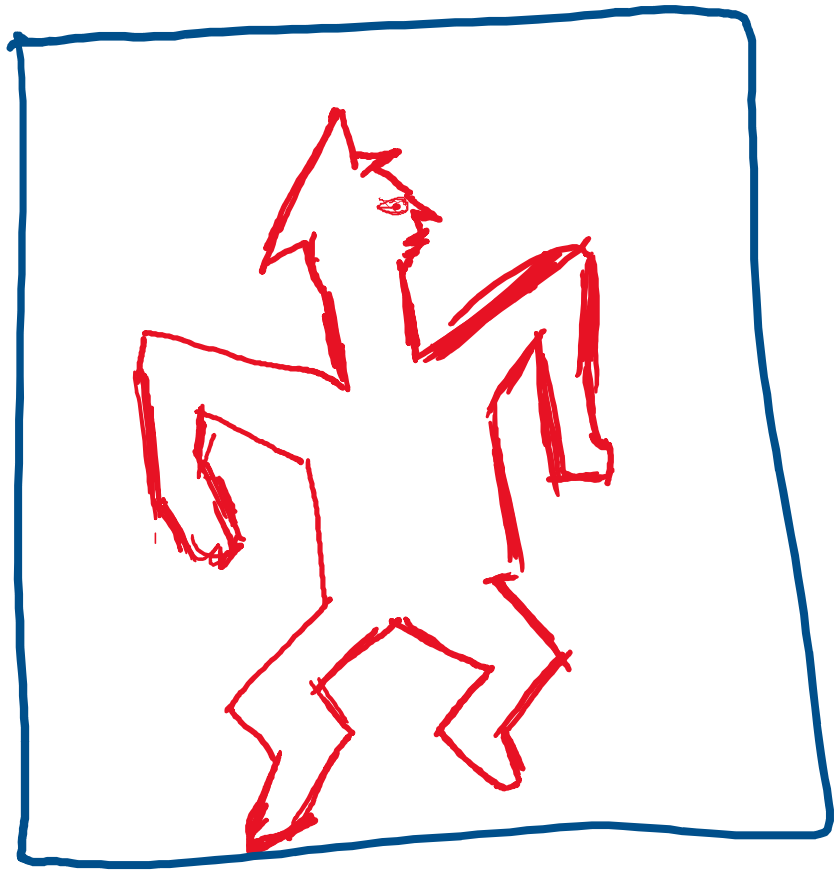
Race Conditions



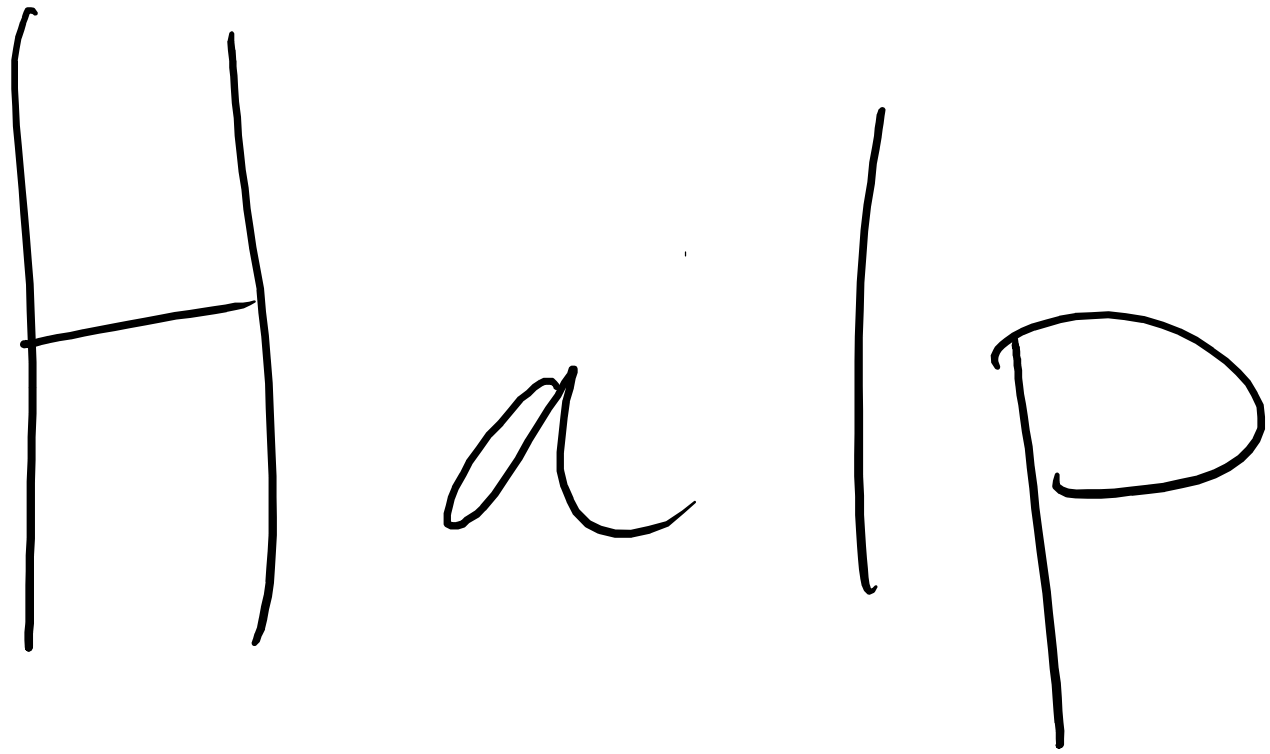
Crashes



Deadlocks



@hillelogram

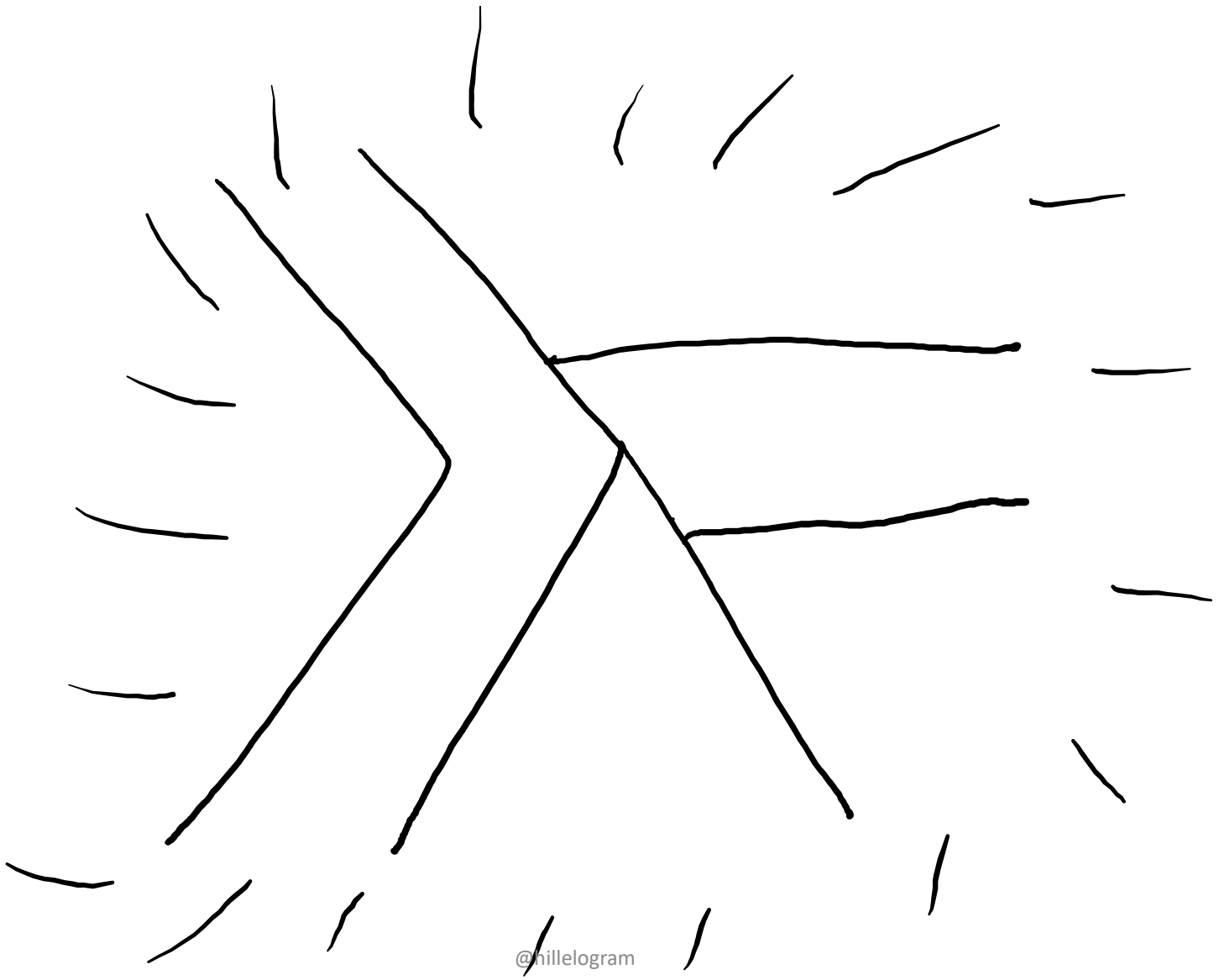

Halp

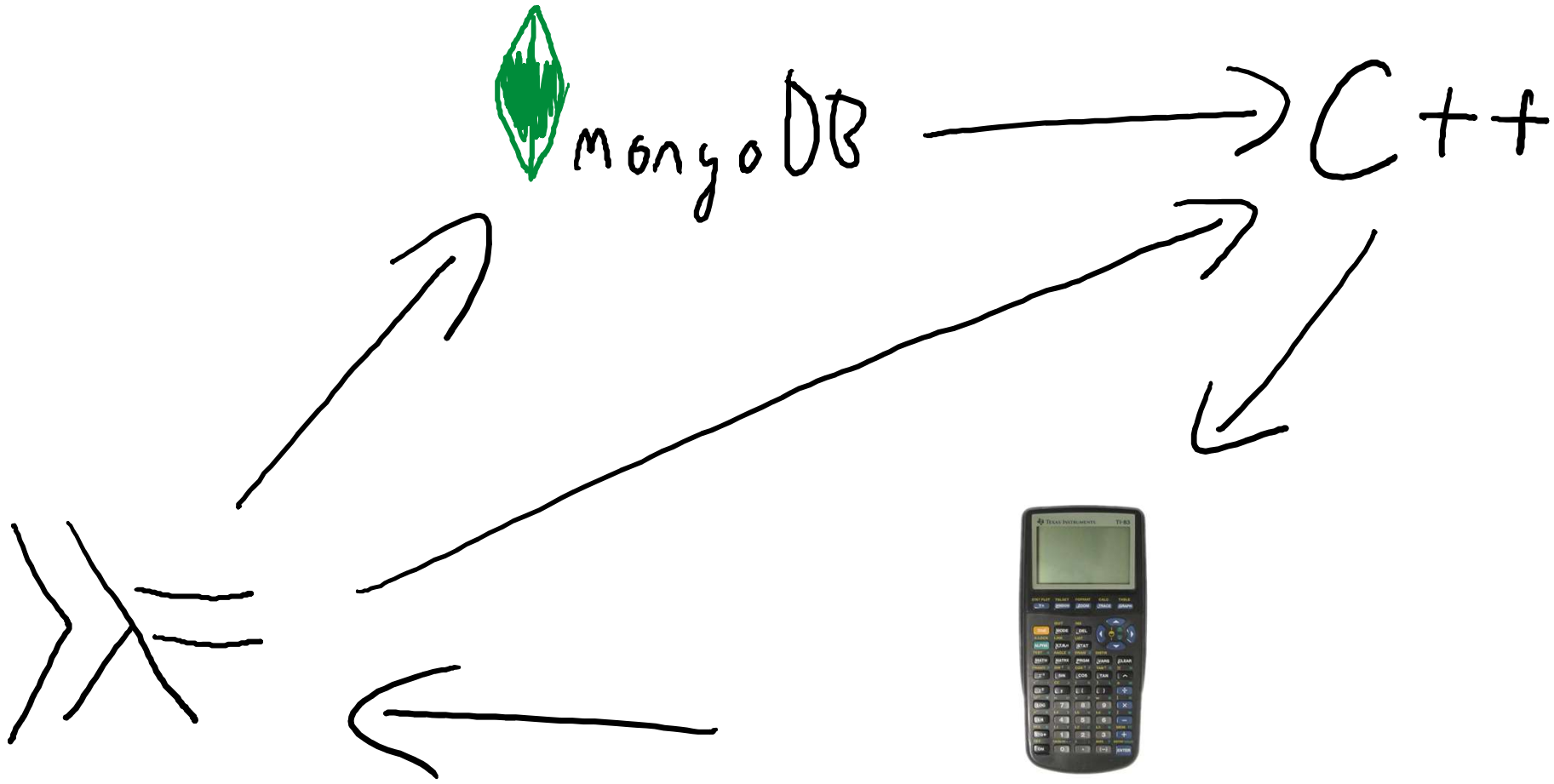
Unit Tests

When “I receive a message”
Then “it sends an alert”

When “I receive a message”
And “the message queue duplicated the message”
And “another worker receives the copy later but finishes first”
And "Github is down"
Then “it sends an alert”

