

If At First You Don't Succeed, Try, Try, Again...?

Insights and LLM-informed Tooling for Detecting Retry Bugs in Software Systems

Bogdan “Bo” Stoica*, Utsav Sethi*, Yiming Su, Cyrus Zhou,
Shan Lu, Jonathan Mace, Madan Musuvathi, Suman Nath

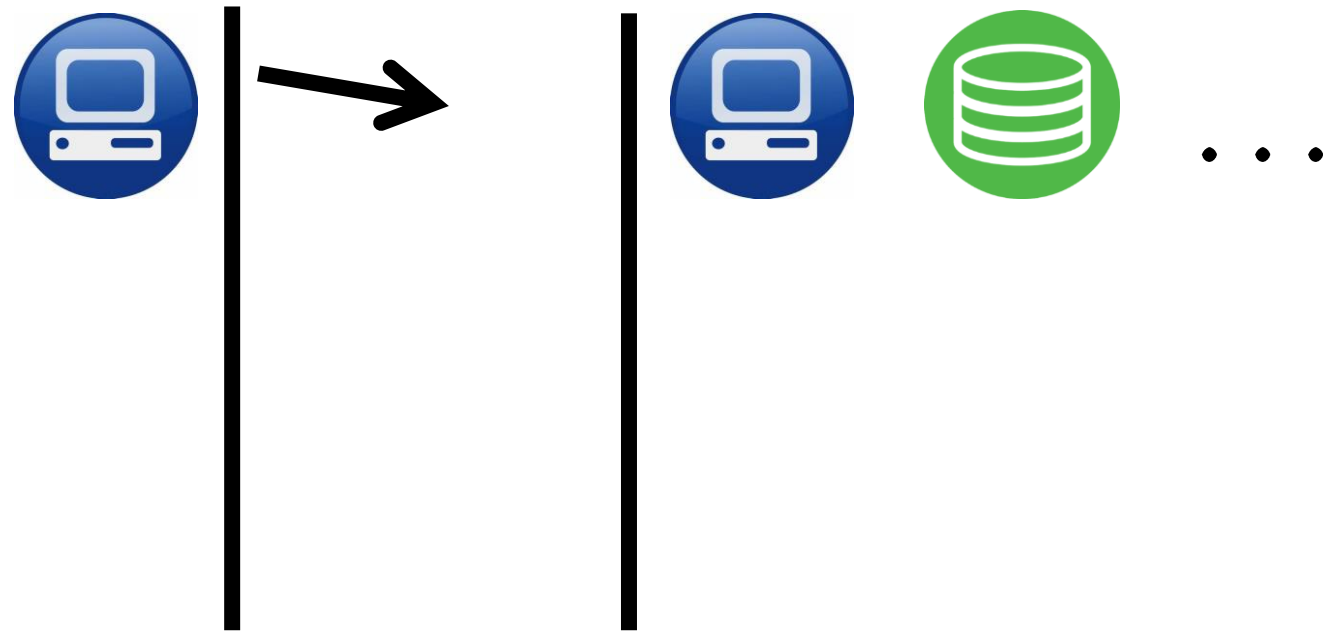


** equal contribution*

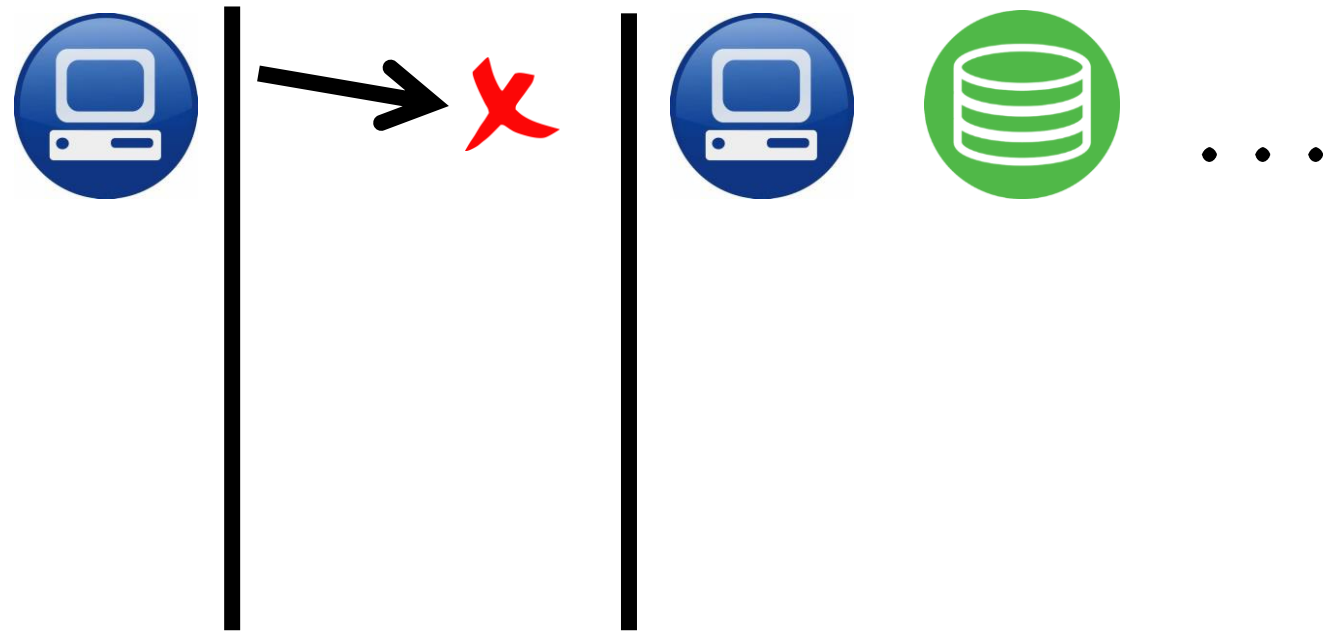
