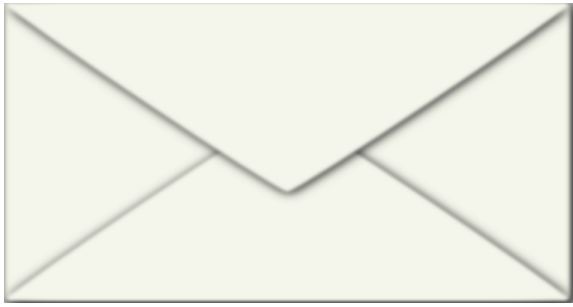
Automated Troubleshooting for SDN Control Software

Colin Scott, Andreas Wundsam, Andrew Or, Sam Whitlock, Eugene Huang, Kyriakos Zarifis, Scott Shenker

What inputs do SDN controllers see?



- 20,000 servers * 4 VMs / server = 80,000 VMs
- (6 migrations / day / VM) + (2 power up | down / day / VM) * 80,000 VMs = 640,000 VM events / day ~= **450 VM changes / minute** [1]
- **8.5 network error events / minute** [2]
- 1 policy change / tenant / day * 2000 tenants ~= **1 policy change / minute**

What inputs do SDN controllers see?

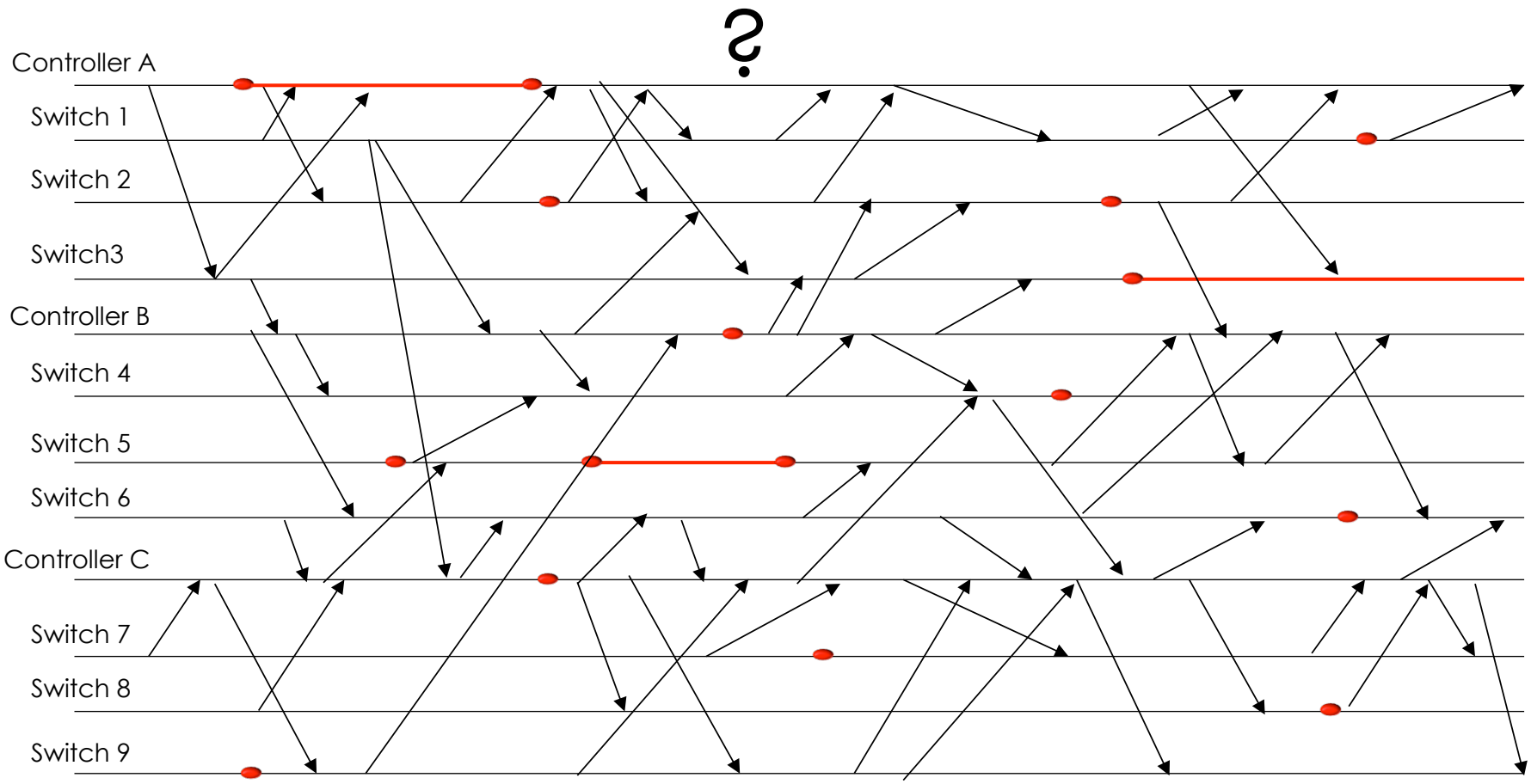
= ~500 inputs / minute

Something goes wrong!