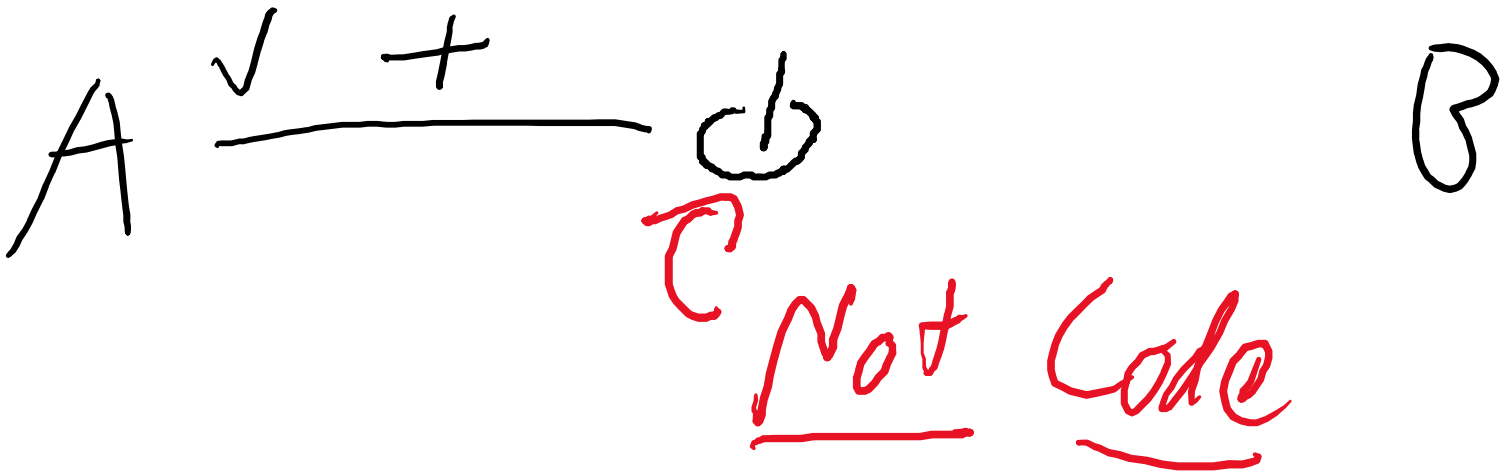
Static Typing


@hillelogram

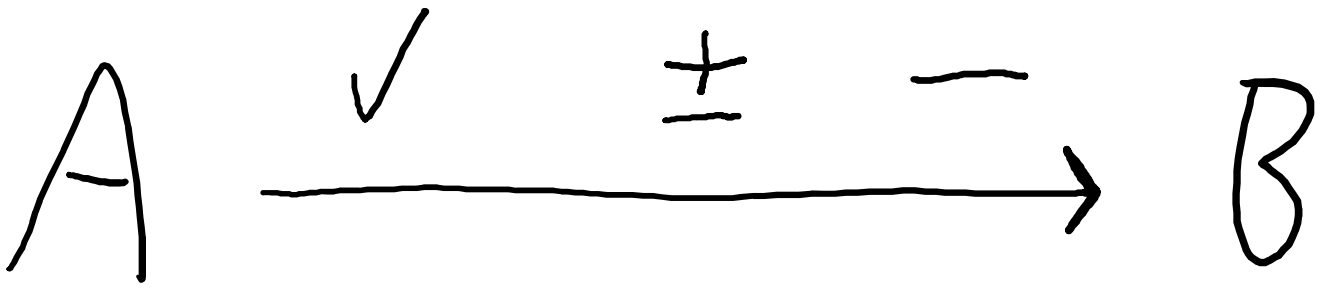


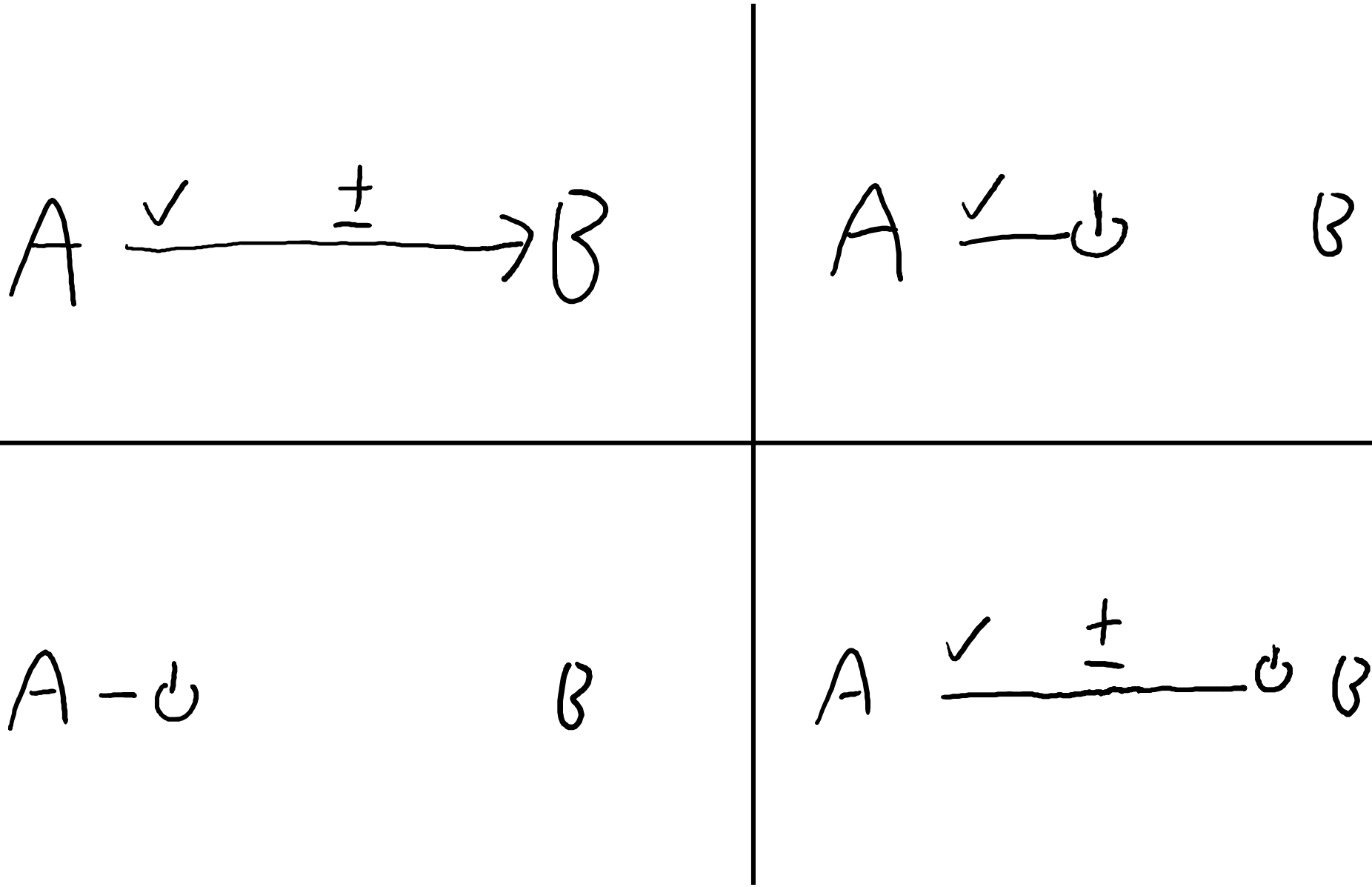
@hillelogram

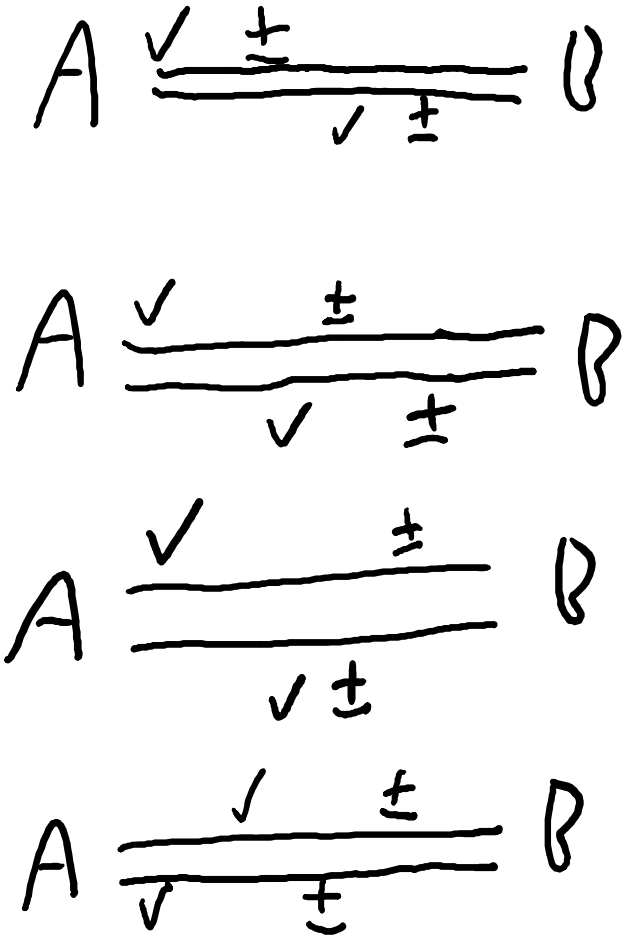
We don't get purity


A
√
+
B
Not Code

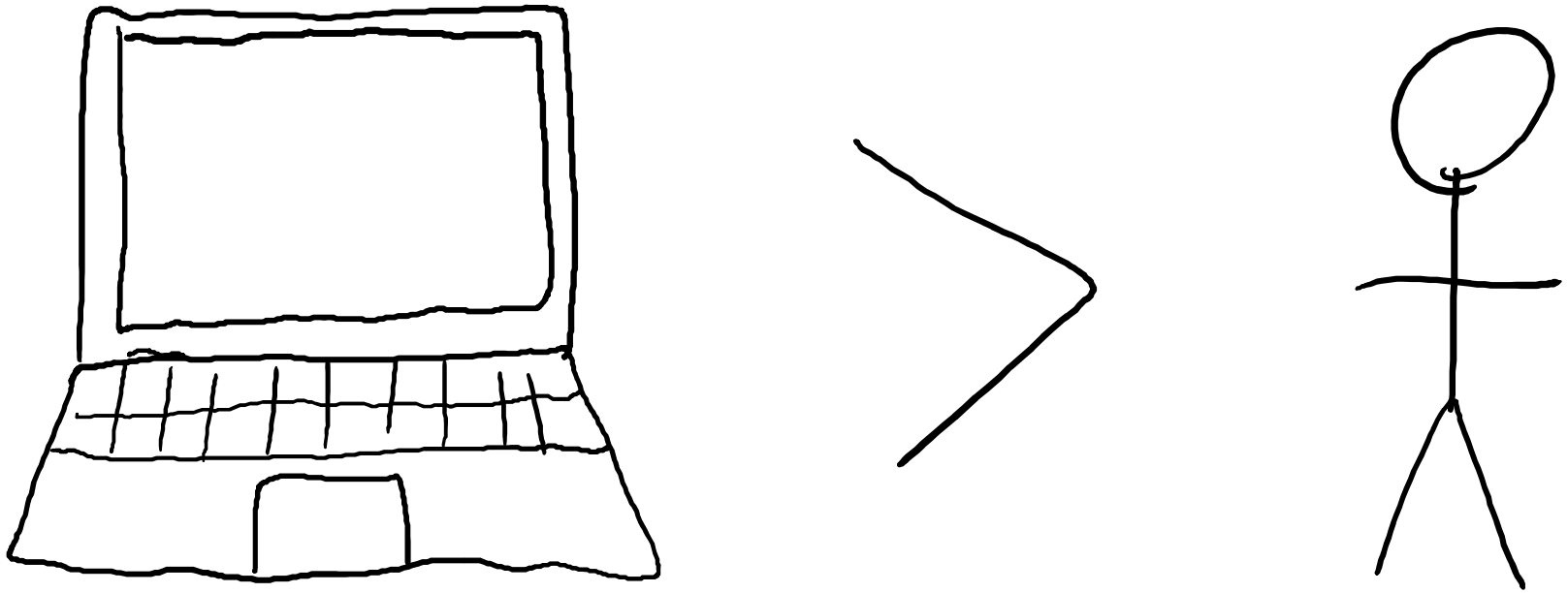
```
def transfer(amount, to)
  raise if self.balance < amount
  to.balance += amount
  self.balance -= amount
end
```




A
B
A
B
A
B
A
B



Chris Hastings, www.drmcninja.com



@hillelogram

Formal Specification

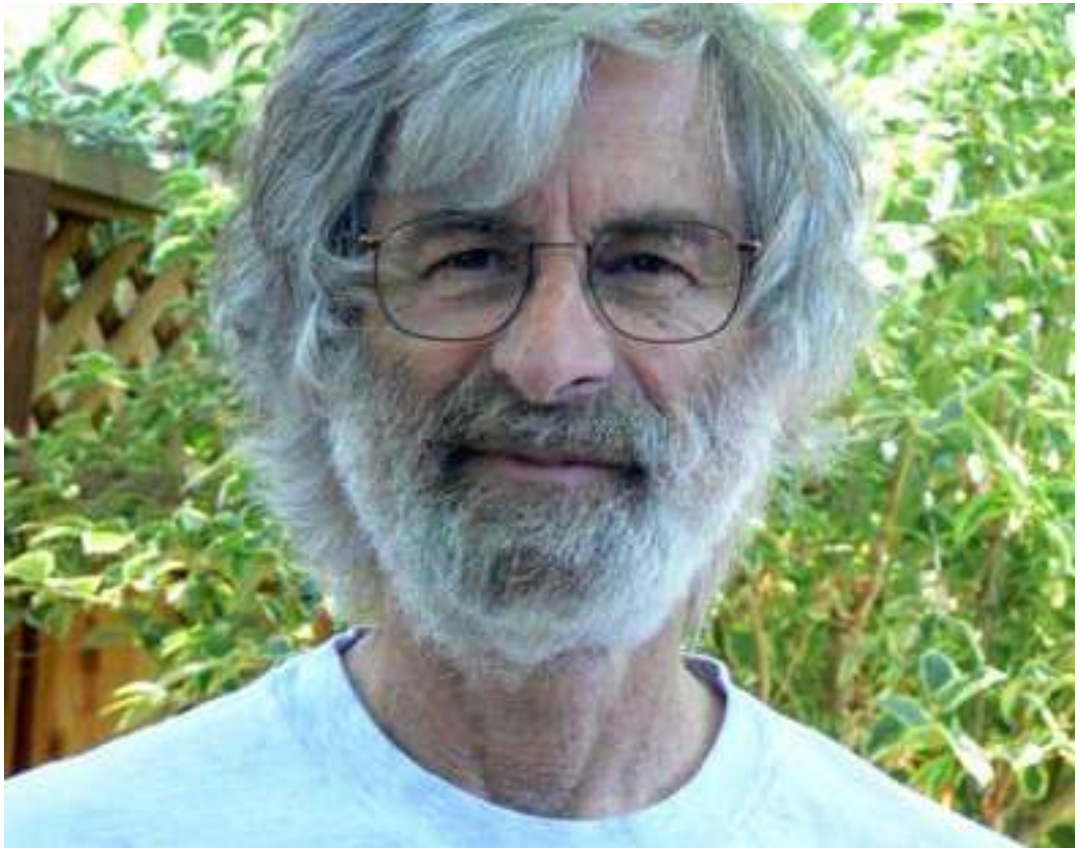
@hillelogram

Specs do not
check code


TLA+
@hillelogram

Things Lamport did

- TLA+
- Paxos
- Byzantine Generals
- Digital signatures
- Sequential Consistency
- LaTeX