Retry, task re-execution on errors



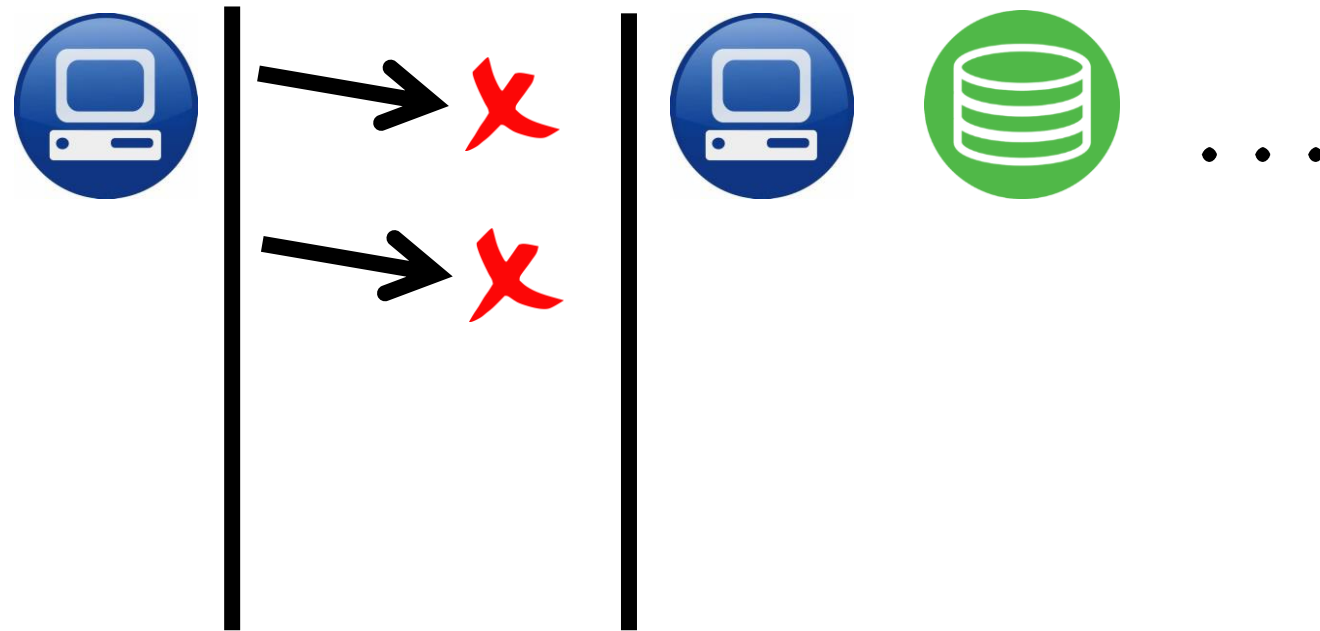
Retry, task re-execution on errors



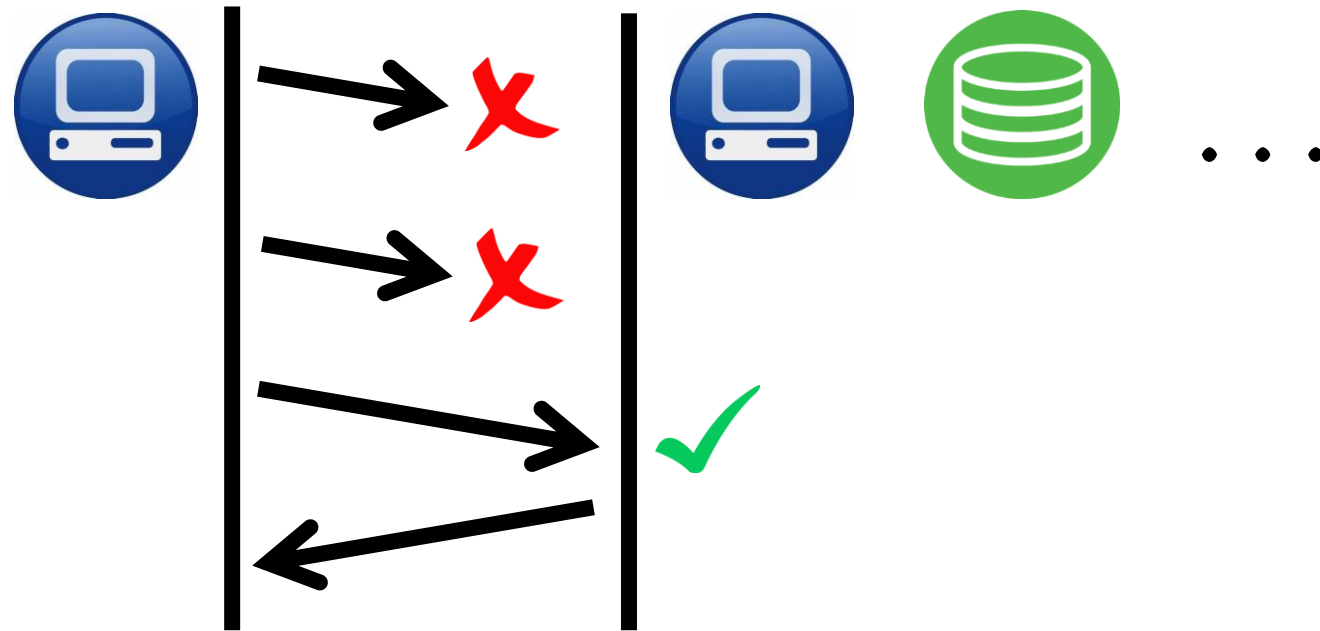
Retry, task re-execution on errors



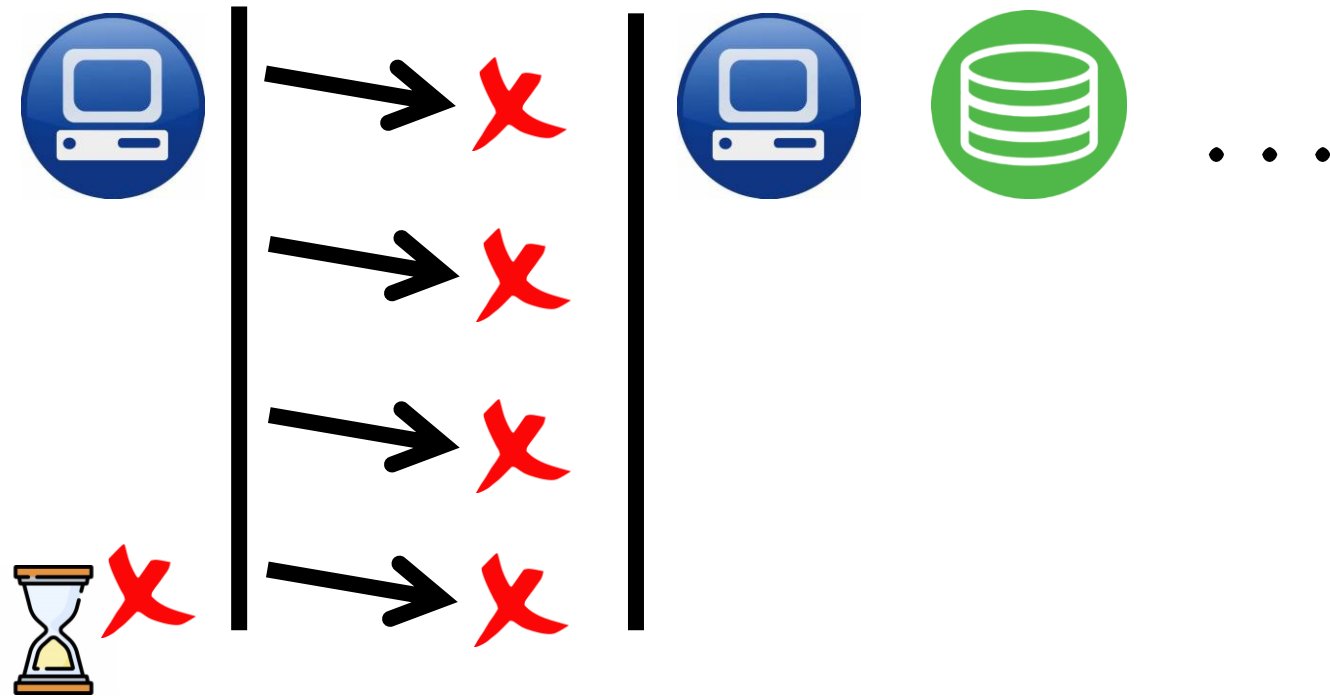
Retry, task re-execution on errors



Retry, task re-execution on errors



Retry, task re-execution on errors



Retry is crucial in modern systems ...

 **Improves systems resilience**

Retry is crucial in modern systems ...



Improves systems resilience



Last line of defense against soft(hard)ware errors

Retry is challenging to implement ...

Retry is challenging to implement ...

 **No standard code structure**

Retry is challenging to implement ...






