Test-suite for Automating Data-consistency checks on HBase

Pradeep S, Mallikarjun V

Flipkart

About Flipkart

- India’s largest online retailer:
  - 10M page visits a day
  - 2M shipments a day
  - 30M products across more than 70 categories
- Big Billion Days ($300M sales, top ranked app on Google Play Store)

Agenda

- Yak: HBase Cluster @ Flipkart
  - Need for scalable, multi-tenant, strongly consistent data stores
- Yak: Need for Data-correctness Guarantees
- Why Data-consistency Test-suite?
- How we did the Test-suite?

Yak: HBase Cluster @ Flipkart

- HBase for OLTP Key-Value store
- Bring-Your-Own-Box multi-tenancy on HBase
- RSGroup based isolation on HBase 1.2.4 & 2.1.3
- WAL based Change-Data-Capture into Kafka
- Stores critical data-sets in e-commerce like: Orders, Payments

Yak: Data-correctness Guarantees Required

- Read-Your-Own-Write consistency at single row level
- ‘Atleast-once’ guarantee in the change-data-capture
- Ordering guarantee in the change-data-capture
- Predictable recovery times, no data-loss upon failures
- Data-reliability for HBase admin operations

Why Data-consistency Test-suite?

- To enable faster upgrade cycles
- Releases dont degrading on the data-guarantees
- To assist in reproducing edge-cases scenarios
- To accommodate additional failure scenarios

Different Approaches for the Test-suite

- Algorithm testing: TLA+, etc.
- Code testing: Test suite of HBase, Hadoop, Zookeeper
- Testing on running cluster: Jepsen, ChaosMonkey & ITBLL of HBase, etc.