Best practice: Logs

Manual analysis of log files

Best practice: Logs



Our First Goal

Make it easy for developers to find bugs

Our Second Goal

Identify a **minimal** sequence of inputs that triggers the bug

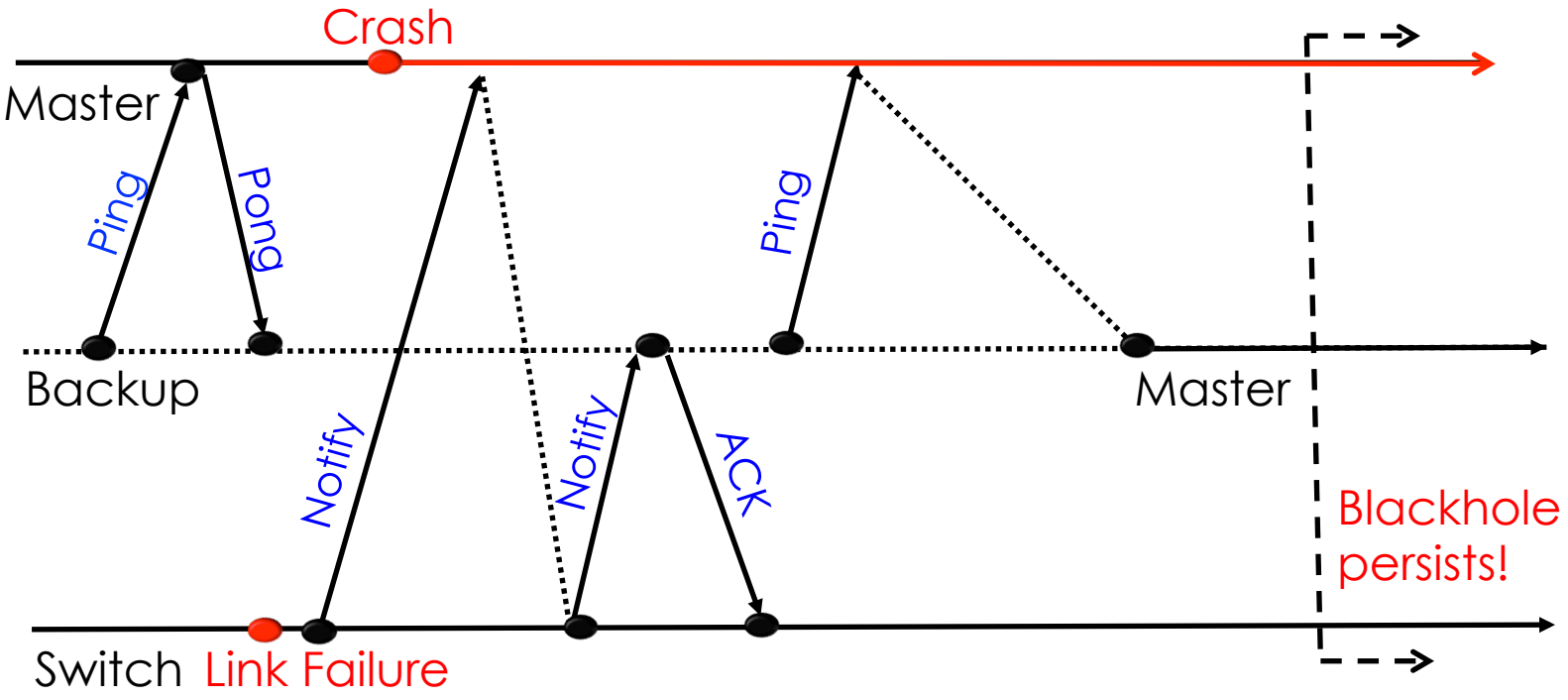