No standard code structure



Non-trivial to determine which errors to retry or not

Retry is challenging to implement ...

-  **No standard code structure**
-  **Non-trivial to determine which errors to retry or not**
-  **Need to handle side-effects of failed tasks**

Retry is difficult to test ...

 **High testing coverage is difficult to achieve**

Retry is difficult to test ...

 **High testing coverage is difficult to achieve**

 **Hard-to-replicate transient triggers**

Fault retry impacts end users ...

Fault retry impacts end users ...



HBase / HBASE-20492

UnassignProcedure is stuck in retry loop on region stuck in OPENING state



Hadoop HDFS / HDFS-17590

NullPointerException triggered in `createBlockReader` during retry iteration

Fault retry impacts end users ...



HBase / HBASE-20492

UnassignProcedure is stuck in retry loop on region stuck in OPENING state



Spark / SPARK-28163

Kafka ignores user configuration on FETCH_OFFSET_NUM_RETRY

IOException triggered in `createBlockReader` during retry iteration

Fault retry impacts end users ...



HBase / HBASE-20492

UnassignProcedure is stuck in retry loop on region stuck in OPENING state



Spark / SPARK-28163

Kafka ignores user configuration on FETCH_OFFSET_NUM_RETRY

IOException triggered : ...





Hadoop HDFS / HDFS-15439

Setting dfs.mover.retry.max.attempts to negative value will retry forever.


...ing retry iteration

Fault retry impacts end users ...



 HBase / HBASE-7 Unassignment  Spark / SPARK-28163 retry loop on region stuck in OPENING state

Kafka ignores user configuration on FETCH_OFFSET_NUM_RETRY

 Spark / SPARK-28163 Kafka ignores user configuration on FETCH_OFFSET_NUM_RETRY
retry exception` triggered`

 Hadoop HDFS / HDFS-15439 Setting dfs.mover.retry.max.attempts to negative value will retry forever.
retry iteration


Fault retry impacts end users ...

 HBase / HBASE-7 Unassigned
 Spark / SPARK-28163 Kafka ignores user configuration
retry loop on region stuck in OPENING state



Now, an opportune time to investigate retry bugs

 Spark / SPARK-28163 Kafka ignores user configuration on FETCH_OFFSET_NUM_RETRIES_RETRY
KafkaException` triggered ...

 Hadoop HDFS / HDFS-15439 Setting dfs.mover.retry.max.attempts to negative value will retry forever.
retry iteration

Key contributions



A systematic study of real-world retry bugs

Key contributions



A systematic study of real-world retry bugs



A new framework for detecting retry bugs

Key contributions



A systematic study of real-world retry bugs



A new framework for detecting retry bugs



Static analysis enhanced by large language models

Key contributions



A systematic study of real-world retry bugs



A new framework for detecting retry bugs



Static analysis enhanced by large language models



Re-purposed retry-focused unit testing

Understanding the challenges of retry

70 bug reports

👉 Reported in last 5 years

👉 Confirmed & fix

8 open-source systems



Key findings from bug reports

 **Retry implemented in different, complex ways**

Key findings from bug reports

 **Retry implemented in different, complex ways**

Loops

Finite state machines

Queue re-append

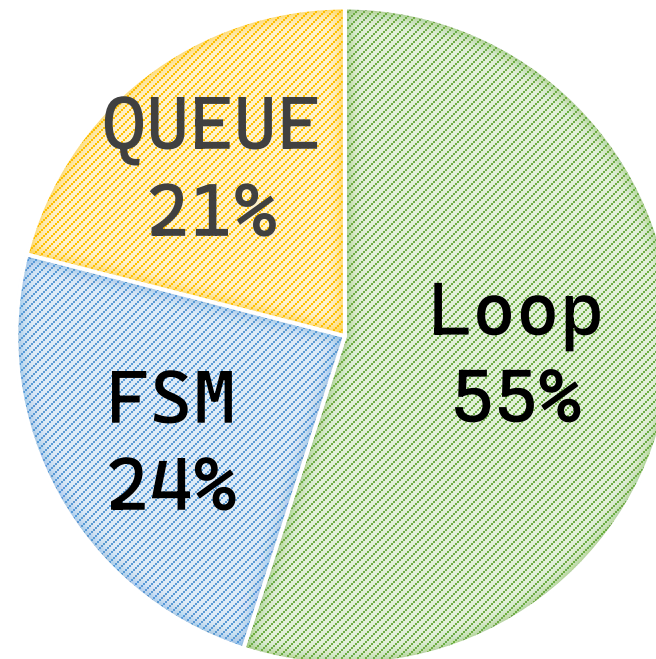
Key findings from bug reports

👉 **Retry implemented in different, complex ways**

Loops

Finite state machines

Queue re-append



Key findings from bug reports

 Existing tests not suited to uncover retry bugs

Key findings from bug reports



Existing tests not suited to uncover retry bugs

0.1 – 0.5% tests labeled for testing retry

Largely check if exception is caught, not buggy behavior

Key findings from bug reports

 **Retry bugs at both policy and mechanism levels**

IF should retry on error (**policy**)