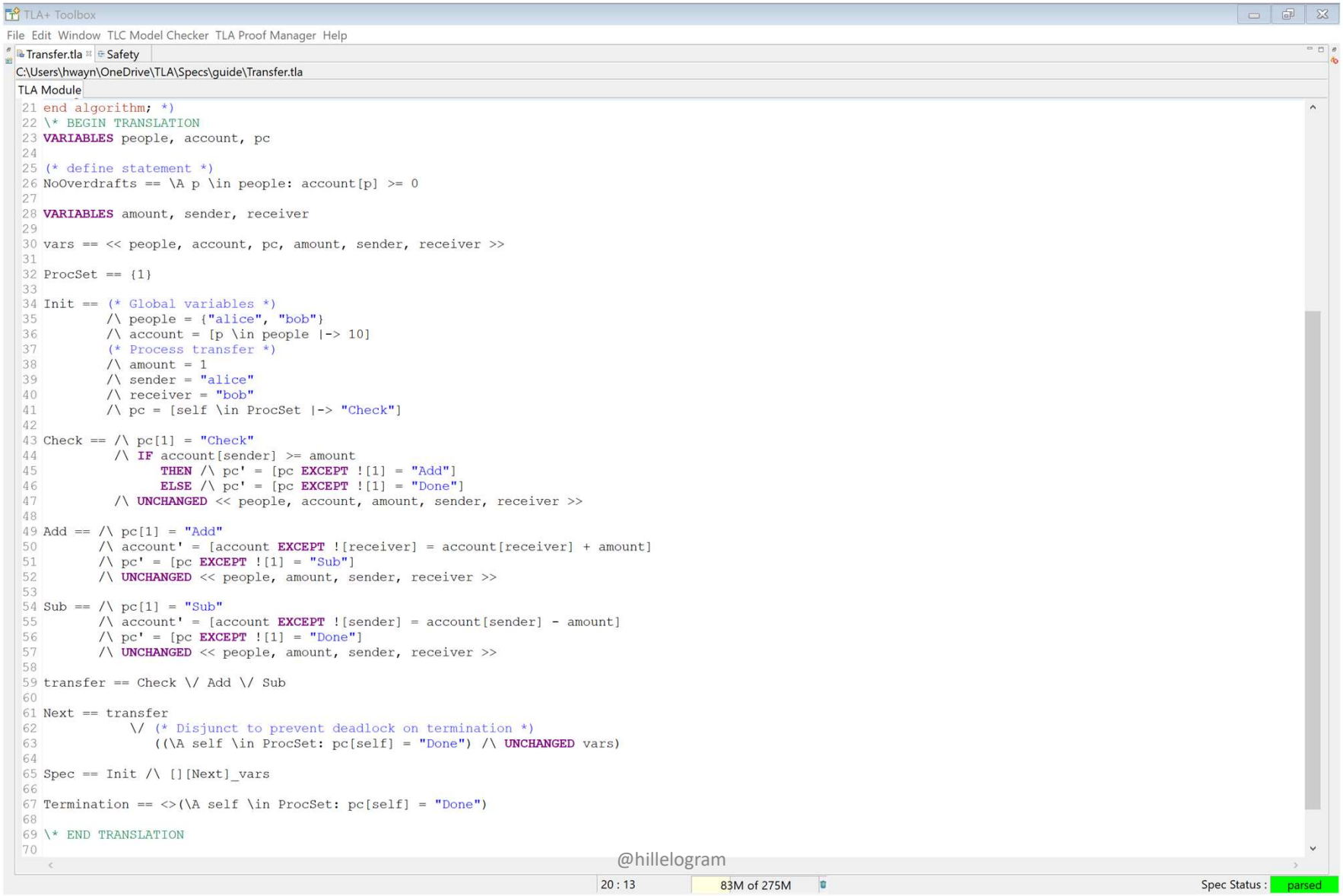
Why TLA+:

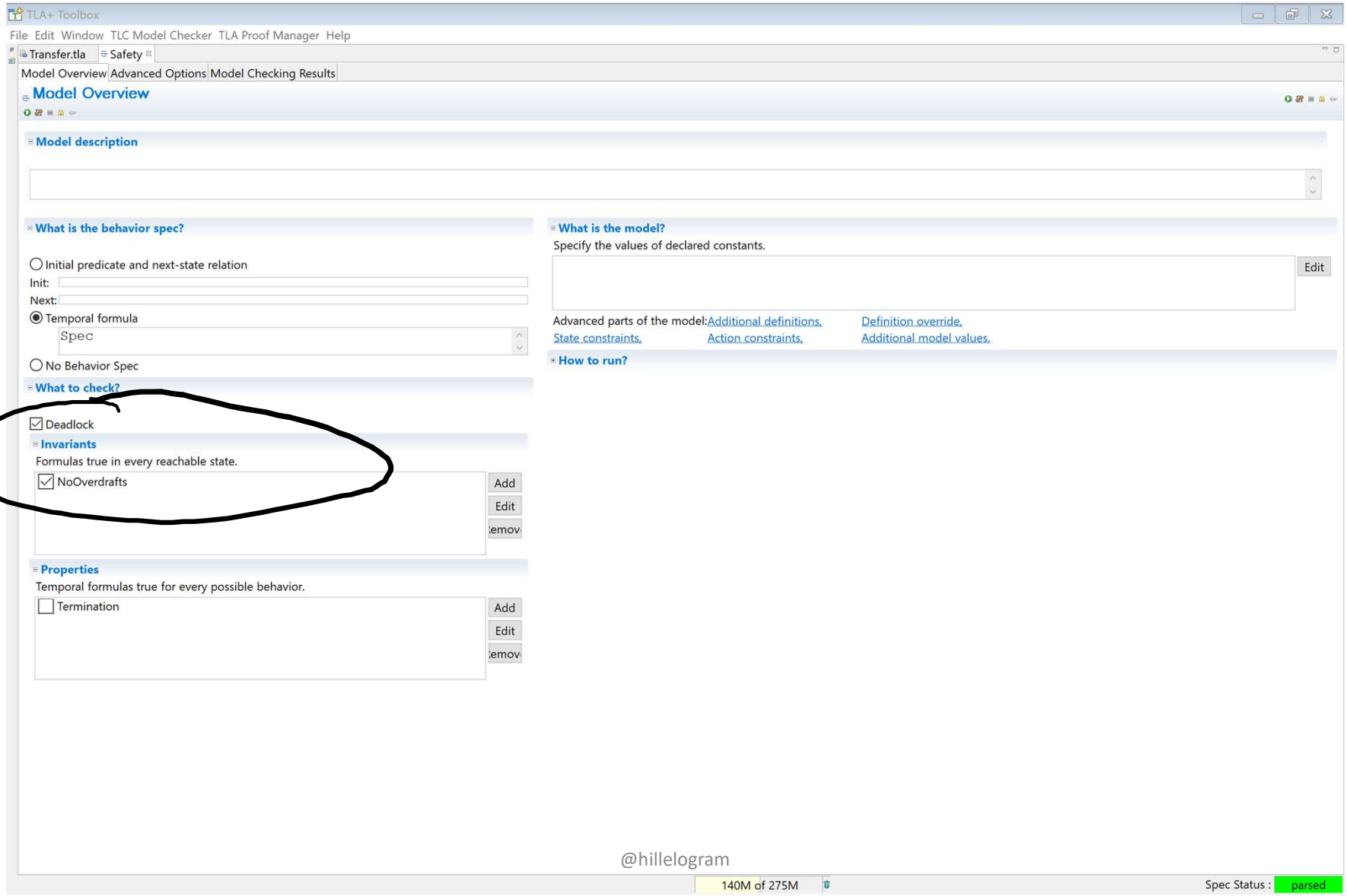
- Math
- Jargon
- Blah blah blah

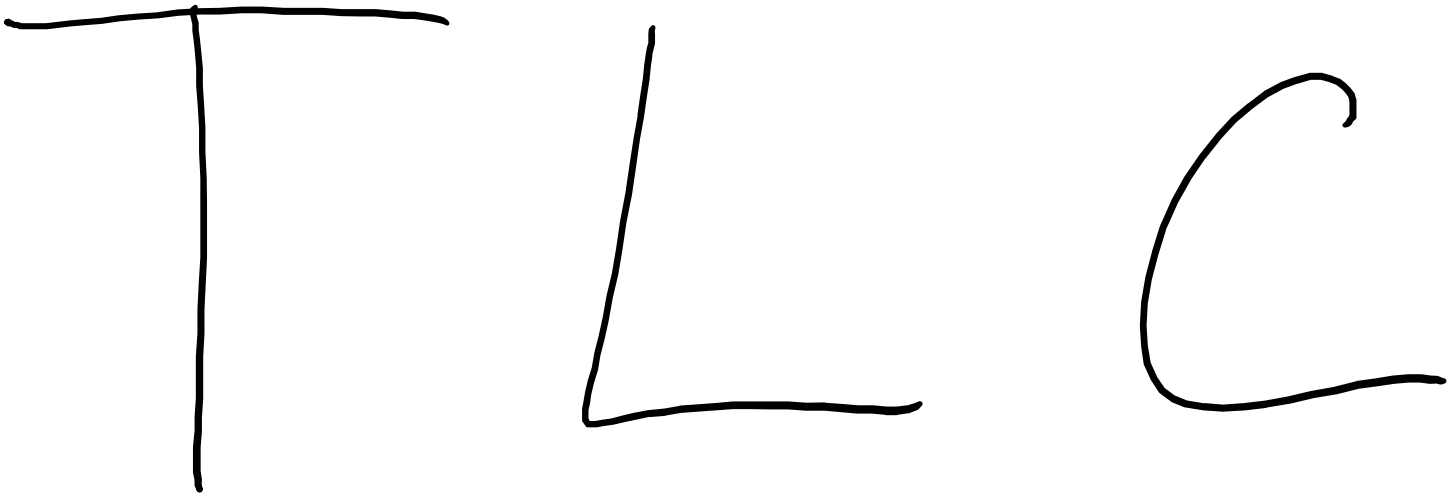
```
variables alice_account = 10, bob_account = 10;

\* Our algorithm goes here

NoOverdrafts == alice_account >= 0 /\ bob_account >= 0
```

```
\* ~~variables alice_account = 10, bob_account = 10;~~
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

\* account["alice"] = 10
\* stuff goes here

NoOverdrafts == \A p \in people: account[p] >= 0
```

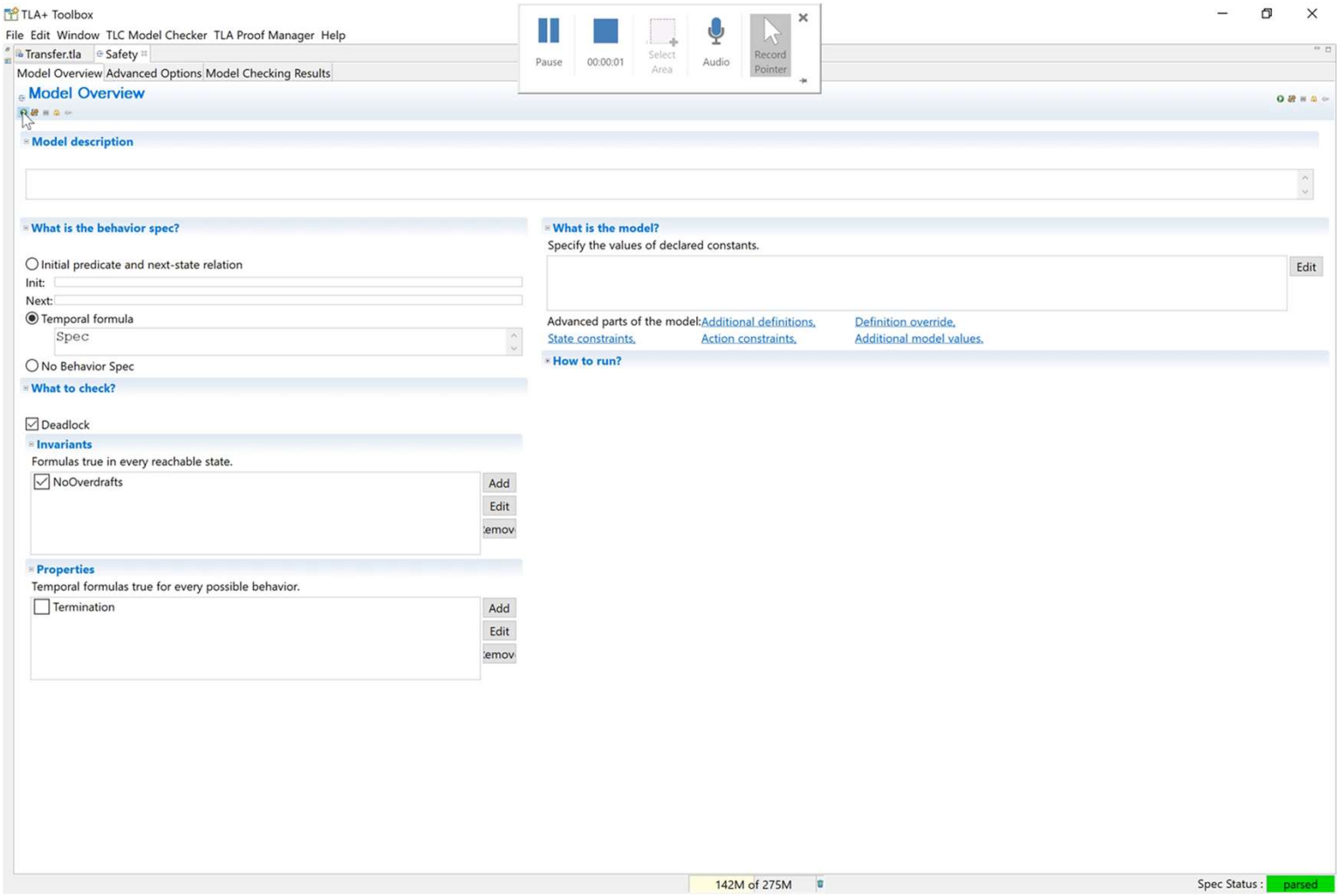
```
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

process transfer = 1
  variables amount = 1,
            sender = "alice",
            receiver = "bob";
begin
  Check: if account[sender] >= amount then
    Add: account[receiver] := account[receiver] + amount;
    Sub: account[sender] := account[sender] - amount;
  end if;
end process;

NoOverdrafts == \A p \in people: account[p] >= 0
```


TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
Transfer.tla
Safety
C:\Users\hwayn\OneDrive\TLA\Specs\guide\Transfer.tla
TLA Module
21 end algorithm; *)
22 * BEGIN TRANSLATION
23 VARIABLES people, account, pc
24
25 (* define statement *)
26 NoOverdrafts == \A p \in people: account[p] >= 0
27
28 VARIABLES amount, sender, receiver
29
30 vars == << people, account, pc, amount, sender, receiver >>
31
32 ProcSet == {1}
33
34 Init == (* Global variables *)
35 /\ people = {"alice", "bob"}
36 /\ account = [p \in people |-> 10]
37 (* Process transfer *)
38 /\ amount = 1
39 /\ sender = "alice"
40 /\ receiver = "bob"
41 /\ pc = [self \in ProcSet |-> "Check"]
42
43 Check == /\ pc[1] = "Check"
44 /\ IF account[sender] >= amount
45 THEN /\ pc' = [pc EXCEPT ![1] = "Add"]
46 ELSE /\ pc' = [pc EXCEPT ![1] = "Done"]
47 /\ UNCHANGED << people, account, amount, sender, receiver >>
48
49 Add == /\ pc[1] = "Add"
50 /\ account' = [account EXCEPT ![receiver] = account[receiver] + amount]
51 /\ pc' = [pc EXCEPT ![1] = "Sub"]
52 /\ UNCHANGED << people, amount, sender, receiver >>
53
54 Sub == /\ pc[1] = "Sub"
55 /\ account' = [account EXCEPT ![sender] = account[sender] - amount]
56 /\ pc' = [pc EXCEPT ![1] = "Done"]
57 /\ UNCHANGED << people, amount, sender, receiver >>
58
59 transfer == Check \/ Add \/ Sub
60
61 Next == transfer
62 \/ (* Disjunct to prevent deadlock on termination *)
63 ((\A self \in ProcSet: pc[self] = "Done") /\ UNCHANGED vars)
64
65 Spec == Init /\ [][Next]_vars
66
67 Termination == <>(\A self \in ProcSet: pc[self] = "Done")
68
69 * END TRANSLATION
70
@hillelogram
20 : 13
83M of 275M
Spec Status : parsed


TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
Transfer.tla
Safety
Model Overview
Advanced Options
Model Checking Results
Model Overview
Model description
What is the behavior spec?
Initial predicate and next-state relation
Init:
Next:
Temporal formula
Spec
No Behavior Spec
What to check?
Deadlock
Invariants
Formulas true in every reachable state.
NoOverdrafts
Add
Edit
Properties
Temporal formulas true for every possible behavior.
Termination
Add
Edit
What is the model?
Specify the values of declared constants.
Edit
Advanced parts of the model: Additional definitions, Definition override,
State constraints, Action constraints, Additional model values.
How to run?
@hillelogram
140M of 275M
Spec Status : parsed

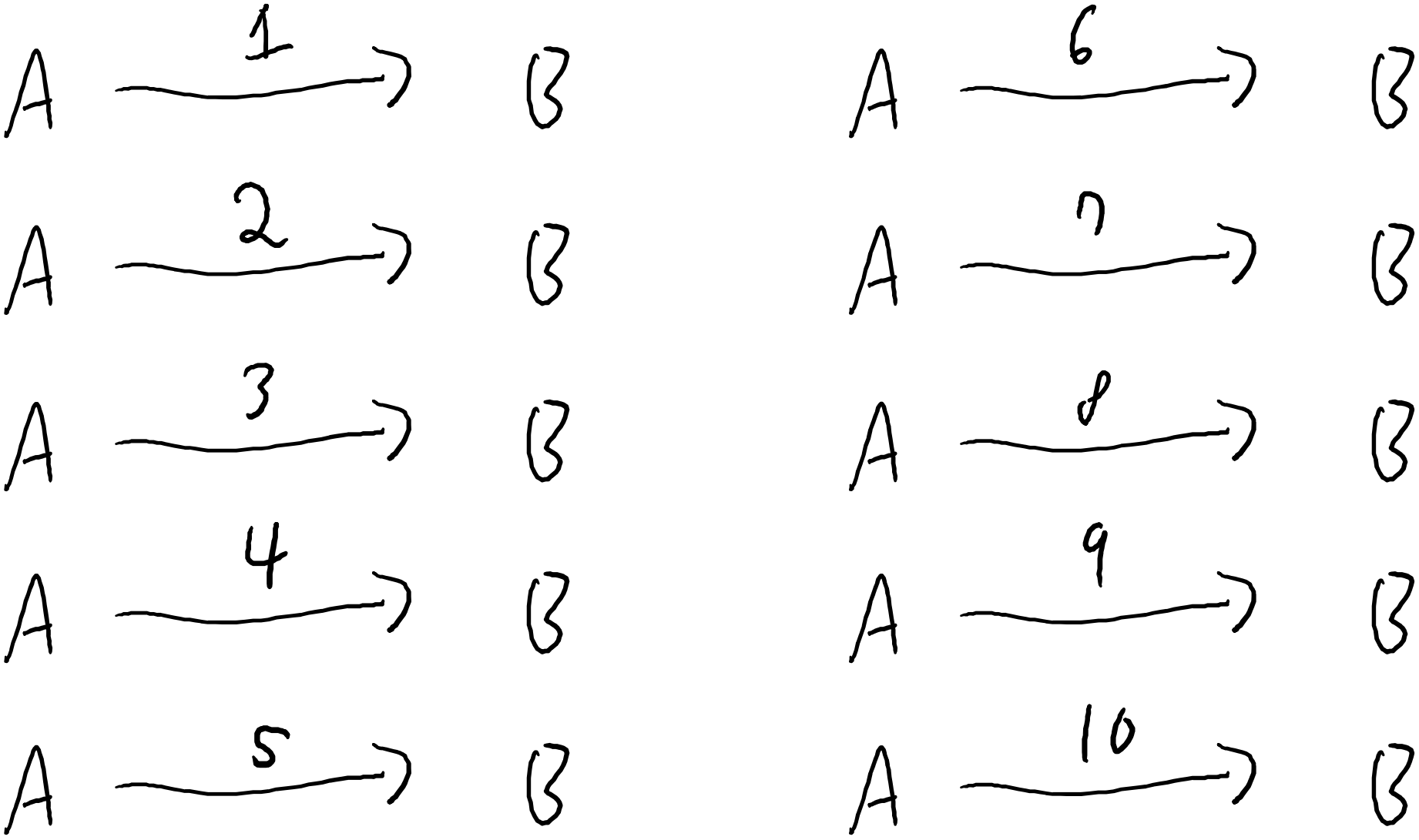

TLC

@hillelogram


TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
Pause
00:00:01
Select Area
Audio
Record Pointer
Transfer.tla
Safety
Model Overview
Advanced Options
Model Checking Results
Model Overview
Model description
What is the behavior spec?
Initial predicate and next-state relation
Init:
Next:
Temporal formula
Spec
No Behavior Spec
What to check?
Deadlock
Invariants
Formulas true in every reachable state.
NoOverdrafts
Add
Edit
Properties
Temporal formulas true for every possible behavior.
Termination
Add
Edit
What is the model?
Specify the values of declared constants.
Edit
Advanced parts of the model:
Additional definitions,
Definition override,
State constraints,
Action constraints,
Additional model values,
How to run?
142M of 275M
Spec Status :
parsed

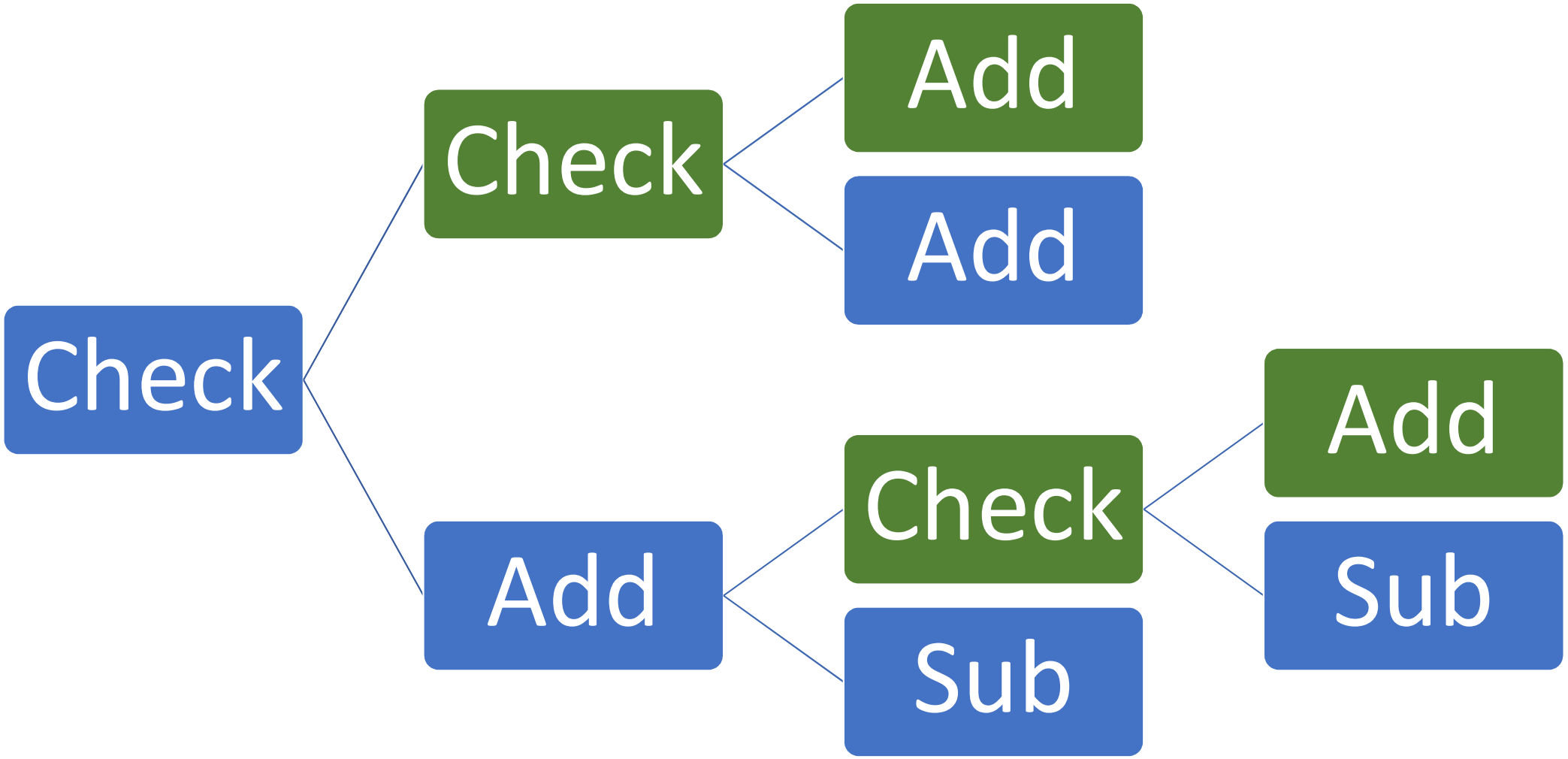
```
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

process transfer = 1
  variables amount = 1,
            sender = "alice",
            receiver = "bob";
begin
  Check: if account[sender] >= amount then
    Add: account[receiver] := account[receiver] + amount;
    Sub: account[sender] := account[sender] - amount;
  end if;
end process;

NoOverdrafts == \A p \in people: account[p] >= 0
```

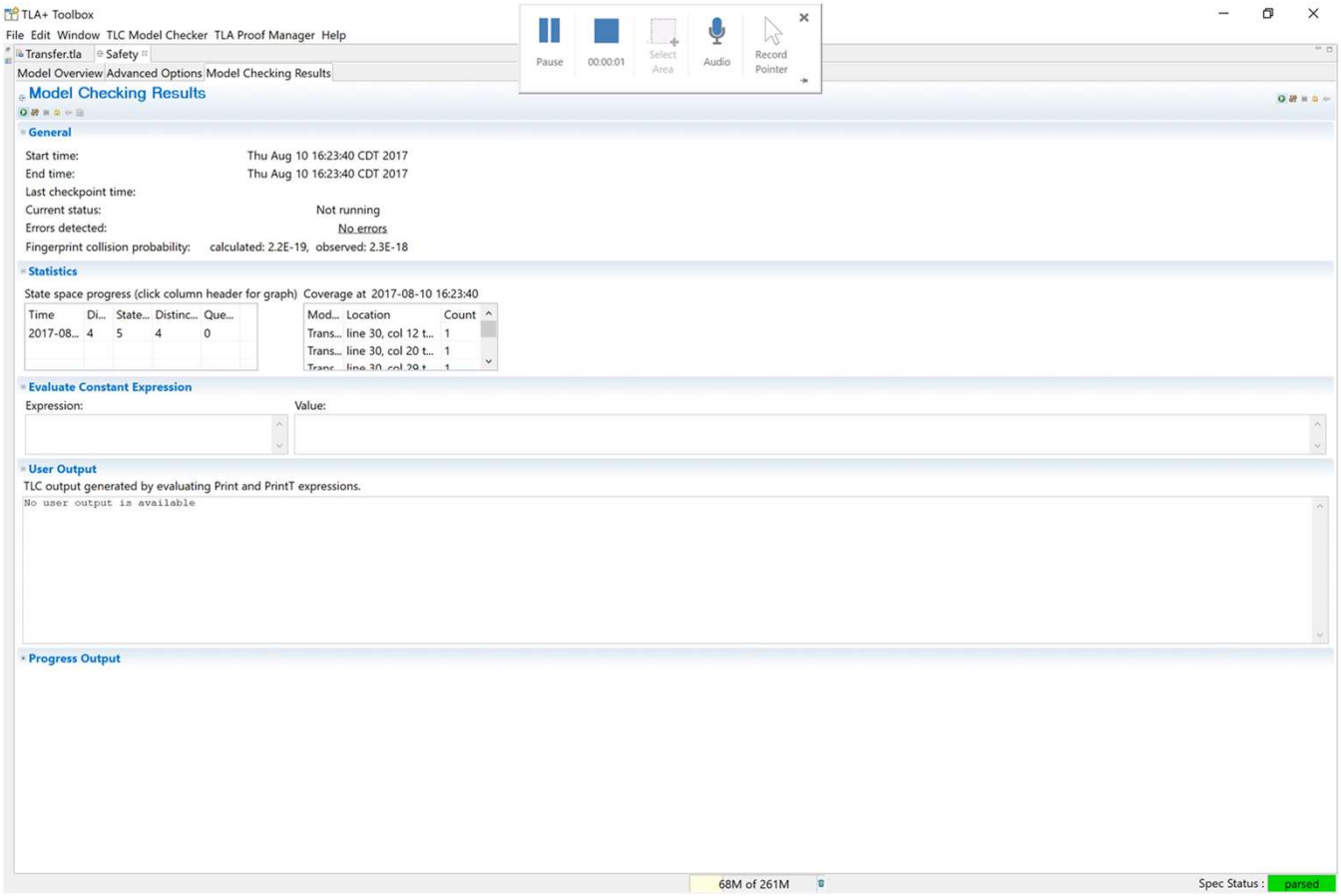
```
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

process transfer = 1
  variables amount \in 1..10,
            sender = "alice",
            receiver = "bob";
begin
  Check: if account[sender] >= amount then
    Add: account[receiver] := account[receiver] + amount;
    Sub: account[sender] := account[sender] - amount;
  end if;
end process;

NoOverdrafts == \A p \in people: account[p] >= 0
```