Minimal Causal Sequence

$$MCS \subset Log \; s.t.$$

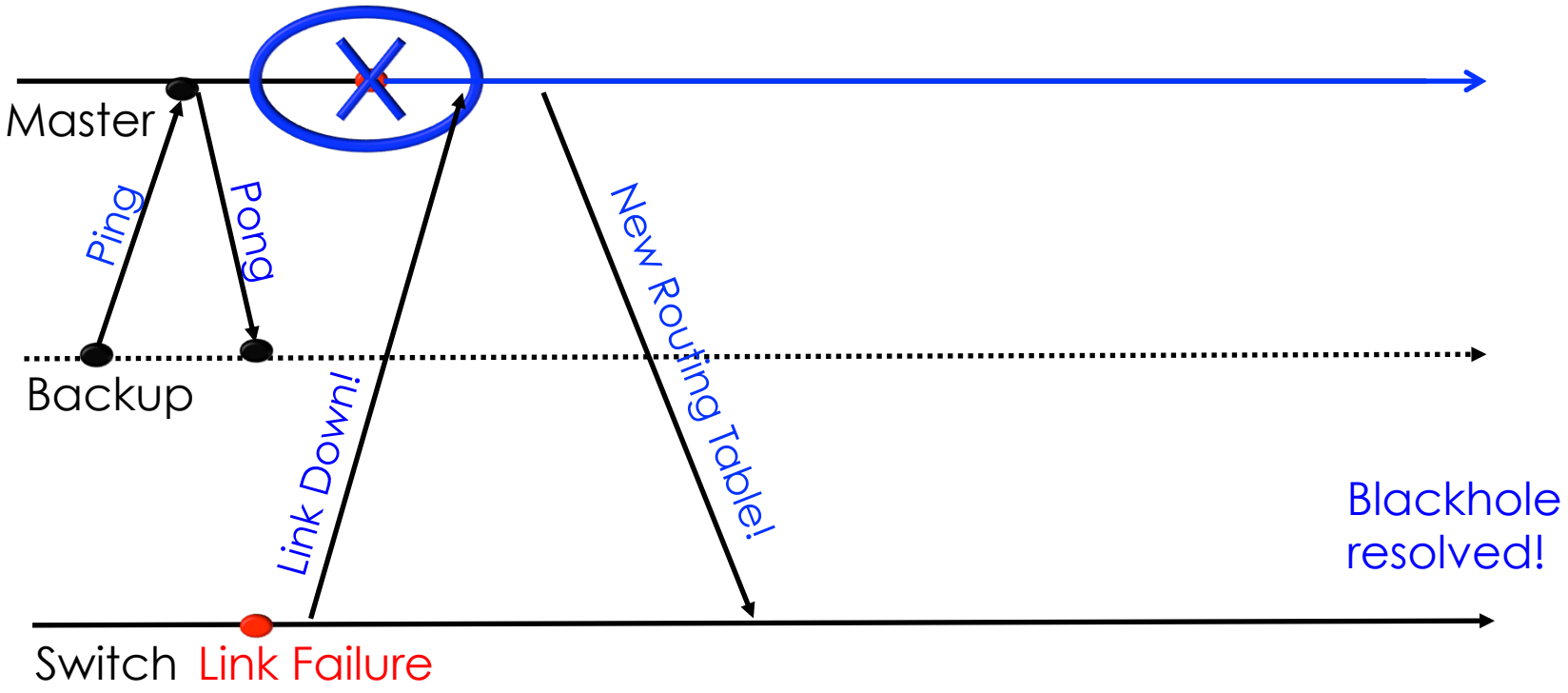
$$i.\; replay(MCS) = \chi$$

$$ii.\; \forall_{e \in MCS} replay(MCS - \{e\}) \neq \chi$$

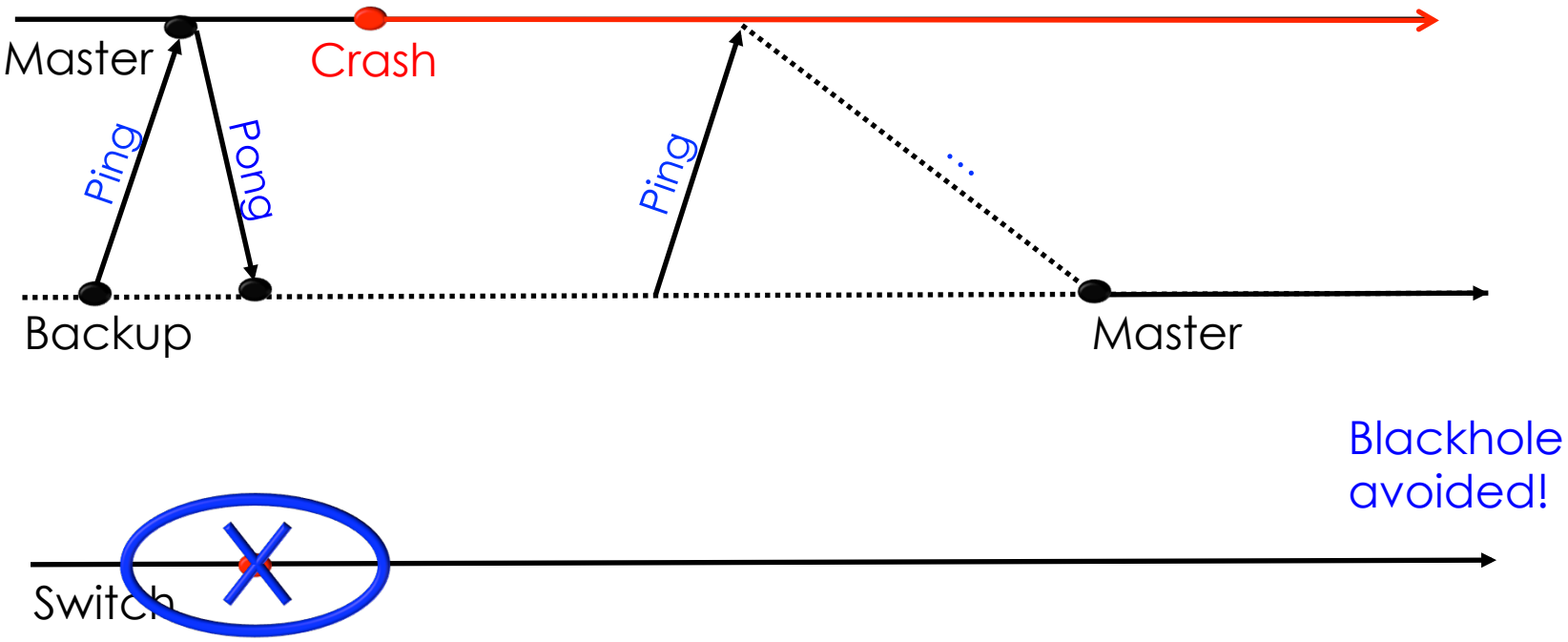
Minimal Causal Sequence