WHEN to retry, # times, frequency (**policy**)

Key findings from bug reports

 **Retry bugs at both policy and mechanism levels**

IF should retry on error (**policy**)

WHEN to retry, # times, frequency (**policy**)

HOW to implement to handle side-effects (**mechanism**)

Key findings from bug reports



Retry bugs at both policy and mechanism levels

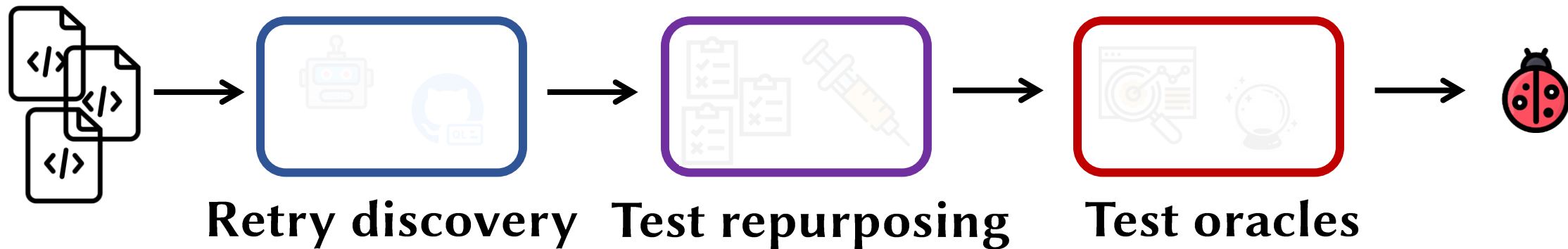
36% **IF** should retry on error (**policy**)

33% **WHEN** to retry, # times, frequency (**policy**)

31% **HOW** to implement to handle side-effects (**mechanism**)

WASABI, a toolkit for finding retry bugs

WASABI, a toolkit for finding retry bugs



Dynamic testing

WASABI, a toolkit for finding retry bugs



HOW (mechanism) & **WHEN** (policy)



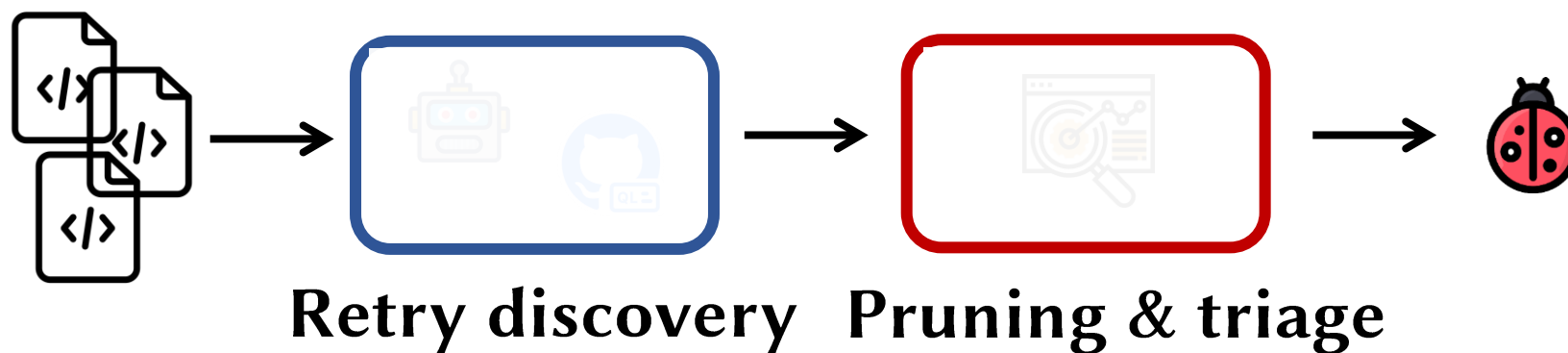
Retry discovery

Test repurposing

Test oracles

Dynamic testing

WASABI, a toolkit for finding retry bugs



Static checking

WASABI, a toolkit for finding retry bugs

Static checking

IF (policy) & WHEN (policy)

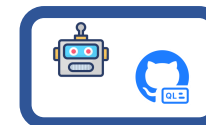
Retry discovery Pruning & triage

WASABI, a toolkit for finding retry bugs



Dynamic testing

Which code snippet performs retry?



Retry discovery

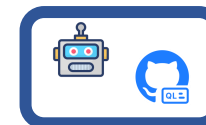
HDFS/DFSInputStream.java

```
bool createBlockReader():  
1. while (true) {  
2.   try {  
3.     block = refreshBlock(block);  
4.     dnInfo = getDNInfoFor(block);  
5.     if (dnInfo == null)  
6.       break;  
7.   }  
8.   catch (IOException e) {  
9.     LOG("Failed to connect to "+  
        dnInfo.addr + "Retried " +  
        ++retryCount + " times");  
10.    addToDeadNodes(dnInfo.info);  
11.  }  
12. }
```

HDFS/SecondaryNameNode.java

```
void parse():  
1. while (true) {  
2.   try {  
3.     block = parser.getBlock(buff);  
4.     length += block.length();  
5.     if (length >= buff.len) {  
6.       break;  
7.     }  
8.   }  
9.   catch (IOException e) {  
10.    throw new  
        ParseException(  
        "Unable to parse buffer.);  
11.  }  
12. }
```

Which code snippet performs retry?