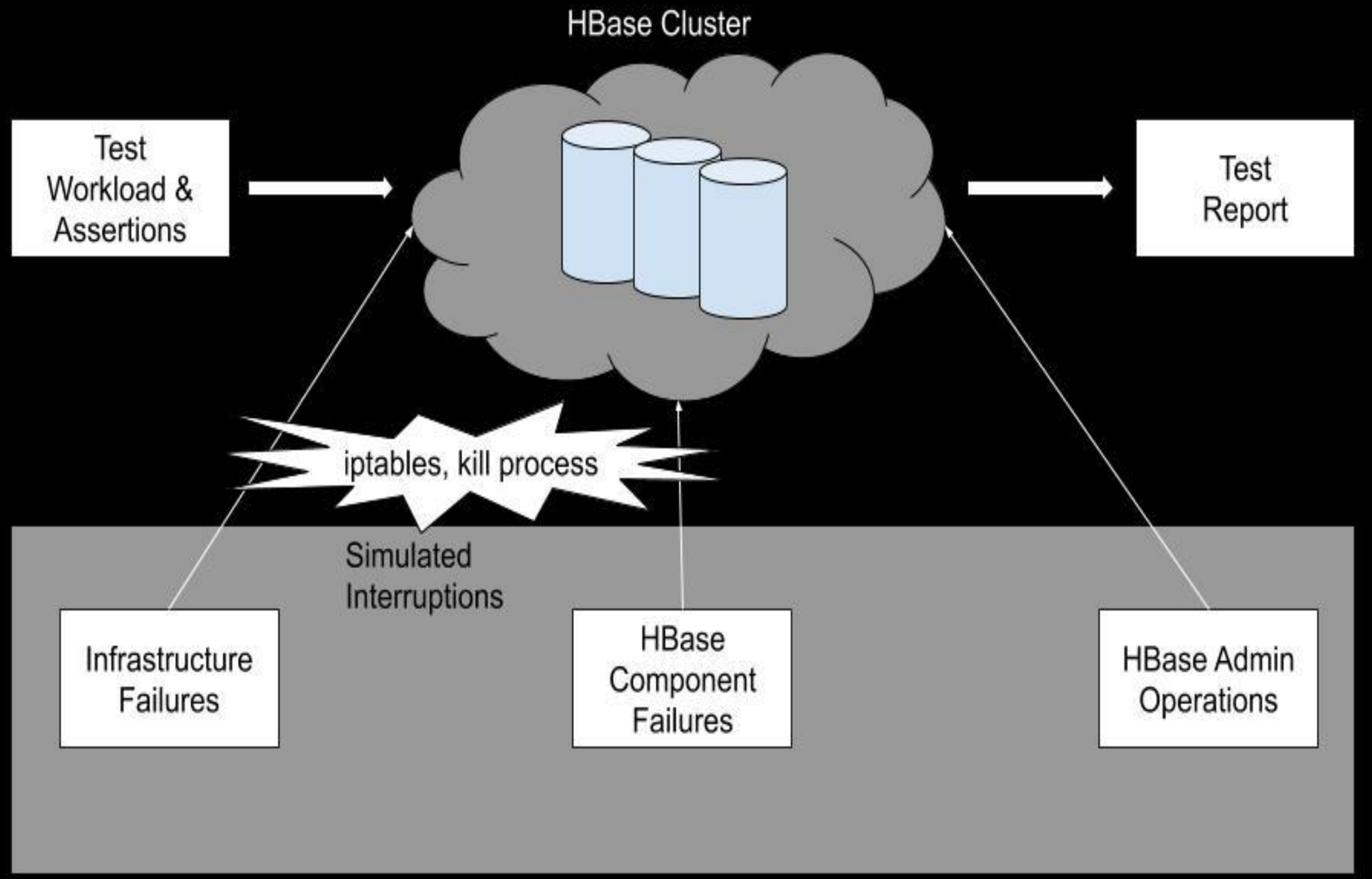
Different Approaches for the Test-suite

- ~~Algorithm testing: TLA+, etc.~~ - Implementation, Deployment NOT tested
- ~~Code testing: Test suite of HBase~~ - Deployment, Integration NOT tested
- Testing on running cluster: Jepsen, ChaosMonkey & ITBLL of HBase, etc.