Minimal Causal Sequence

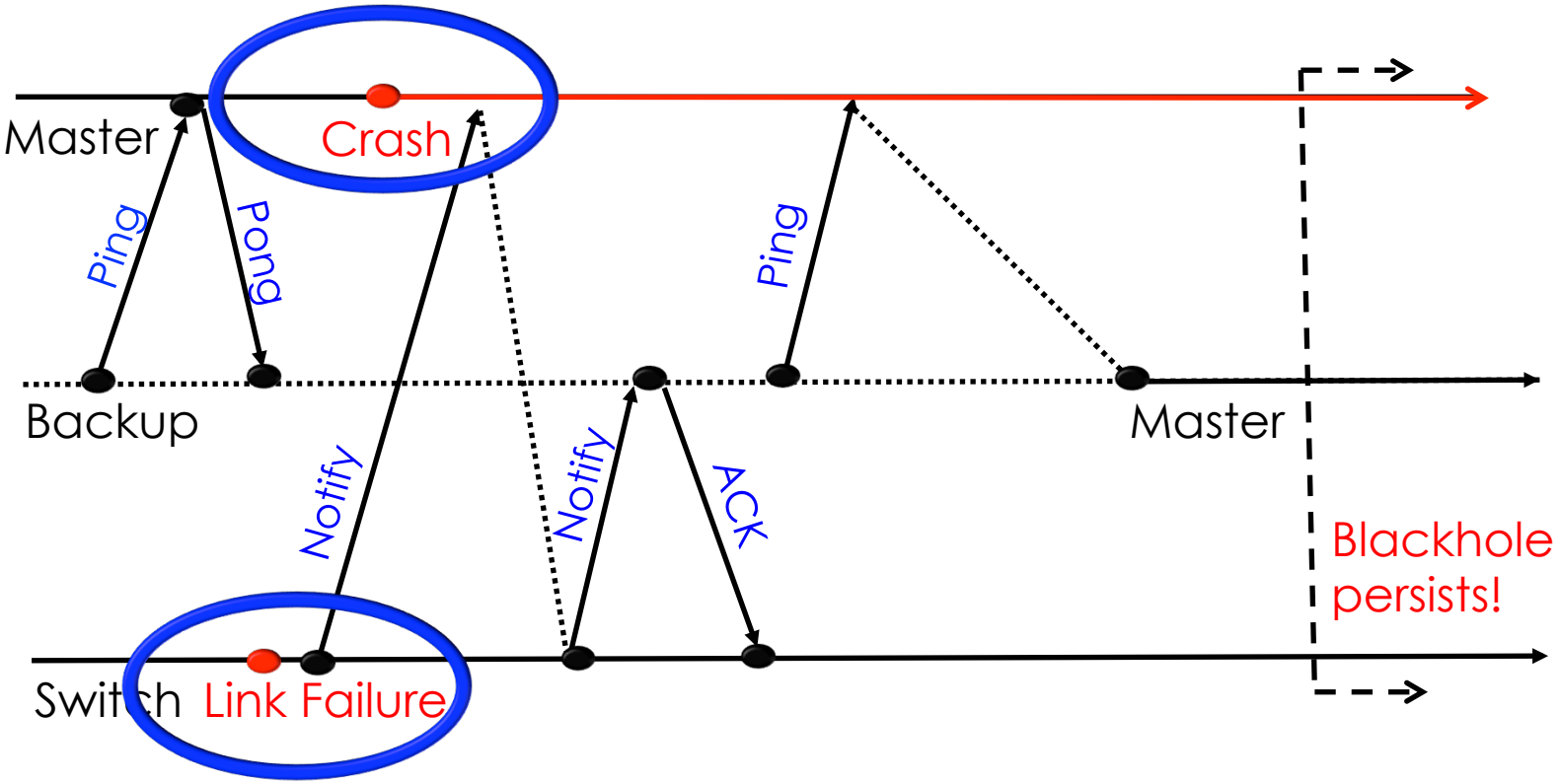


Minimal Causal Sequence

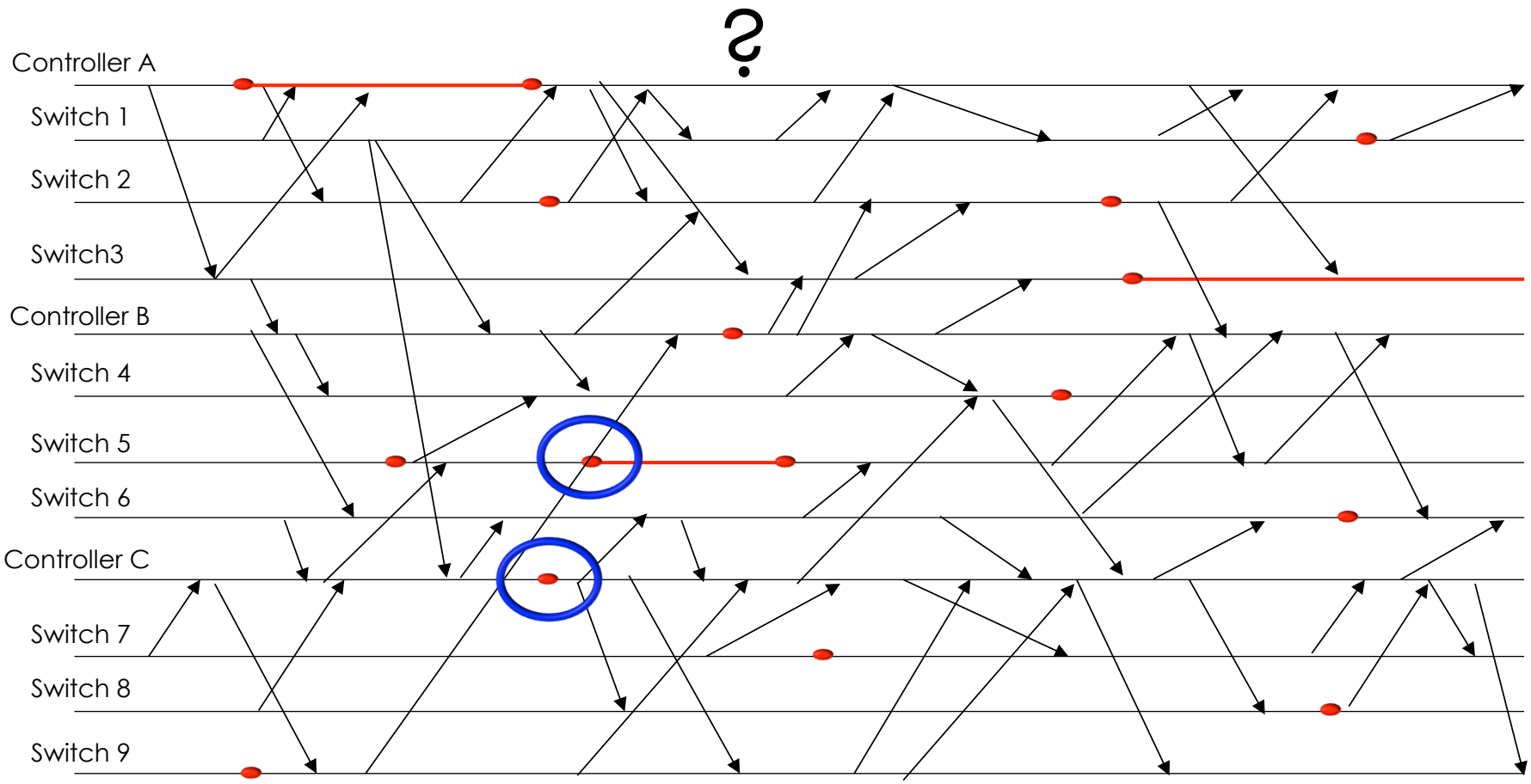


Minimal Causal Sequence

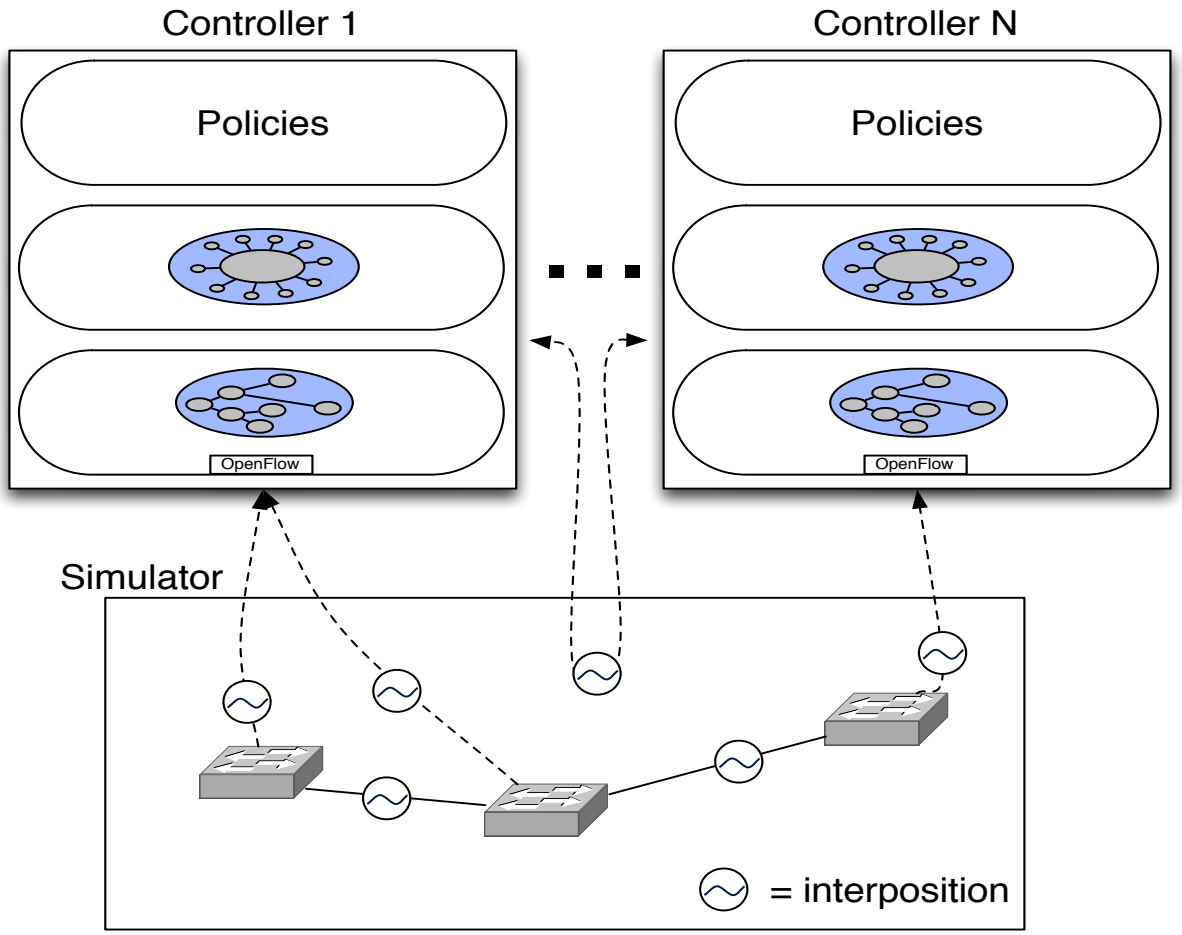
Both are necessary conditions!