Retry discovery

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "+
10.        dnInfo.addr + "Retried " +
11.        ++retryCount + " times");
12.     addToDeadNodes(dnInfo.info);
13.   }
14. }
```

HDFS/SecondaryNameNode.java

```
void parse():
```

```
1. while (true) {
2.   try {
3.     block = parser.getBlock(buff);
4.     length += block.length();
5.     if (length >= buff.len) {
6.       break;
7.     }
8.   }
9.   catch (IOException e) {
10.    throw new
11.        ParseException(
12.            "Unable to parse buffer.");
13.   }
14. }
```

Which code snippet performs retry?



Retry discovery

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

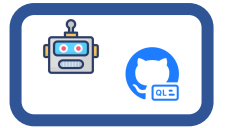
```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "+
10.        dnInfo.addr + "Retried " +
11.        ++retryCount + " times");
12.     addToDeadNodes(dnInfo.info);
13.   }
14. }
```

HDFS/ScoundaryNameNode.java

```
void parse():
```

```
1. while (true) {
2.   try {
3.     block = parser.getBlock(buff);
4.     length += block.length();
5.     if (length >= buff.len) {
6.       break;
7.     }
8.   }
9.   catch (IOException e) {
10.    throw new
11.        ParseException(
12.            "Unable to parse buffer.");
13.   }
14. }
```

Which code snippet performs retry?



Retry discovery

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "+
10.        dnInfo.addr + "Retried" +
11.        ++retryCount + " times");
12.     addToDeadNodes(dnInfo.info);
13.   }
14. }
```

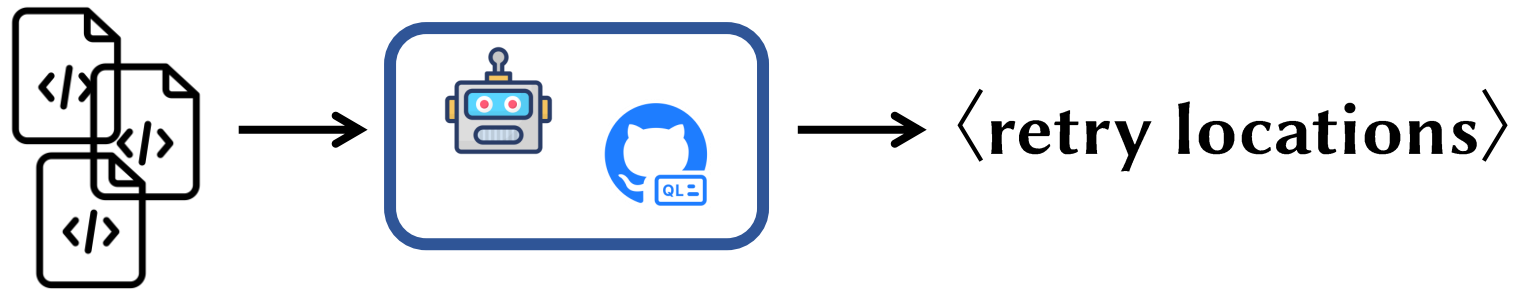
Retry

```
void parse():
```

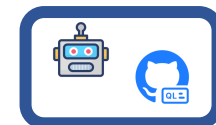
```
1. while (true) {
2.   try {
3.     block = parser.getBlock(buff);
4.     length += block.length();
5.     if (length >= buff.len) {
6.       break;
7.     }
8.   }
9.   catch (IOException e) {
10.    throw new
11.        ParseException(
12.            "unable to parse buffer.");
13.   }
14. }
```