Reason: Integrated test with { Algorithm + Code + Deployment }

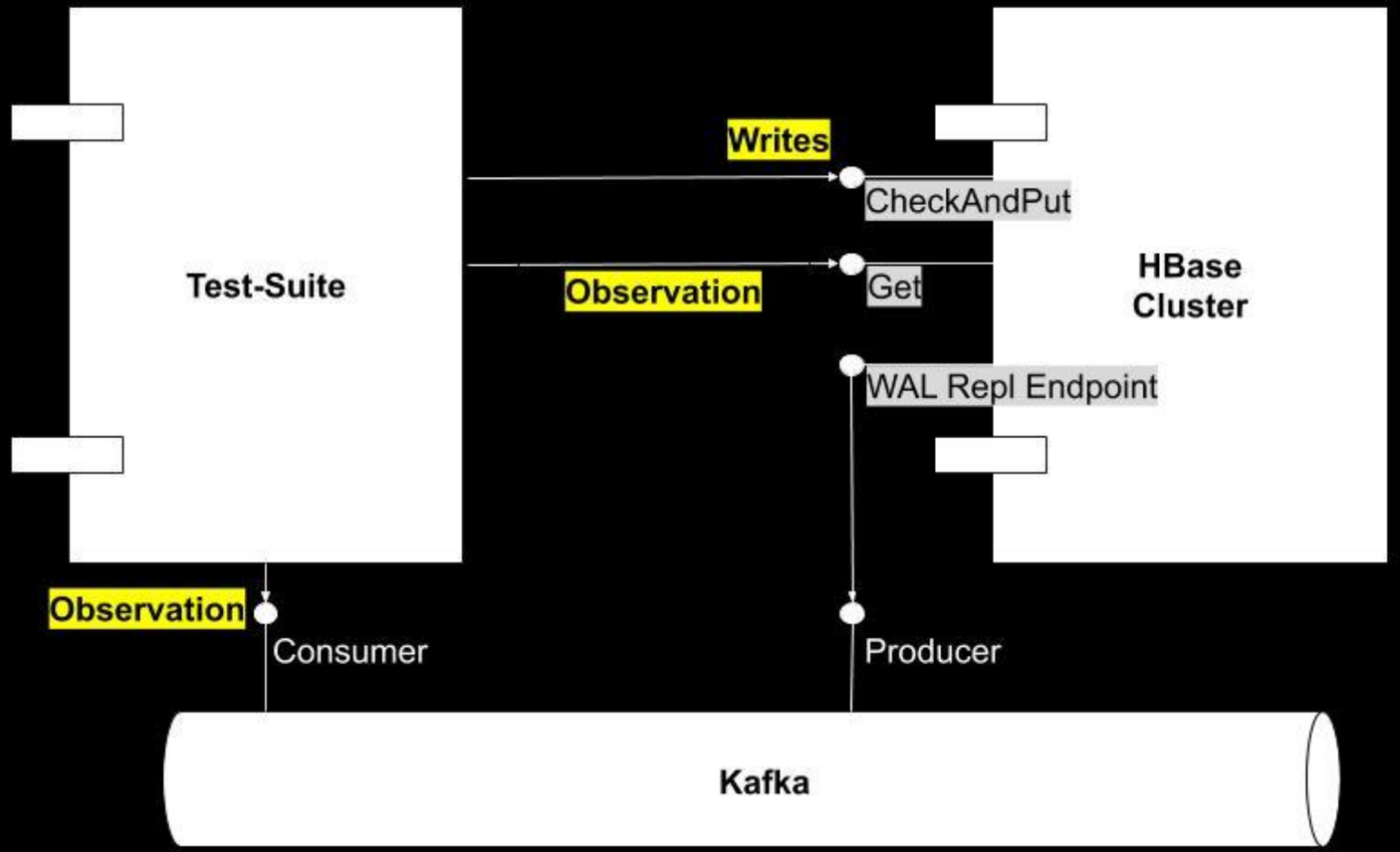
Test-Suite

- Jepsen, ChaosMonkey based
- Test workload on HBase
- Simulated Interrupts:
  - Infrastructure failures
  - HBase component failures
  - Admin operations
- Test Report

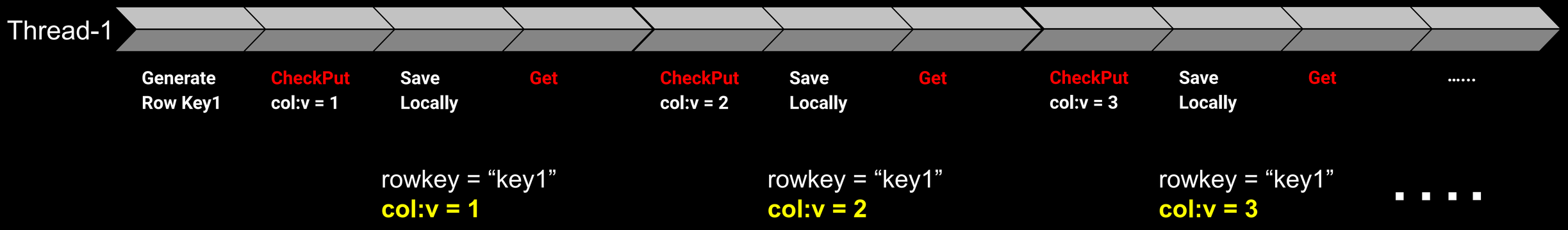


Test-Suite - Test Workload

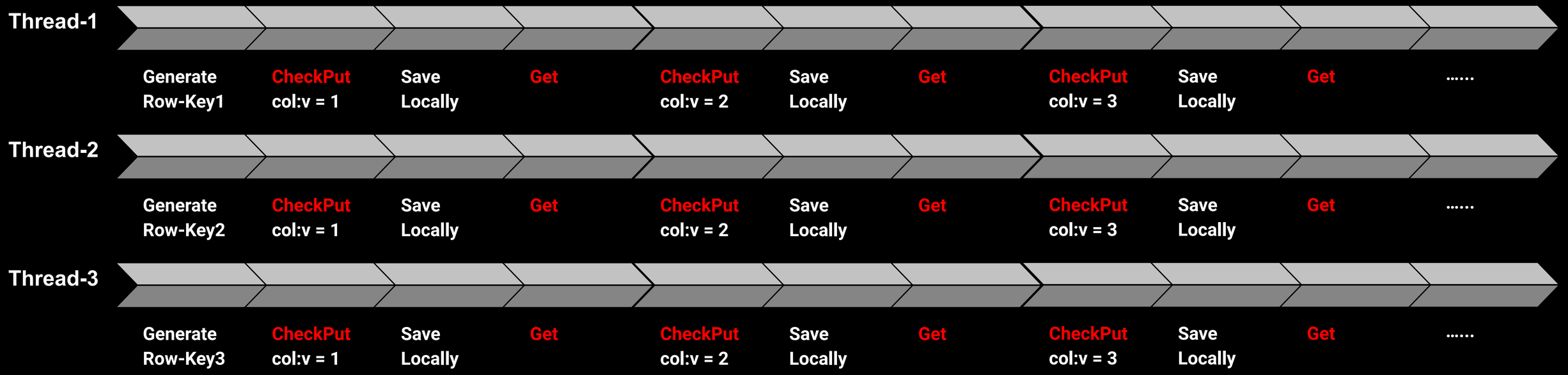
- Writes using CheckAndPut
- Observation points:
  - HBase Get
  - *Kafka consumer from CDC*