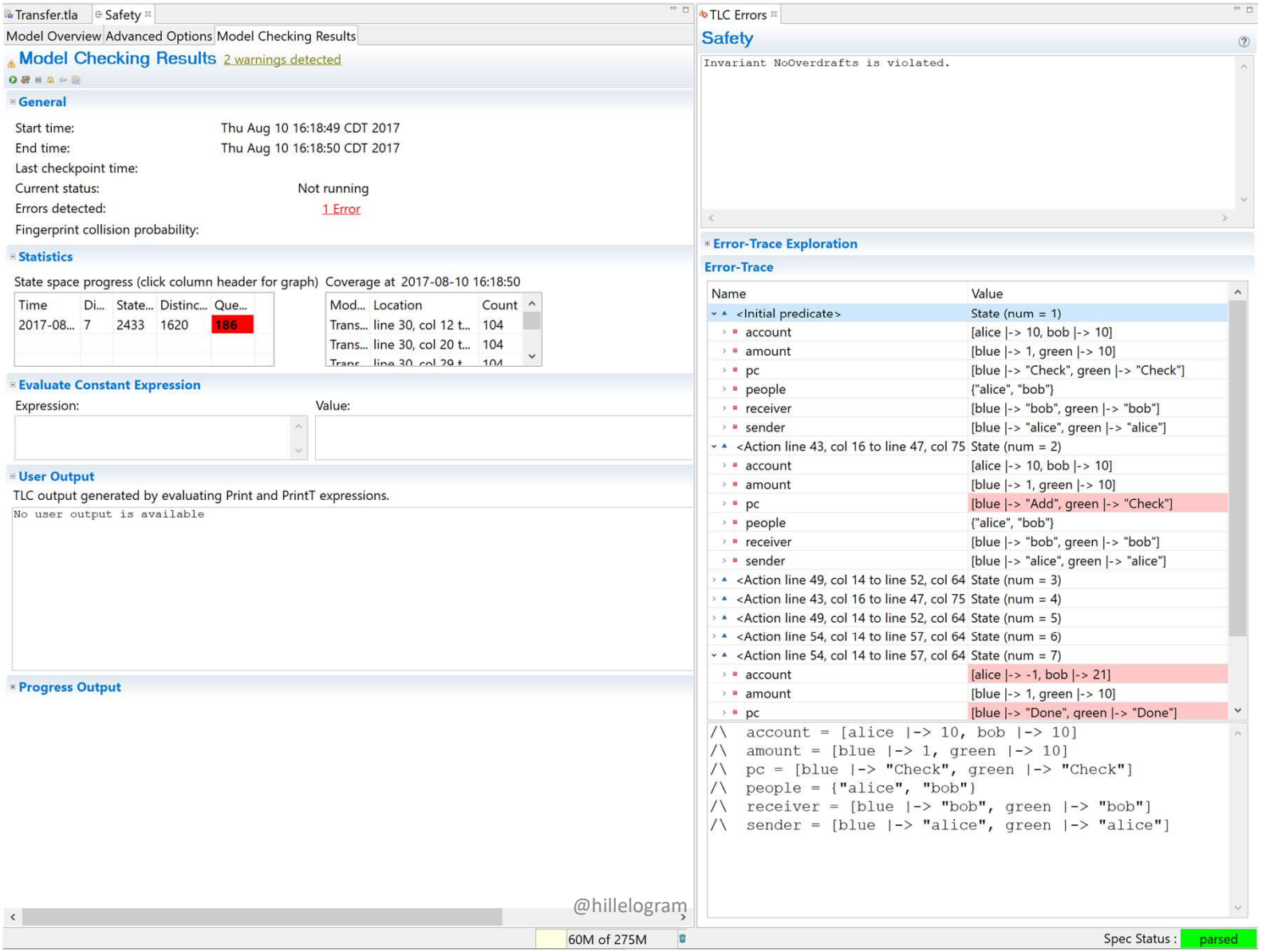


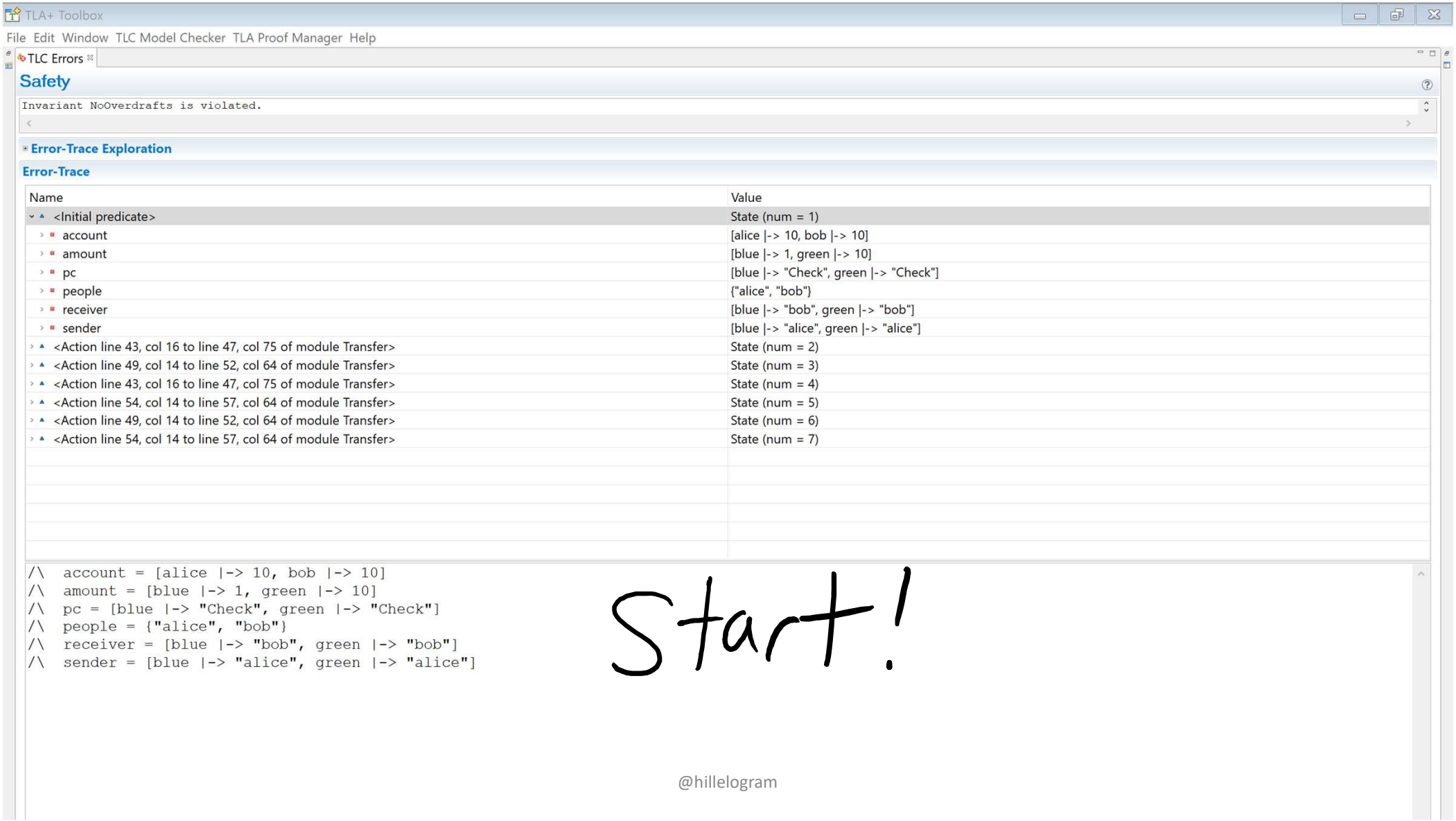
```
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

process transfer = 1
  variables amount \in 1..10,
            sender = "alice",
            receiver = "bob";
begin
  Check: if account[sender] >= amount then
    Add: account[receiver] := account[receiver] + amount;
    Sub: account[sender] := account[sender] - amount;
  end if;
end process;

NoOverdrafts == \A p \in people: account[p] >= 0
```

```
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

process transfer \in {"blue", "green"}
  variables amount \in 1..10,
            sender = "alice",
            receiver = "bob";
begin
  Check: if account[sender] >= amount then
    Add: account[receiver] := account[receiver] + amount;
    Sub: account[sender] := account[sender] - amount;
  end if;
end process;

NoOverdrafts == \A p \in people: account[p] >= 0
```

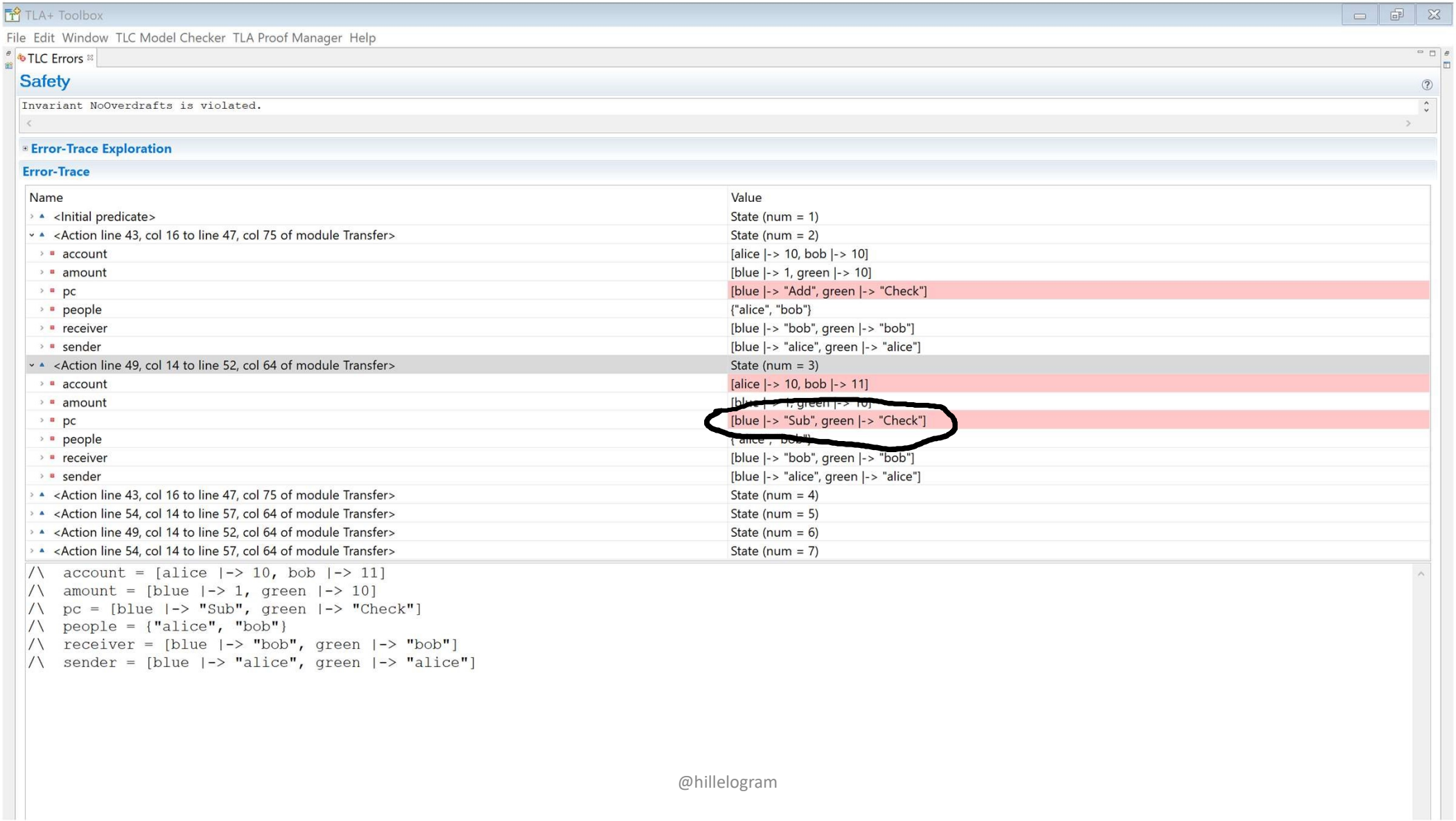

Check
Check
Add
Add
Add
Check
Add
Sub
Sub

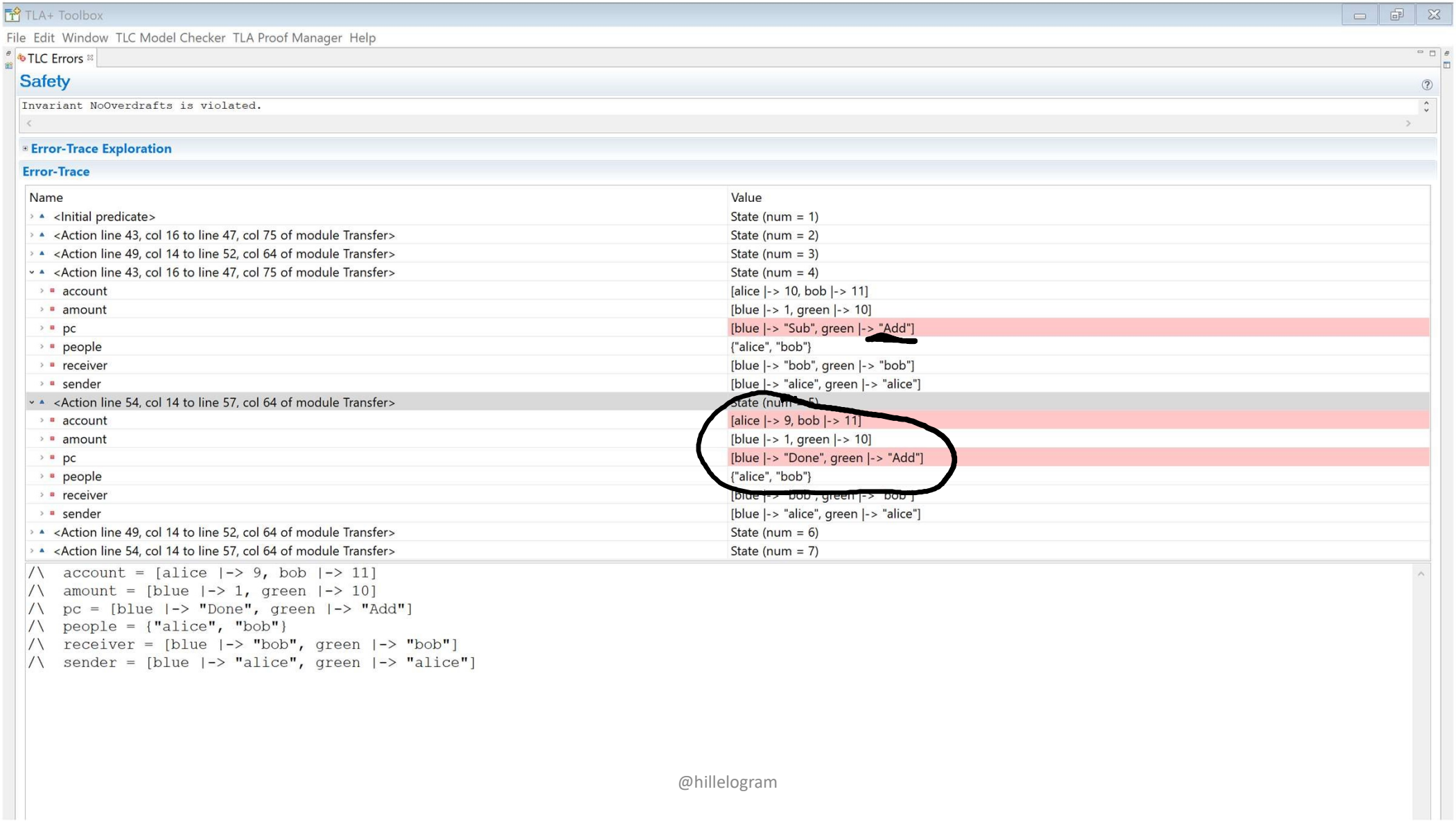

TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
Pause
00:00:01
Select Area
Audio
Record Pointer
Transfer.tla
Safety
Model Overview
Advanced Options
Model Checking Results
Model Checking Results
General
Start time: Thu Aug 10 16:23:40 CDT 2017
End time: Thu Aug 10 16:23:40 CDT 2017
Last checkpoint time:
Current status: Not running
Errors detected: No errors
Fingerprint collision probability: calculated: 2.2E-19, observed: 2.3E-18
Statistics
State space progress (click column header for graph)
Time Di... State... Distinc... Que...
2017-08... 4 5 4 0
Coverage at 2017-08-10 16:23:40
Mod... Location Count
Trans... line 30, col 12 t... 1
Trans... line 30, col 20 t... 1
Evaluate Constant Expression
Expression:
Value:
User Output
TLC output generated by evaluating Print and PrintT expressions.
No user output is available
Progress Output
68M of 261M
Spec Status : parsed