Not retry

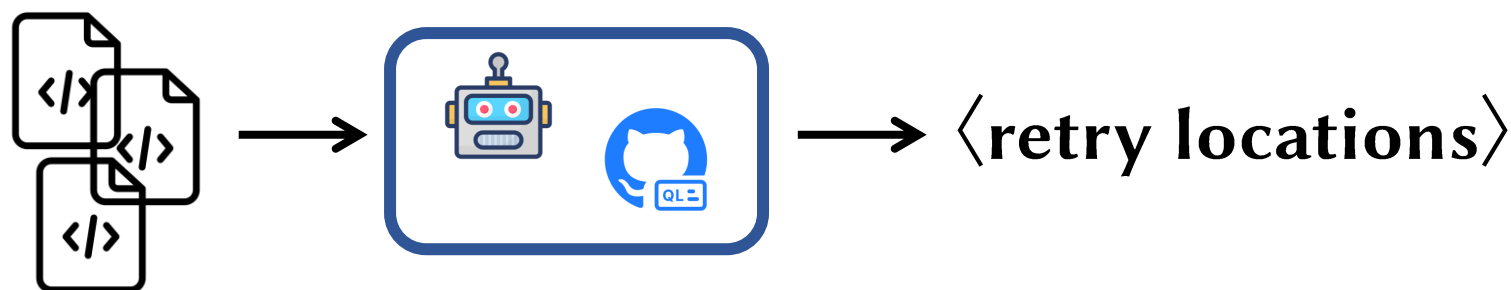
Identifying retry logic



Identifying retry logic

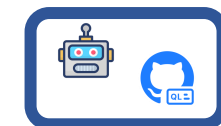


Retry discovery

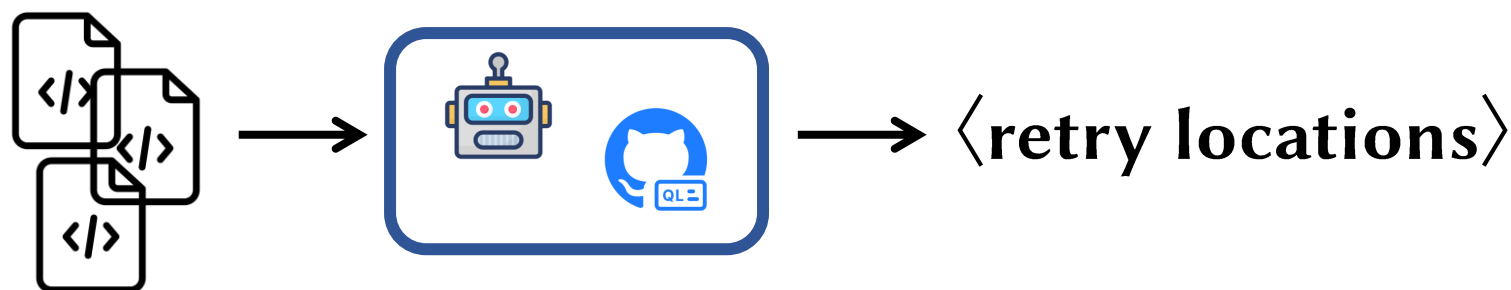


Fuzzy code comprehension capabilities of LLMs

Identifying retry logic



Retry discovery



Fuzzy code comprehension capabilities of LLMs

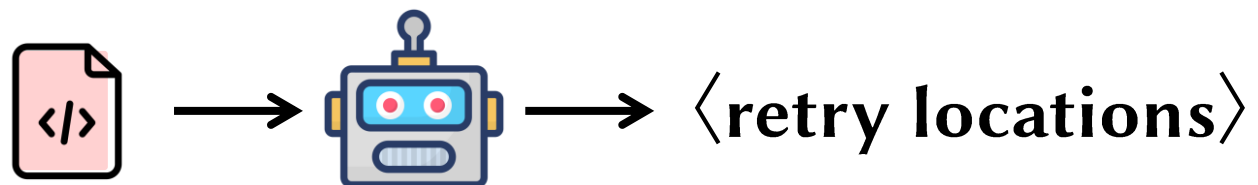


Traditional data & control flow analysis

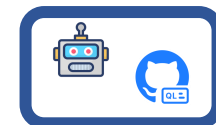
Identifying retry logic



GPT-4



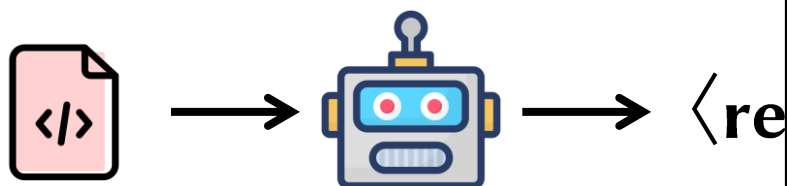
Identifying retry logic



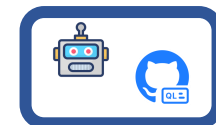
Retry discovery

Q: Does the following code perform retry anywhere ? Answer (YES) or (NO) .

GPT-4



Identifying retry logic

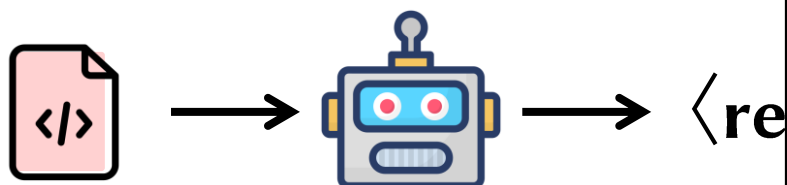


Retry discovery

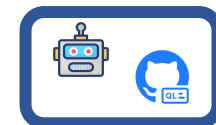
Q: Does the following code perform retry anywhere ? Answer (YES) or (NO) .

** REMEMBER that retry mechanisms can be implemented through for or while loops or code structures like state machines and queues .**

GPT-4



Identifying retry logic



Retry discovery

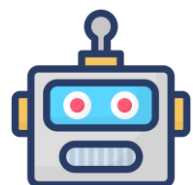
Q: Does the following code perform retry anywhere ? Answer (YES) or (NO) .