Test-Suite - Test Workload

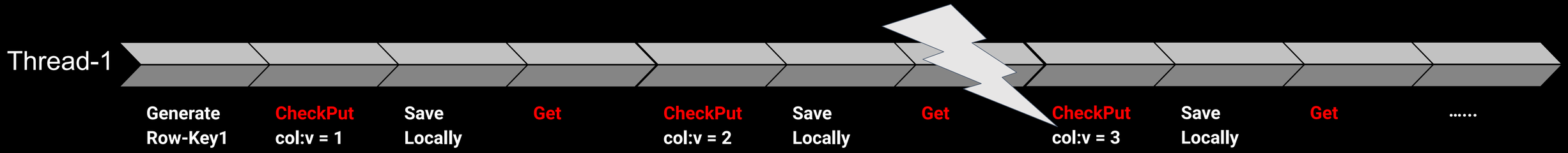


Test-Suite - Test Workload



N Threads… on M Machines

Test-Suite - Assertions

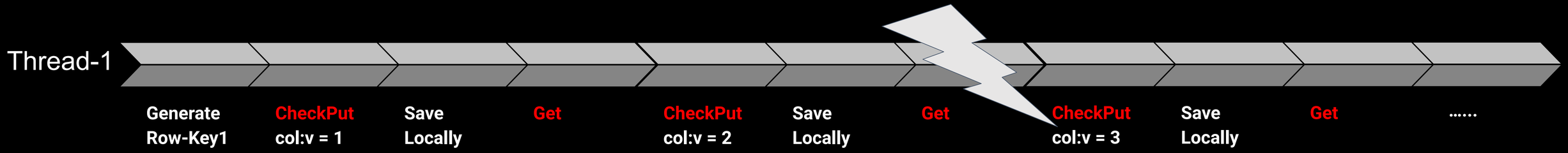


- Locally saved-state is compared against Get call result for no data-loss

For Key1 after every CheckAndPut:

```
Assert ( ThreadLocal<Version> == Version from HBase )
```

Test-Suite - Assertions



- Kafka listener reads the events of the key:

For a key1 having writes as: 1,2,3,4

1,2,2,2,3,4

1,2,3,2,3,4

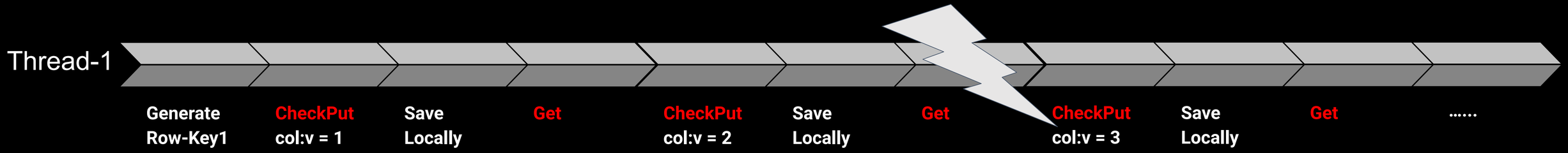
1,2,4,2,3,4 ✗

1,2,4 ✗

For the key1:

```
Assert (
read event version <=
latest seen version +1
)
```

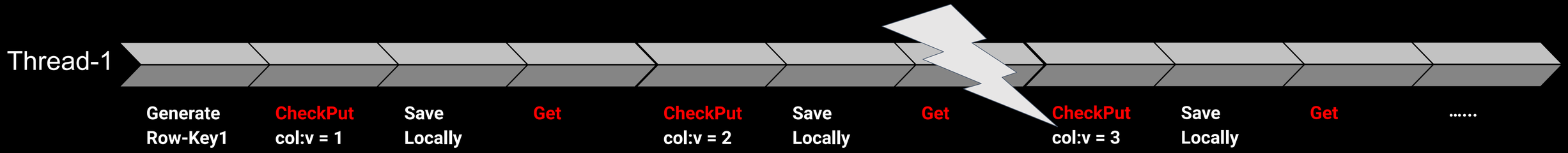
Test-Suite - Assertions



- HBase regions of a table are within the RSGroup nodes
- HDFS data-blocks stored within the RSGroup nodes

```
Assert (Region Assignment within RSGroup Nodes)
Assert (HDFS blocks within RSGroup Nodes)
```