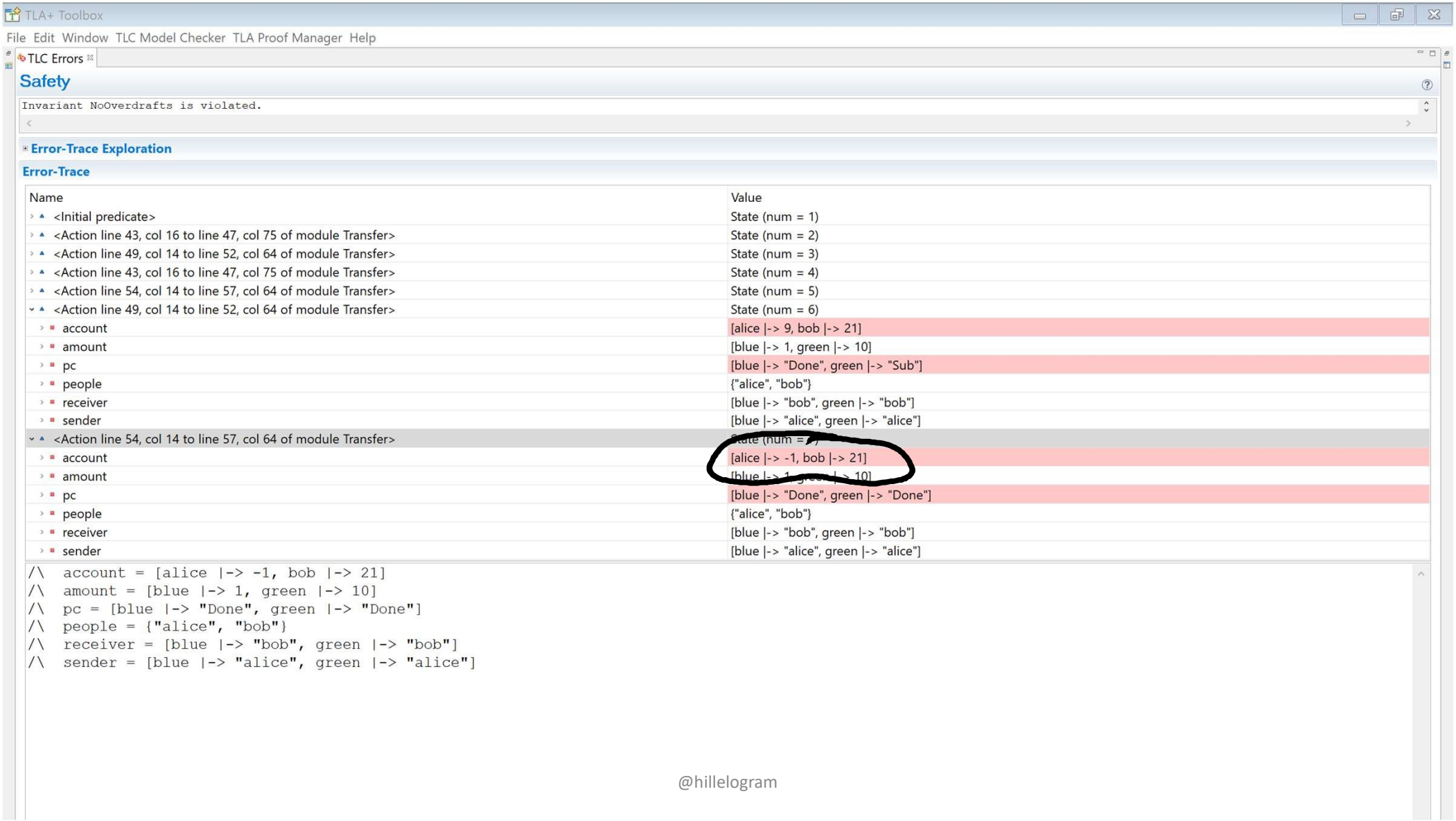

Transfer.tla
Safety
Model Overview
Advanced Options
Model Checking Results
Model Checking Results
2 warnings detected
General
Start time: Thu Aug 10 16:18:49 CDT 2017
End time: Thu Aug 10 16:18:50 CDT 2017
Last checkpoint time:
Current status: Not running
Errors detected: 1 Error
Fingerprint collision probability:
Statistics
State space progress (click column header for graph)
Time Di... State... Distinc... Que...
2017-08... 7 2433 1620 186
Coverage at 2017-08-10 16:18:50
Mod... Location Count
Trans... line 30, col 12 t... 104
Trans... line 30, col 20 t... 104
Evaluate Constant Expression
Expression:
Value:
User Output
TLC output generated by evaluating Print and PrintT expressions.
No user output is available
Progress Output
@hillelogram
60M of 275M
TLC Errors
Safety
Invariant NoOverdrafts is violated.
Error-Trace Exploration
Error-Trace
Name
Value
<Initial predicate>
State (num = 1)
account
[alice |-> 10, bob |-> 10]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Check", green |-> "Check"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 43, col 16 to line 47, col 75
State (num = 2)
account
[alice |-> 10, bob |-> 10]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Add", green |-> "Check"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 49, col 14 to line 52, col 64
State (num = 3)
<Action line 43, col 16 to line 47, col 75
State (num = 4)
<Action line 49, col 14 to line 52, col 64
State (num = 5)
<Action line 54, col 14 to line 57, col 64
State (num = 6)
<Action line 54, col 14 to line 57, col 64
State (num = 7)
account
[alice |-> -1, bob |-> 21]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Done", green |-> "Done"]
/\ account = [alice |-> 10, bob |-> 10]
/\ amount = [blue |-> 1, green |-> 10]
/\ pc = [blue |-> "Check", green |-> "Check"]
/\ people = {"alice", "bob"}
/\ receiver = [blue |-> "bob", green |-> "bob"]
/\ sender = [blue |-> "alice", green |-> "alice"]
Spec Status : parsed


TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
TLC Errors
Safety
Invariant NoOverdrafts is violated.
Error-Trace Exploration
Error-Trace
Name
Value
<Initial predicate>
State (num = 1)
account
[alice |-> 10, bob |-> 10]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Check", green |-> "Check"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 2)
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 3)
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 4)
<Action line 54, col 14 to line 57, col 64 of module Transfer>
State (num = 5)
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 6)
<Action line 54, col 14 to line 57, col 64 of module Transfer>
State (num = 7)
/\ account = [alice |-> 10, bob |-> 10]
/\ amount = [blue |-> 1, green |-> 10]
/\ pc = [blue |-> "Check", green |-> "Check"]
/\ people = {"alice", "bob"}
/\ receiver = [blue |-> "bob", green |-> "bob"]
/\ sender = [blue |-> "alice", green |-> "alice"]

Start!

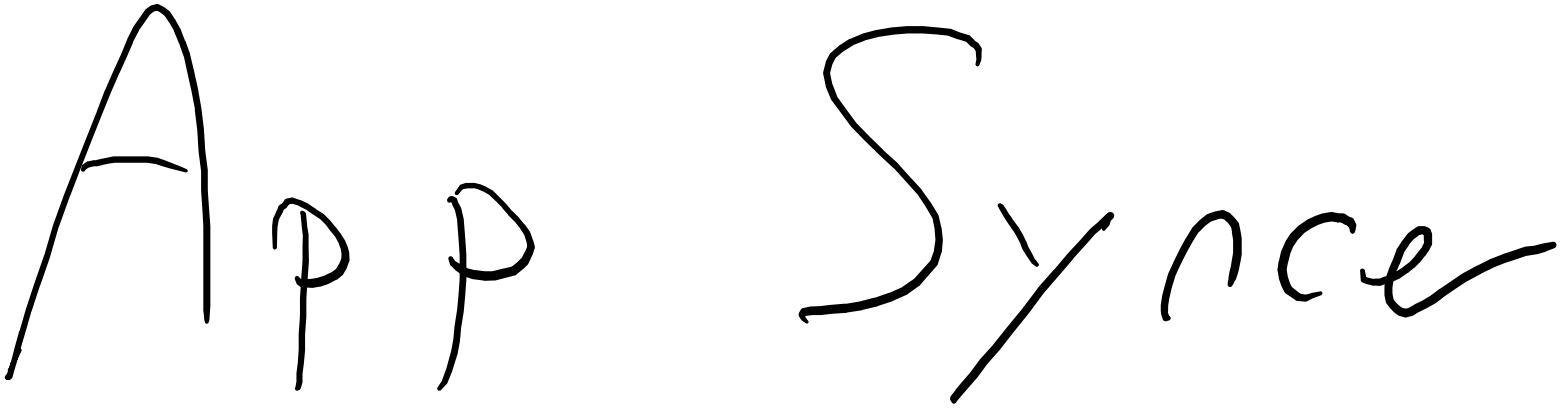

TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
TLC Errors
Safety
Invariant NoOverdrafts is violated.
Error-Trace Exploration
Error-Trace
Name
Value
<Initial predicate>
State (num = 1)
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 2)
account
[alice |-> 10, bob |-> 10]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Add", green |-> "Check"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 3)
account
[alice |-> 10, bob |-> 11]
amount
pc
[blue |-> "Sub", green |-> "Check"]
people
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 4)
<Action line 54, col 14 to line 57, col 64 of module Transfer>
State (num = 5)
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 6)
<Action line 54, col 14 to line 57, col 64 of module Transfer>
State (num = 7)
/\ account = [alice |-> 10, bob |-> 11]
/\ amount = [blue |-> 1, green |-> 10]
/\ pc = [blue |-> "Sub", green |-> "Check"]
/\ people = {"alice", "bob"}
/\ receiver = [blue |-> "bob", green |-> "bob"]
/\ sender = [blue |-> "alice", green |-> "alice"]

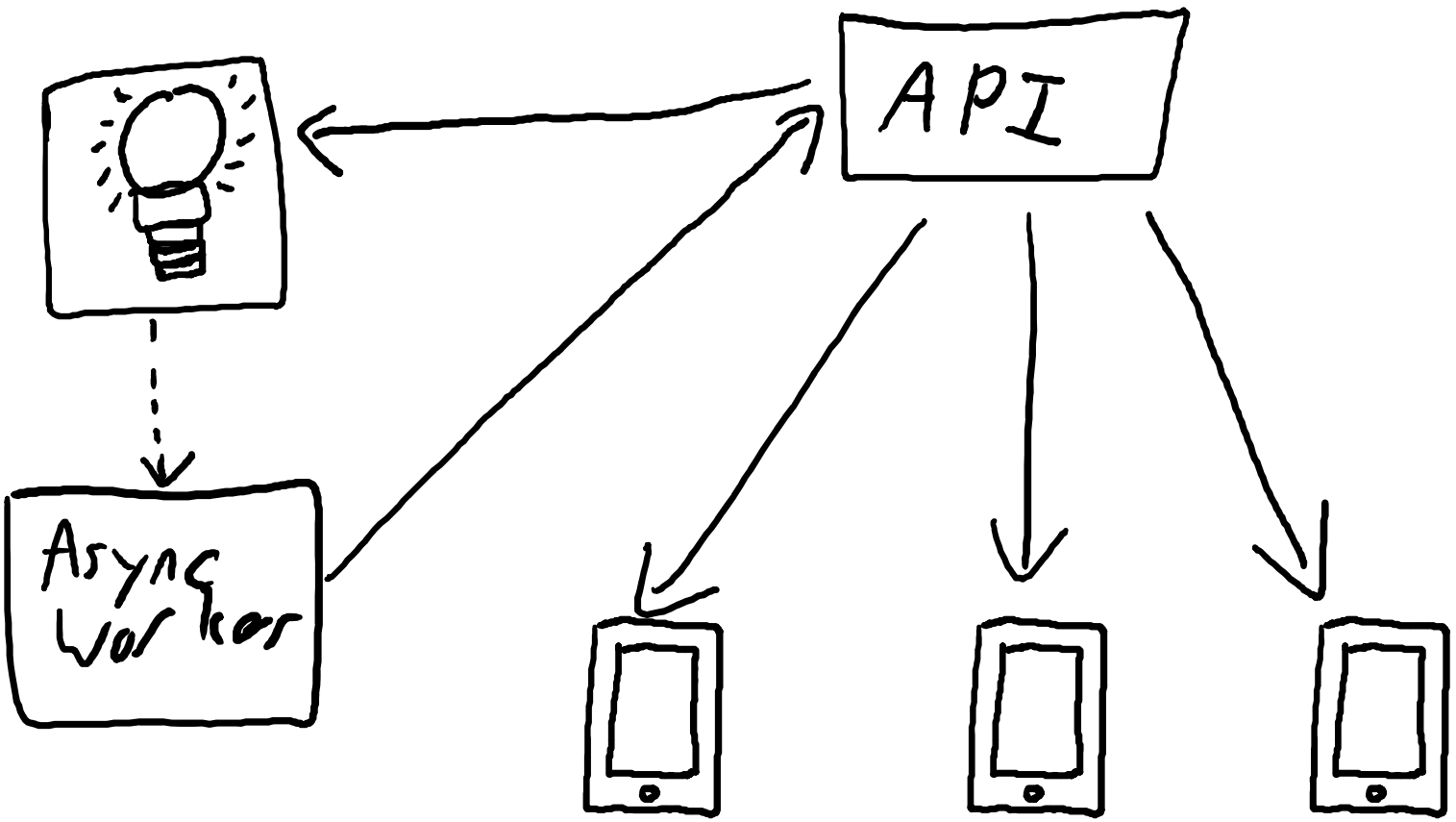

TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
TLC Errors
Safety
Invariant NoOverdrafts is violated.
Error-Trace Exploration
Error-Trace
Name
Value
<Initial predicate>
State (num = 1)
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 2)
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 3)
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 4)
account
[alice |-> 10, bob |-> 11]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Sub", green |-> "Add"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 54, col 14 to line 57, col 64 of module Transfer>
account
[alice |-> 9, bob |-> 11]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Done", green |-> "Add"]
people
{"alice", "bob"}
sender
[blue |-> "alice", green |-> "alice"]
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 6)
<Action line 54, col 14 to line 57, col 64 of module Transfer>
State (num = 7)
/\ account = [alice |-> 9, bob |-> 11]
/\ amount = [blue |-> 1, green |-> 10]
/\ pc = [blue |-> "Done", green |-> "Add"]
/\ people = {"alice", "bob"}
/\ receiver = [blue |-> "bob", green |-> "bob"]
/\ sender = [blue |-> "alice", green |-> "alice"]