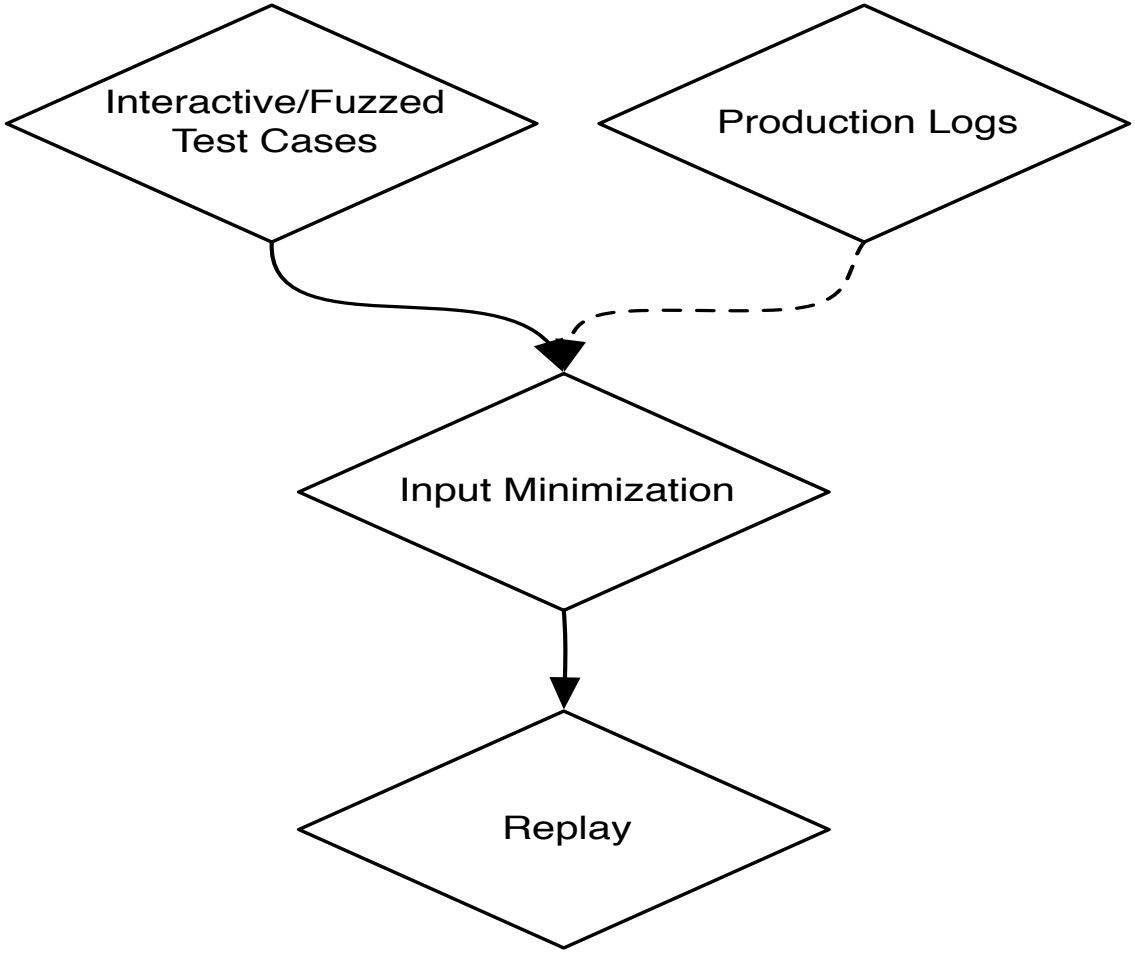
Minimal Causal Sequence



Approach: Simulated Execution



Workflow


Interactive/Fuzzed
Test Cases
Production Logs
Input Minimization
Replay

Approach: Delta Debugging

Modify history!

Possible failure causes

Set up first hypothesis

Test first hypothesis

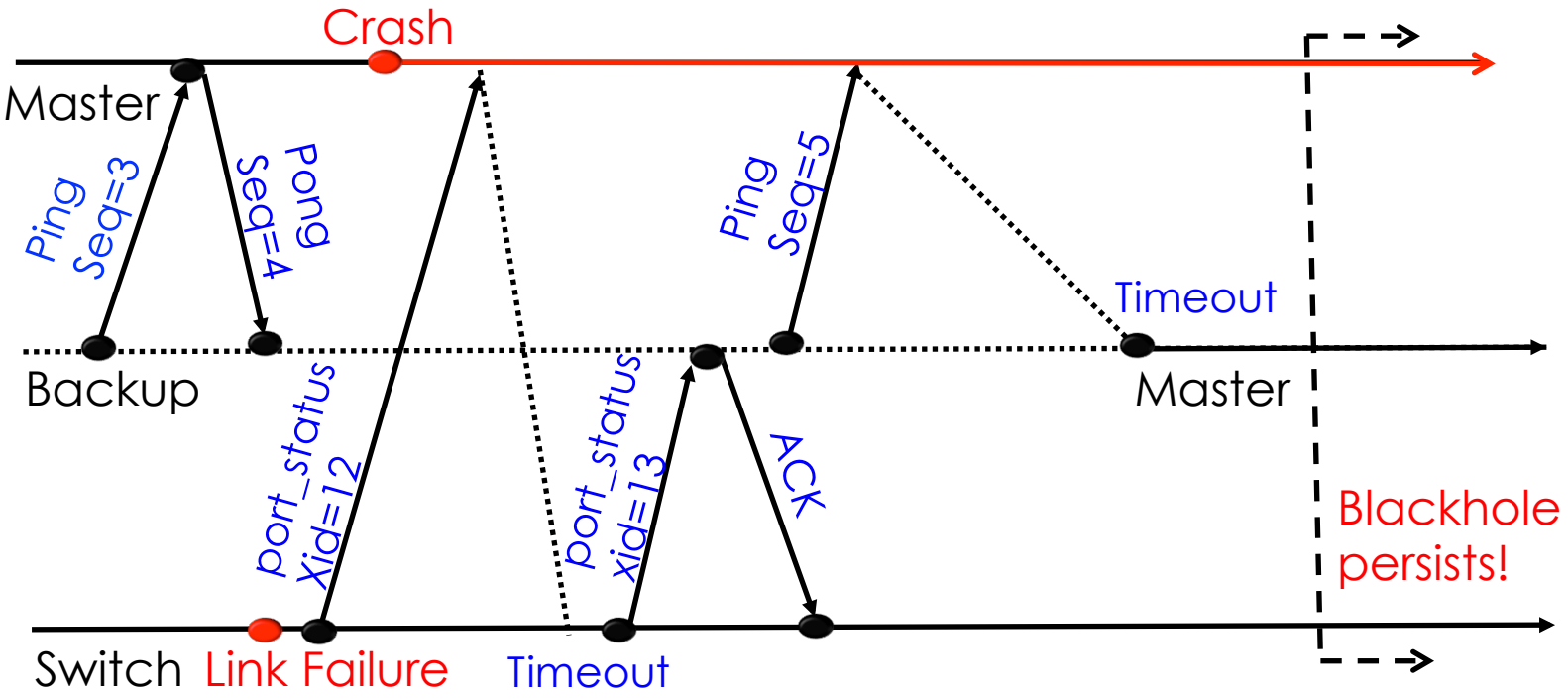
Second hypothesis

Third hypothesis

Fourth hypothesis...