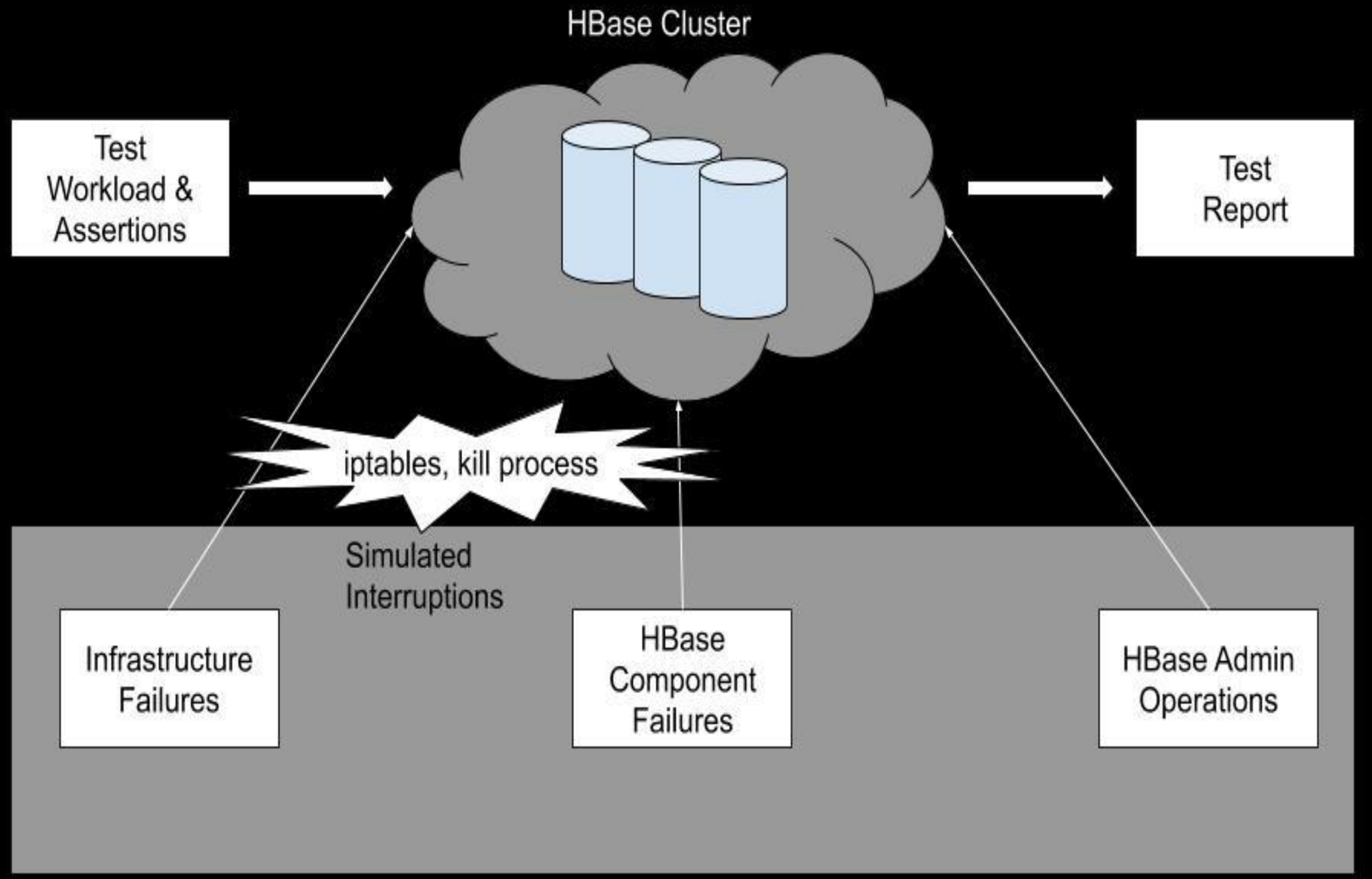
Test-Suite - Assertions



- Locally saved-state is compared against Get call result for no data-loss
- Kafka listener reads the events of the key:
  - To assert no data-loss as the versions are incremental
  - To assert no-ordering loss
- RSGroup isolation checks: WAL & HDFS data-blocks & Region Assignment