TLA+ Toolbox
File Edit Window TLC Model Checker TLA Proof Manager Help
TLC Errors
Safety
Invariant NoOverdrafts is violated.
Error-Trace Exploration
Error-Trace
Name
Value
<Initial predicate>
State (num = 1)
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 2)
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 3)
<Action line 43, col 16 to line 47, col 75 of module Transfer>
State (num = 4)
<Action line 54, col 14 to line 57, col 64 of module Transfer>
State (num = 5)
<Action line 49, col 14 to line 52, col 64 of module Transfer>
State (num = 6)
account
[alice |-> 9, bob |-> 21]
amount
[blue |-> 1, green |-> 10]
pc
[blue |-> "Done", green |-> "Sub"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
<Action line 54, col 14 to line 57, col 64 of module Transfer>
account
[alice |-> -1, bob |-> 21]
amount
pc
[blue |-> "Done", green |-> "Done"]
people
{"alice", "bob"}
receiver
[blue |-> "bob", green |-> "bob"]
sender
[blue |-> "alice", green |-> "alice"]
/\ account = [alice |-> -1, bob |-> 21]
/\ amount = [blue |-> 1, green |-> 10]
/\ pc = [blue |-> "Done", green |-> "Done"]
/\ people = {"alice", "bob"}
/\ receiver = [blue |-> "bob", green |-> "bob"]
/\ sender = [blue |-> "alice", green |-> "alice"]

```
variables people = {"alice", "bob"},
          account = [p \in people |-> 10]

process transfer \in {"blue", "green"}
  variables amount \in 1..10,
            sender = "alice",
            receiver = "bob";
begin
  Transfer:
    if account[sender] >= amount then
      account[receiver] := account[receiver] + amount ||
      account[sender] := account[sender] - amount;
    end if;
end process;

NoOverdrafts == \A p \in people: account[p] >= 0
```

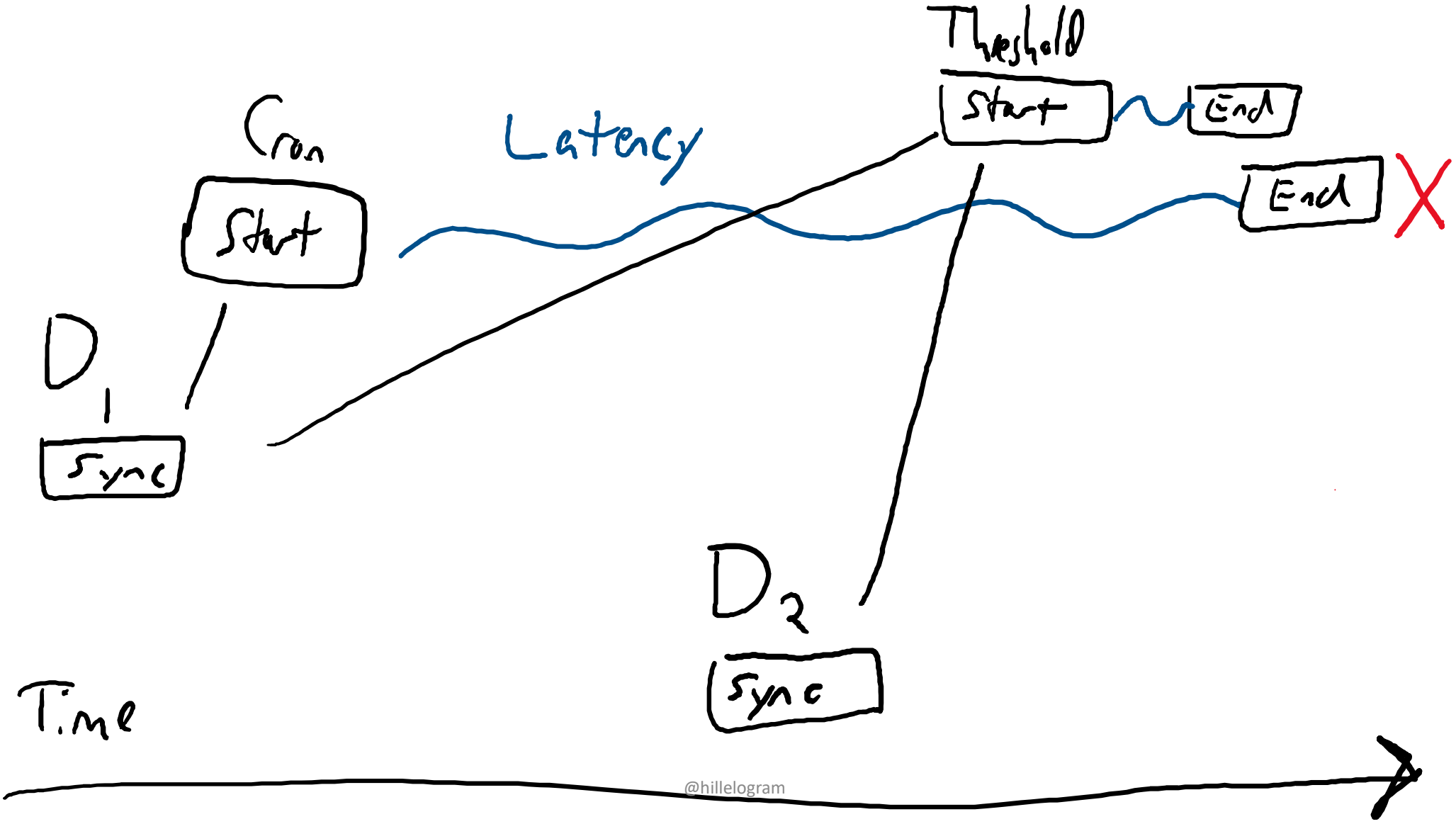
Who Cares?


App Sync

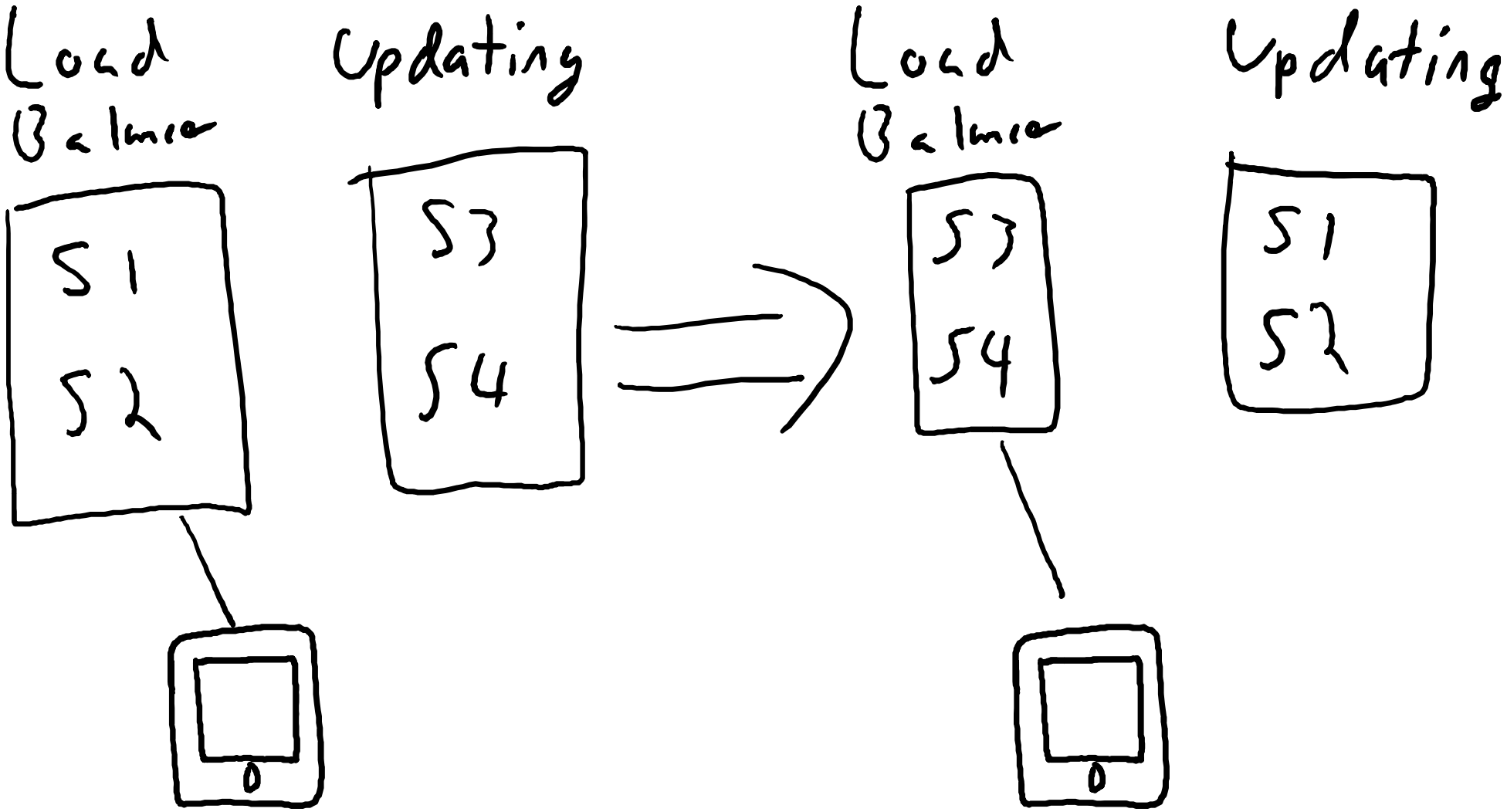

API
Async
Worker

Never try to install an app that is already installed

```
\A d \in Devices:
  InstallCount[d] \in {0, 1}
```


Threshold
Start
End
Cron
Latency
Start
End
X
D1
Sync
D2
Sync
Time
@hillelogram

Zero Downtime Deployments

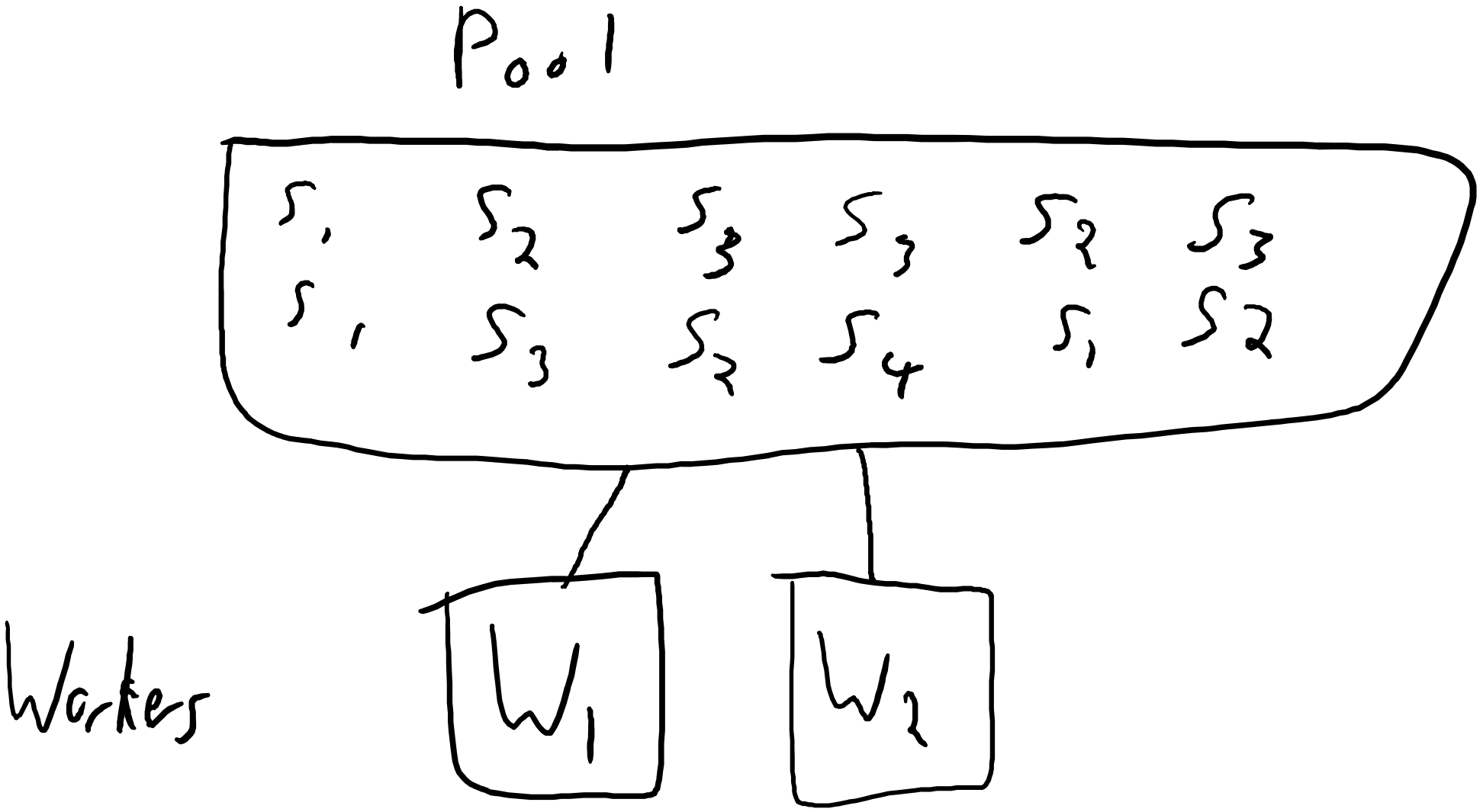

Load Balancer
S1
S2
Updating
S3
S4
Load Balancer
S3
S4
Updating
S1
S2

At least one server is available, and all available servers on same code

```
\E server \in load_balancer:
  /\ status[server] /= UPDATING
  /\ \A s \in load_balancer:
       version[server] = version[s]
```