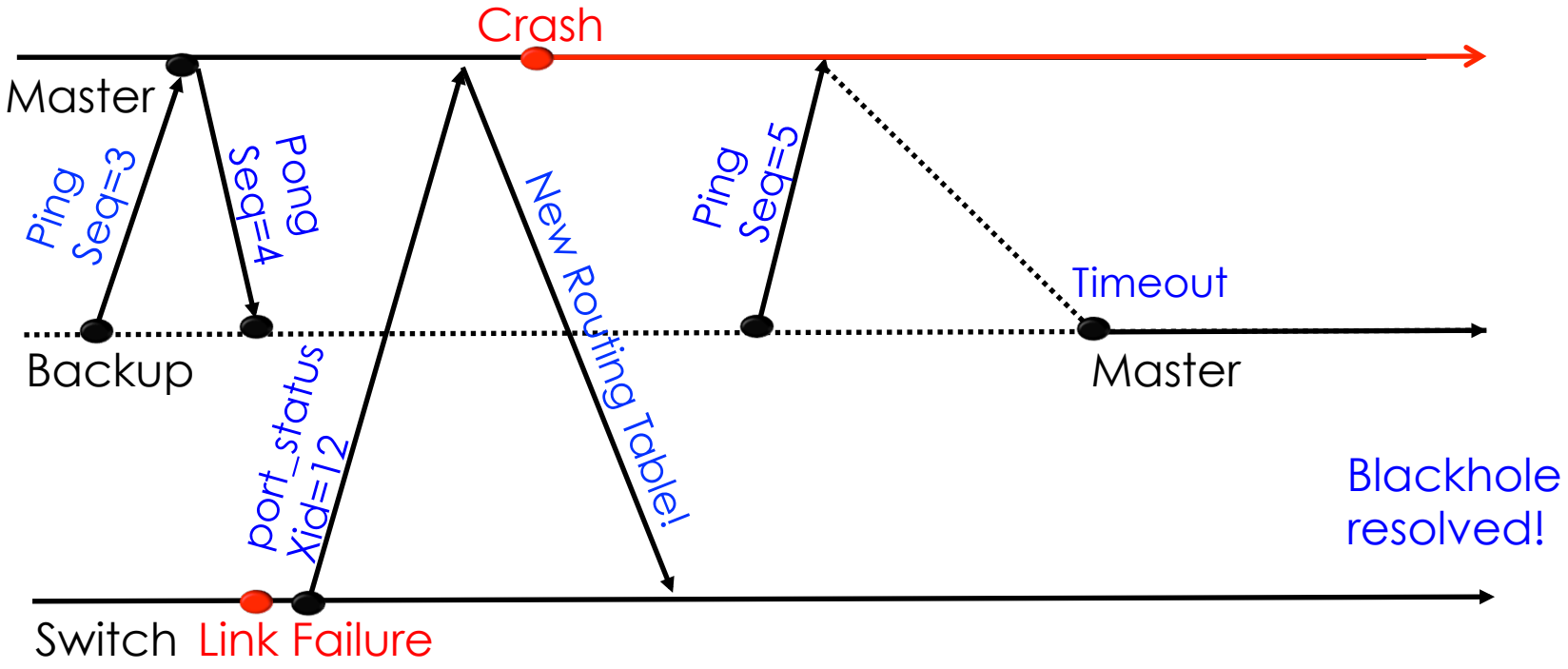
Replaying Pruned Inputs is Hard

Must consider internal events



Replaying Pruned Inputs is Hard

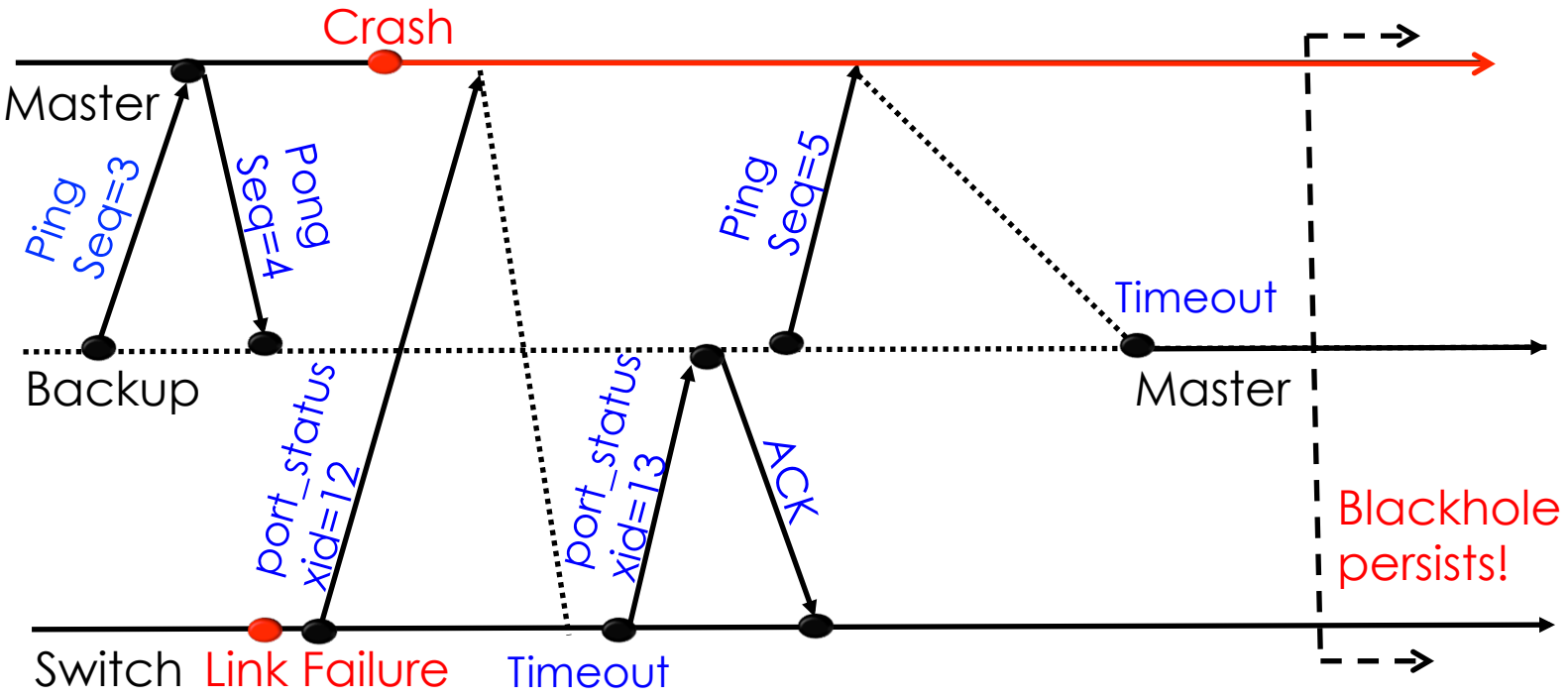
Must consider internal events



Pruning Alters Internal Events!

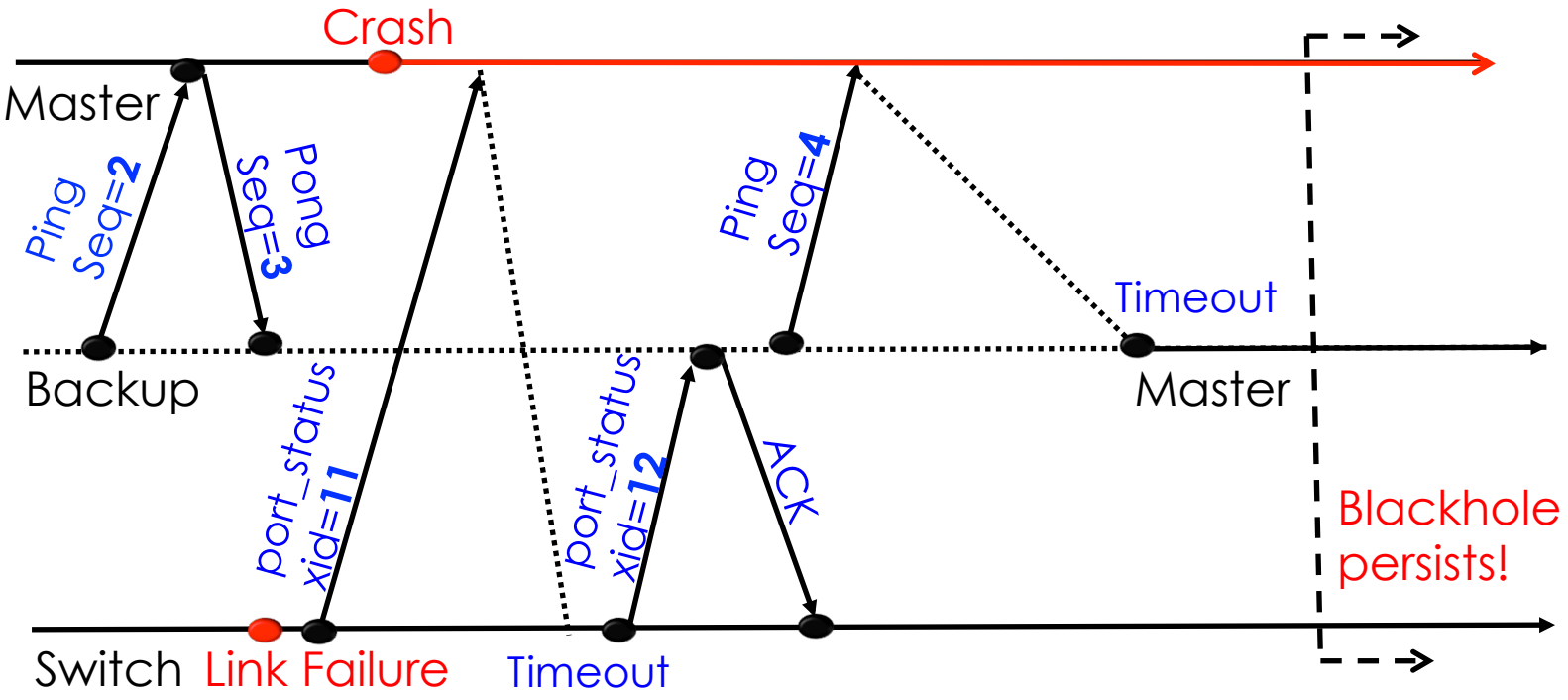
Pruning Alters Internal Events!