Interruptions

- Simulate issue & heal
- Types of Interruptions:
  - Infrastructure failures
  - HBase component failures
  - HBase admin operations



Interruptions - Infrastructure Failures

- Network failures:
  - Network partition/failure: within zookeeper, region-servers, namenode etc.
  - Packet loss: within zookeeper, master, region-servers, namenode etc.
  - Tools: iptables, tc, comcast
- Other failure modes - to be added:
  - Clock skew
  - Packet delays
  - Disk read-only etc

Interruptions - Component Failures

- **Kill hbase components:**
  - region-server, master-node, zookeeper
- **Kill hadoop components:**
  - data-node, journal-node, name-node
- **Node crash:**
  - region-server-node, master-node, name-node, zookeeper, journal-node

Interruptions - HBase Admin Operations

- Split/merge region
- Assign/move region
- Restart: region-server, data-node
- Stop: region-server, data-node, etc.
- Kafka properties reload for CDC

Issues Uncovered & Fixed

- Assign an already assigned region causing data-loss - Fixed in 2.x
- Ordering loss in tail process from WAL file upon a region failover - Fixed in 2.x
- WAL not-isolated across region-server group - HBASE-21641

Further Plans

- Migration to HBase ChaosMonkey for Interruptions
- Opensource the test-suite
- WAL push to Kafka - https://github.com/flipkart-incubator/hbase-sep

Thanks !

Appendix

Limitations

- Tests are not deterministic. Needs 100s of iterations.
- Doesn’t catch all the bugs. But catches the practical issues for the specific use-case.

Test-Suite - Sample Report

Yak Testbed (TestCount - 58)

Test Name	Interruptions	HbaseMetrics Output	KafkaMetrics Output	Running Time	Process Status	Overall Status
PutGetKafkaTest	(see below)	(see below)	(see below)	(see below)	(see below)	PASSED

PutGetKafkaTest — Interruptions

Interruption	Status	Nodes	StartTime	EndTime	Duration(ms)
mergeRegions	SUCCESS	3	08:36:02	08:36:30	27536
assignRegions	SUCCESS	2	08:36:02	08:36:24	21678

HbaseMetrics Output

Metric	Status	Failures
dataMismatchRH	YES	98
dataMismatchWH	NO	0
dataLoss	NO	0
checkPutFail	YES	264
checkPutException	NO	0
getFail	NO	0
inconsistencies	NO	0
walIsolated	YES	0
hBlockIsolated	NO	198
allRegionsOpen	YES	0
failuresAfterTest	NO	0

KafkaMetrics Output

Metric	Status	Failures
dataMismatchRH	YES	98
dataMismatchWH	NO	0
connectionMismatch	NO	0
dataLoss	NO	0
ordering	YES	0
repetition	NO	0

Running Time

RunningTime	StartTime	EndTime	Duration(ms)
getTime	08:35:44	08:37:17	92813
putTime	08:35:44	08:37:16	92077
kafkaTime	08:35:44	08:37:55	131393

Process Status

NodeType	host	host	host	host	host	host
REGION_SERVER	yak-hbase21-yaktest-dn-1	yak-hbase21-yaktest-dn-2	yak-hbase21-yaktest-dn-3	yak-hbase21-yaktest-dn-4	yak-hbase21-yaktest-dn-5	yak-hbase21-yaktest-dn-6
MASTER	yak-hbase21-test-nn-1	yak-hbase21-test-nn-2				
JOURNAL_NODE	yak-hbase21-test-zk-1	yak-hbase21-test-zk-2	yak-hbase21-test-zk-3			
NAME_NODE	yak-hbase21-test-nn-1	yak-hbase21-test-nn-2				
ZOOKEEPER	yak-hbase21-test-zk-1	yak-hbase21-test-zk-2	yak-hbase21-test-zk-3			
DATA_NODE	yak-hbase21-yaktest-dn-1	yak-hbase21-yaktest-dn-2	yak-hbase21-yaktest-dn-3	yak-hbase21-yaktest-dn-4	yak-hbase21-yaktest-dn-5	yak-hbase21-yaktest-dn-6

Overall Status: PASSED

PutGetKafkaTest — Interruptions

Interruption	Status	Nodes	StartTime	EndTime	Duration(ms)
mergeRegions	SUCCESS	3	08:36:02	08:36:30	27536
assignRegions	SUCCESS	2	08:36:02	08:36:24	21678