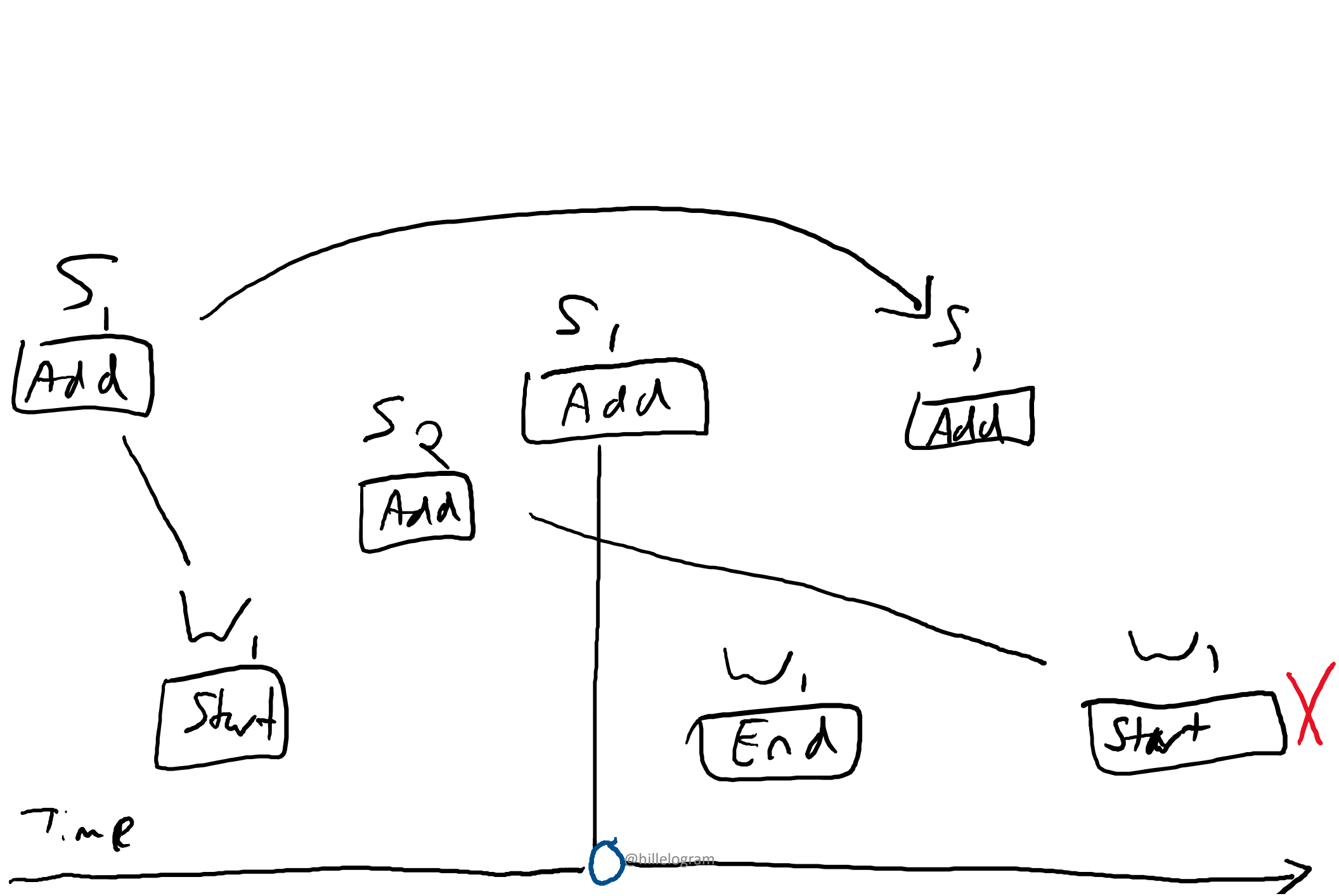
(actually worked pretty well)

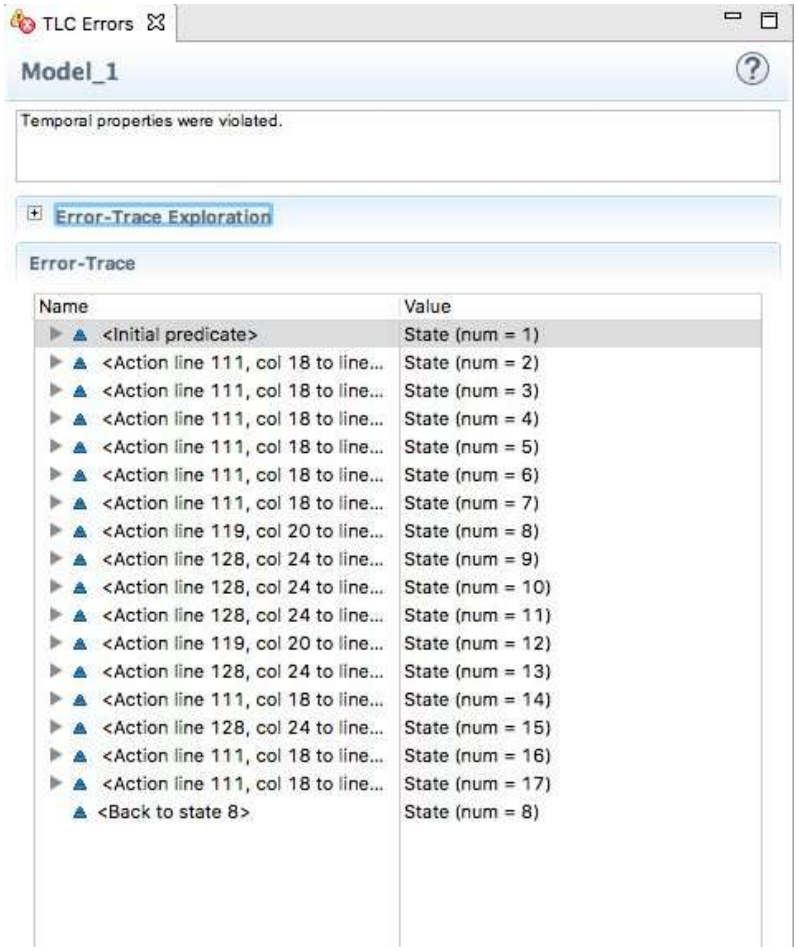
"Redacted"


Pool
S1 S2 S3 S3 S2 S3
S1 S3 S2 S4 S1 S2
Workers
W1
W2

When worker processes message source, does not switch sources until all messages processed

```
[][\A w \in Workers:
    LET s == on_source[w] IN
    messages_for[s] /= {}
      => UNCHANGED s]_vars
```


S1
Add
S1
Add
S1
Add
S2
Add
W1
Start
W1
End
W1
Start
Time
0
@hillelogram

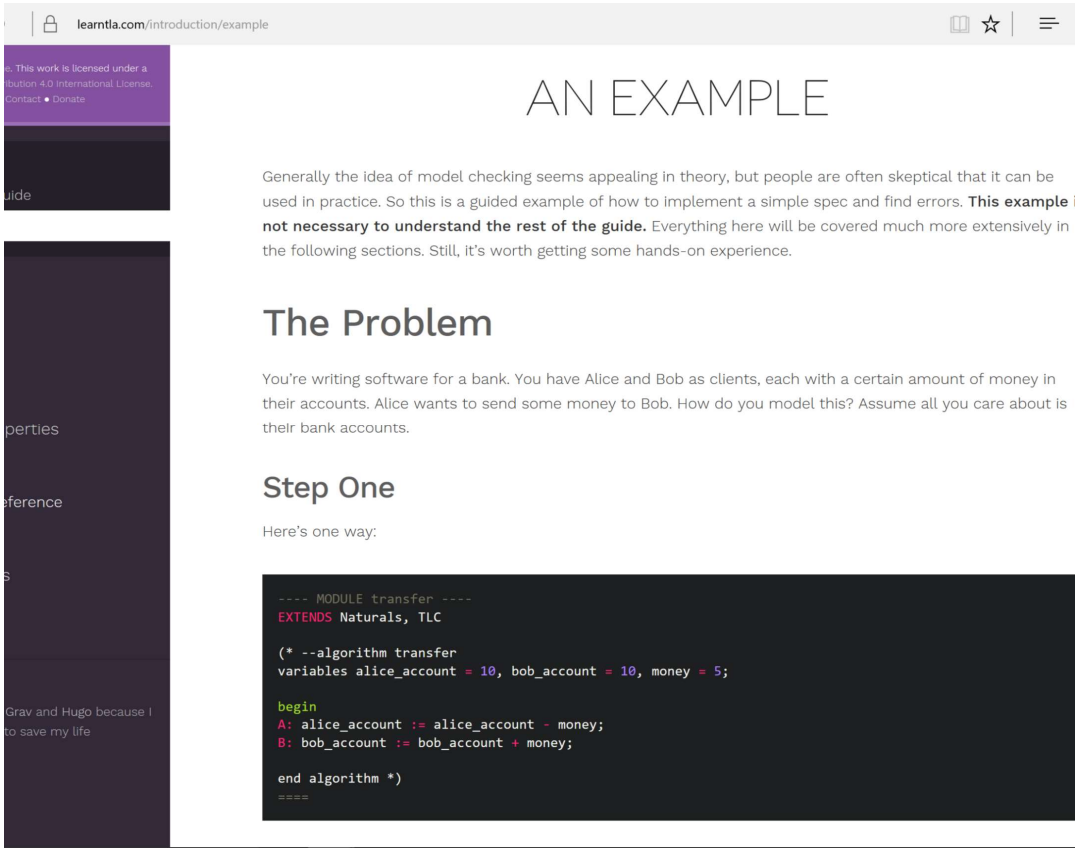

TLC Errors
Model_1
Temporal properties were violated.
Error-Trace Exploration
Error-Trace
Name
Value
<Initial predicate>
State (num = 1)
<Action line 111, col 18 to line...
State (num = 2)
<Action line 111, col 18 to line...
State (num = 3)
<Action line 111, col 18 to line...
State (num = 4)
<Action line 111, col 18 to line...
State (num = 5)
<Action line 111, col 18 to line...
State (num = 6)
<Action line 111, col 18 to line...
State (num = 7)
<Action line 119, col 20 to line...
State (num = 8)
<Action line 128, col 24 to line...
State (num = 9)
<Action line 128, col 24 to line...
State (num = 10)
<Action line 128, col 24 to line...
State (num = 11)
<Action line 119, col 20 to line...
State (num = 12)
<Action line 128, col 24 to line...
State (num = 13)
<Action line 111, col 18 to line...
State (num = 14)
<Action line 128, col 24 to line...
State (num = 15)
<Action line 111, col 18 to line...
State (num = 16)
<Action line 111, col 18 to line...
State (num = 17)
<Back to state 8>
State (num = 8)

Conclusions

- Concurrency is hard
- Specification is good
- TLA+ is cool
- Lamport is scary

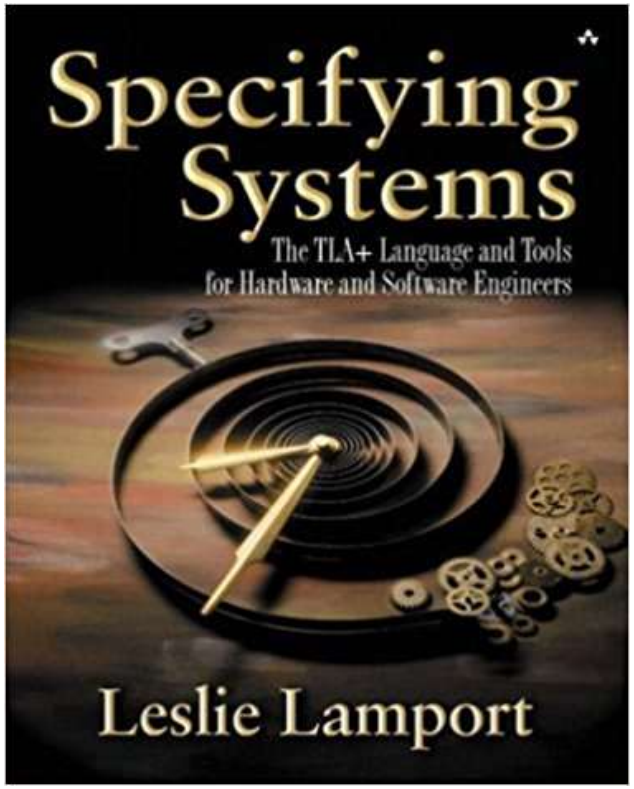
www.learntla.com

- Me
- Beginner-level
- Uses PlusCal
- Lots of examples and exercises
- Easiest 30%



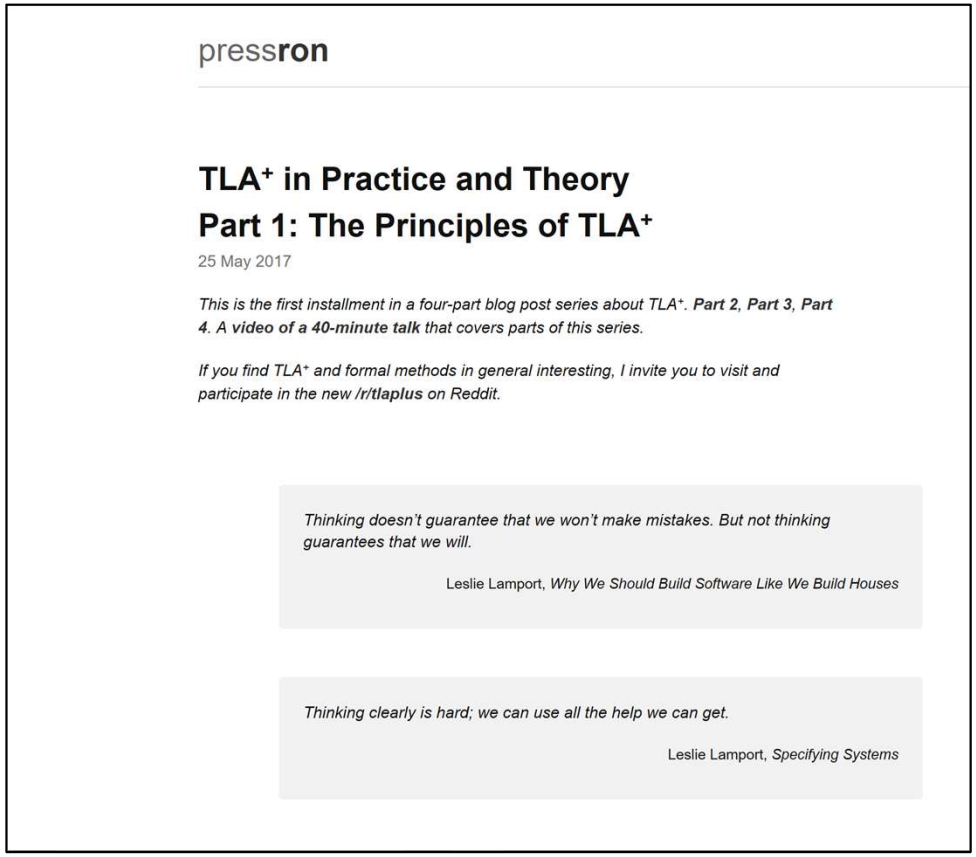
Specifying Systems

- Leslie Lamport
- Canonical text
- Covers theory
- No PlusCal
- Intermediate-level



TLA+ in Practice and Theory (pron.github.io)

- Ron Pressler
- Covers history, theory, math
- More advanced, theoretical
- Wallace and Gromit avatar



Thanks for coming!

@hillelogram

hillelwayne.com

learntla.com