Prune Earlier Input..



Pruning Alters Internal Events!

Sequence Numbers Differ!



Solution: Equivalence Classes

Mask Over Extraneous Fields

Internal message	Masked values
OpenFlow headers	transaction id
OpenFlow FLOW_MODs	cookie, buffer id
Log statements	varargs parameters to printf

Other Subtleties: Old/New Events

Pruning inputs may also cause:

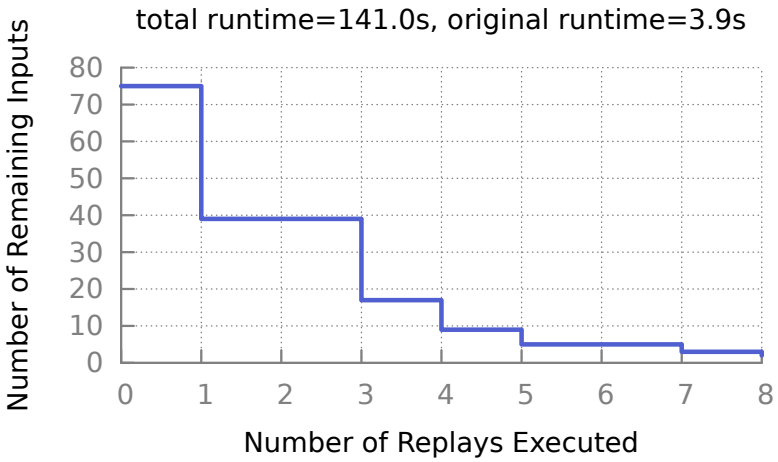
- Expected internal events to disappear
- New internal events to appear

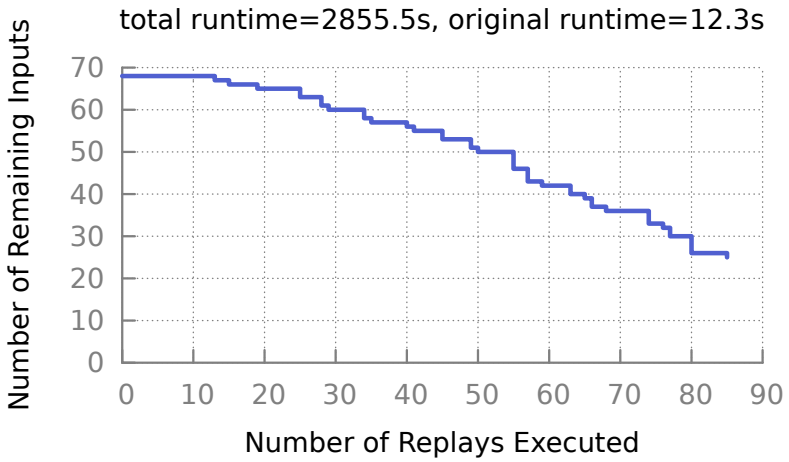
See our paper for more info: eecs.berkeley.edu/~rcs/research/sts.pdf

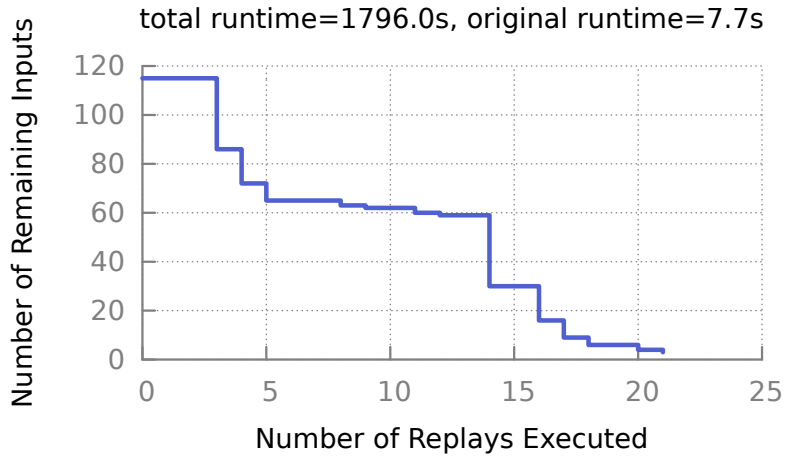
It Works!

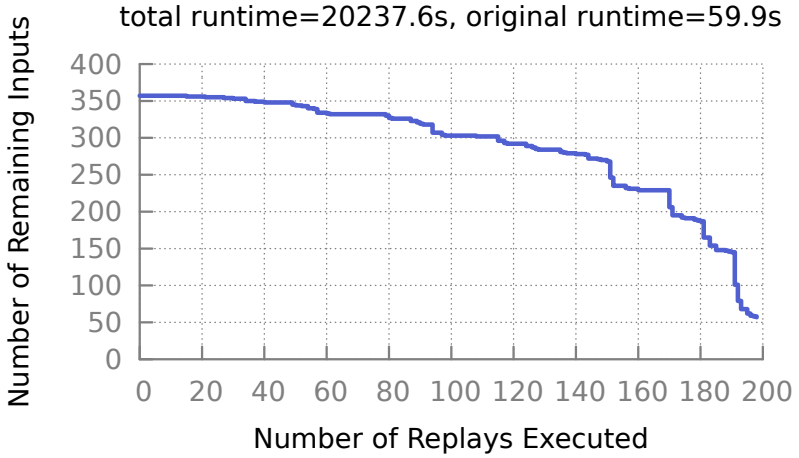
Bug Name	Topology	Replay Success Rate	Total Inputs	MCS Size
POX list removal	2 switch mesh	20/20	69	2
POX in-flight blackhole	2 switch mesh	15/20 [20/20*]	26	11
POX migration blackhole	4 switch mesh	20/20*	29	3
NOX discovery loop	4 switch mesh	18/20	150	18
Floodlight loop	3 switch mesh	15/50	284	36

It Works!