HbaseMetrics Output

Metric	Status	Failures
dataMismatchRH	YES	98
dataMismatchWH	NO	0
dataLoss	NO	0
checkPutFail	YES	264
checkPutException	NO	0
getFail	NO	0
inconsistencies	NO	0
walIsolated	YES	0
hBlockIsolated	NO	198
allRegionsOpen	YES	0
failuresAfterTest	NO	0

KafkaMetrics Output

Metric	Status	Failures
dataMismatchRH	YES	98
dataMismatchWH	NO	0
connectionMismatch	NO	0
dataLoss	NO	0
ordering	YES	0
repetition	NO	0

Running Time

RunningTime	StartTime	EndTime	Duration(ms)
getTime	08:35:44	08:37:17	92813
putTime	08:35:44	08:37:16	92077
kafkaTime	08:35:44	08:37:55	131393

Process Status

NodeType	host	host	host	host	host	host
REGION_SERVER	yak-hbase21-yaktest-dn-1	yak-hbase21-yaktest-dn-2	yak-hbase21-yaktest-dn-3	yak-hbase21-yaktest-dn-4	yak-hbase21-yaktest-dn-5	yak-hbase21-yaktest-dn-6
MASTER	yak-hbase21-test-nn-1	yak-hbase21-test-nn-2				
JOURNAL_NODE	yak-hbase21-test-zk-1	yak-hbase21-test-zk-2	yak-hbase21-test-zk-3			
NAME_NODE	yak-hbase21-test-nn-1	yak-hbase21-test-nn-2				
ZOOKEEPER	yak-hbase21-test-zk-1	yak-hbase21-test-zk-2	yak-hbase21-test-zk-3			
DATA_NODE	yak-hbase21-yaktest-dn-1	yak-hbase21-yaktest-dn-2	yak-hbase21-yaktest-dn-3	yak-hbase21-yaktest-dn-4	yak-hbase21-yaktest-dn-5	yak-hbase21-yaktest-dn-6

Overall Status: PASSED

PutGetKafkaTest — Interruptions

Interruption	Status	Nodes	StartTime	EndTime	Duration(ms)
mergeRegions	SUCCESS	3	08:36:02	08:36:30	27536
assignRegions	SUCCESS	2	08:36:02	08:36:24	21678

HbaseMetrics Output

Metric	Status	Failures
dataMismatchRH	YES	98
dataMismatchWH	NO	0
dataLoss	NO	0
checkPutFail	YES	264
checkPutException	NO	0
getFail	NO	0
inconsistencies	NO	0
walIsolated	YES	0
hBlockIsolated	NO	198
allRegionsOpen	YES	0
failuresAfterTest	NO	0

KafkaMetrics Output

Metric	Status	Failures
dataMismatchRH	YES	98
dataMismatchWH	NO	0
connectionMismatch	NO	0
dataLoss	NO	0
ordering	YES	0
repetition	NO	0

Running Time

RunningTime	StartTime	EndTime	Duration(ms)
getTime	08:35:44	08:37:17	92813
putTime	08:35:44	08:37:16	92077
kafkaTime	08:35:44	08:37:55	131393

Process Status

NodeType	host	host	host	host	host	host
REGION_SERVER	yak-hbase21-yaktest-dn-1	yak-hbase21-yaktest-dn-2	yak-hbase21-yaktest-dn-3	yak-hbase21-yaktest-dn-4	yak-hbase21-yaktest-dn-5	yak-hbase21-yaktest-dn-6
MASTER	yak-hbase21-test-nn-1	yak-hbase21-test-nn-2				
JOURNAL_NODE	yak-hbase21-test-zk-1	yak-hbase21-test-zk-2	yak-hbase21-test-zk-3			
NAME_NODE	yak-hbase21-test-nn-1	yak-hbase21-test-nn-2				
ZOOKEEPER	yak-hbase21-test-zk-1	yak-hbase21-test-zk-2	yak-hbase21-test-zk-3			
DATA_NODE	yak-hbase21-yaktest-dn-1	yak-hbase21-yaktest-dn-2	yak-hbase21-yaktest-dn-3	yak-hbase21-yaktest-dn-4	yak-hbase21-yaktest-dn-5	yak-hbase21-yaktest-dn-6

Overall Status: PASSED