total runtime=141.0s, original runtime=3.9s
Number of Remaining Inputs
0
10
20
30
40
50
60
70
80
0
1
2
3
4
5
6
7
8
Number of Replays Executed


total runtime=2855.5s, original runtime=12.3s
Number of Remaining Inputs
0
10
20
30
40
50
60
70
0
10
20
30
40
50
60
70
80
90
Number of Replays Executed


total runtime=1796.0s, original runtime=7.7s
Number of Remaining Inputs
0
20
40
60
80
100
120
0
5
10
15
20
25
Number of Replays Executed


total runtime=20237.6s, original runtime=59.9s
Number of Remaining Inputs
0
50
100
150
200
250
300
350
400
0
20
40
60
80
100
120
140
160
180
200
Number of Replays Executed

Summary

- Goal: automatically troubleshoot control software bugs
- Approach: iteratively alter history
- Check us out:

ucb-sts.github.com/sts/

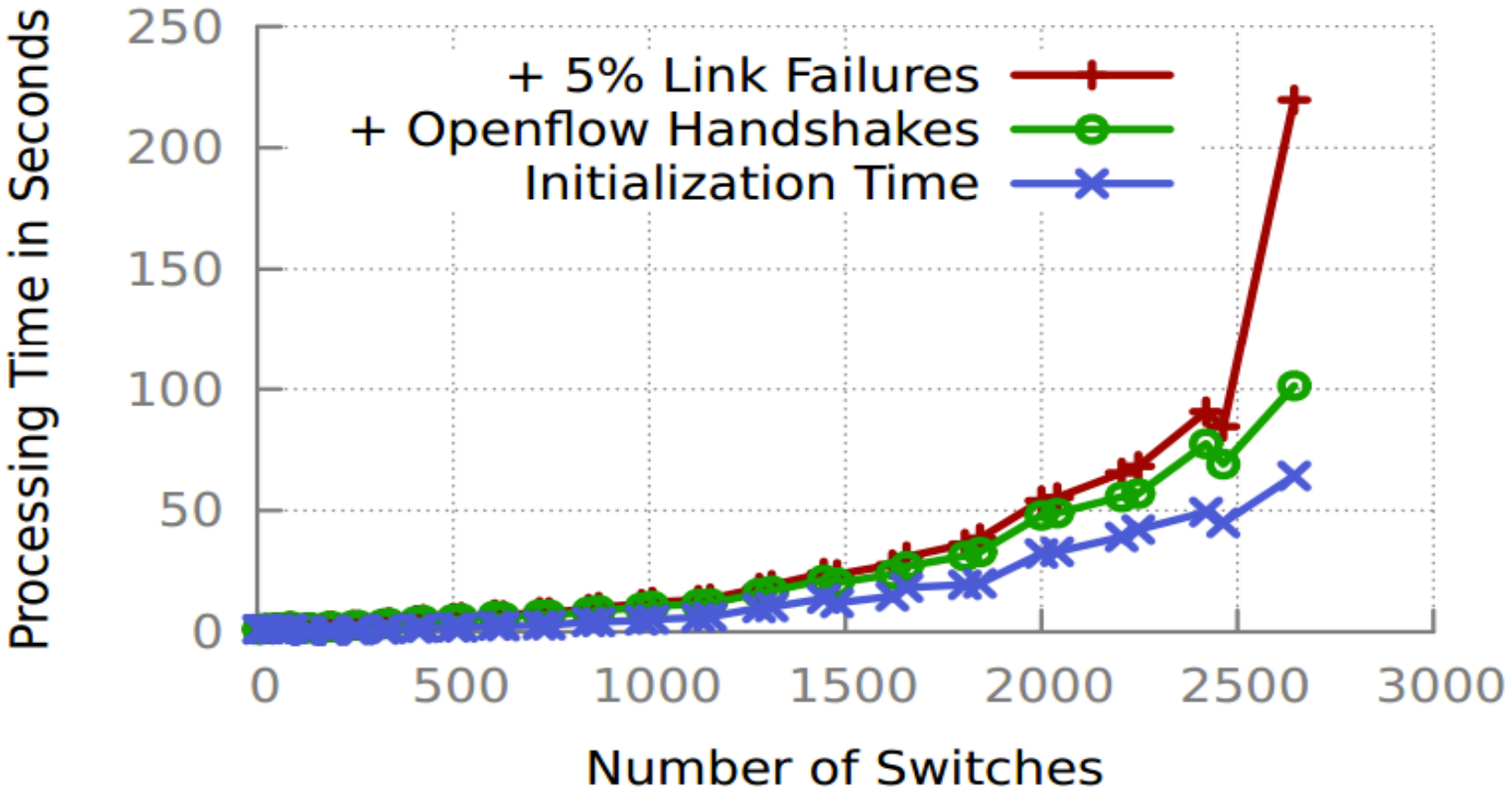
References

- [1] V. Soundararajan and K. Govil. Challenges in building scalable virtualized datacenter management. SIGOPS Operating Systems Review '10.

- [2] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta. VL2: a scalable and flexible data center network, Sec. 3.4. SIGCOMM '09.

Backup

Scalability



Complexity

Best Case	**Worst Case**
- Delta Debugging: Ω(log n) replays - Each replay: O(n) events - Total: Ω(nlog n)	- Delta Debugging: O(n) replays - Each replay: O(n) events - Total: $O(n^2)$

Opportunity: Parallelization

- Could run delta-debugging speculatively:
  - Each replay runs as a separate task
  - Join results at the end
- Decreases overall runtime by up to a factor of N

Coping With Non-Determinism

- Replay multiple times per subsequence
- Probability of not finding bug defined by Bernoulli Distribution:

$$f(p,n) = (1-p)^n$$

Assumptions of Delta Debugging

- *Monotonic:*

$$P \oplus C = \chi \Rightarrow P \oplus (C \cup C') \neq \checkmark$$

- *Unambiguous:*

$$P \oplus C = \chi \wedge P \oplus C' = \chi \Rightarrow P \oplus (C \cap C') \neq \checkmark$$

- *Consistent*

$$P \oplus C \neq ?$$

Local vs. Global Minimality

Definition 8 (Global minimum). *A set* $c \subseteq c_\chi$ *is called the* global minimum *of* c_χ *if:* $\forall c' \subseteq c_\chi \cdot \left(|c'| < |c| \Rightarrow test(c') \neq \chi\right)$ *holds.*

Definition 10 (n-minimal test case). *A test case* $c \subseteq c_\chi$ *is* n-minimal *if:* $\forall c' \subset c \cdot |c| - |c'| \leq n \Rightarrow \left(test(c') \neq \chi\right)$ *holds. Consequently,* c *is 1-minimal if* $\forall \delta_i \in c \cdot test\left(c - \{\delta_i\}\right) \neq \chi$ *holds.*

Limited Internal Event Visibility

- Interpose on logging library to gain visibility into internal state changes
- Still, may not obtain visibility into all relevant events!
- Extreme case: need visibility into thread scheduling decisions.

Forensic Analysis of Production Logs

- Logs need to capture causality: Lamport Clocks or accurate NTP
- Need clear mapping between input/internal events and simulated events
- Must remove redundantly logged events
- Might employ causally consistent snapshots to cope with length of logs

Related Work

- Delta Debugging, Sherlog, Rx, Chronus
- OFRewind, ndb
- Pip, X-trace
- Mininet, ns-3
- Model checkers: NICE, Mace
- Root cause analysis

Future Work

- Many improvements:
  - Parallelize delta debugging
  - Smarter delta debugging time splits
  - Split by topology (actors) rather than time
  - Split by event type
  - Compress time (override gettimeofday)

- Apply to other distributed systems