** REMEMBER that retry mechanisms can be implemented through for or while loops or code structures like state machines and queues .**

- Say NO if the file only `_defines_` or `_creates_` retry policies , or only passes retry parameters to other builders or c'tors.

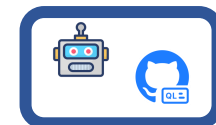
- Say NO if the file does not check for exception or errors before retry .

GPT-4



<re

Identifying retry logic



Retry discovery

Q; Does the following code perform retry anywhere ? Answer (YES) or (NO) .

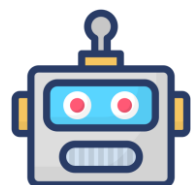
** REMEMBER that retry mechanisms can be implemented through for or while loops or code structures like state machines and queues .**

– Say NO if the file only `_defines_` or `_creates_` retry policies , or only passes retry parameters to other builders or c'tors.

– Say NO if the file does not check for exception or errors before retry .

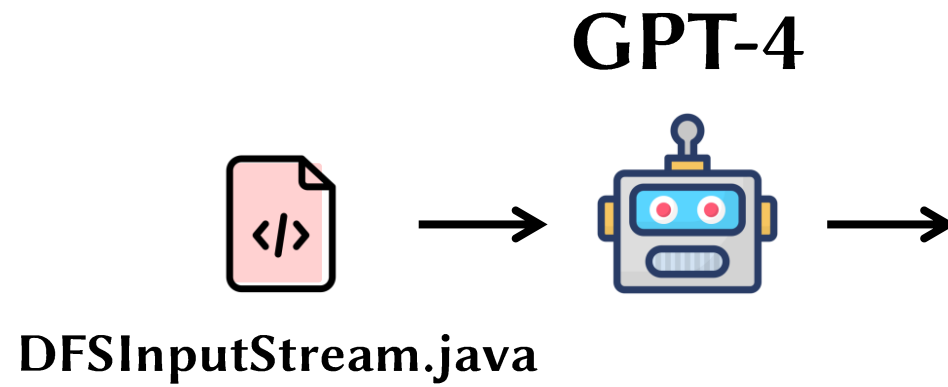
– If YES provide the name of the “retry” method

GPT-4

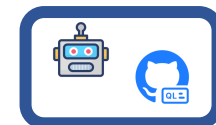


<re

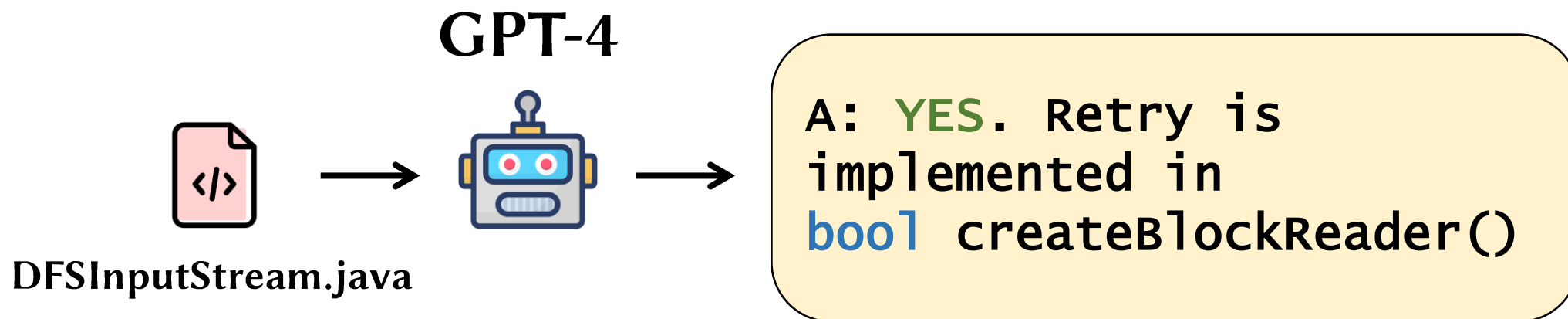
Identifying retry logic



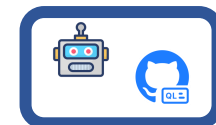
Identifying retry logic



Retry discovery



Finding retry locations & exceptions



Retry discovery

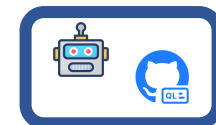
HDFS/DFSInputStream.java

```
bool createBlockReader():  
1. while (true) {  
2.   try {  
3.     block = refreshBlock(block);  
4.     dnInfo = getDNInfoFor(block);  
5.     if (dnInfo == null)  
6.       break;  
7.   }  
8.   catch (IOException e) {  
9.     LOG("Failed to connect to "+  
        dnInfo.addr + "Retried " +  
        ++retryCount + " times");  
10.    addToDeadNodes(dnInfo.info);  
11.  }  
12. }
```



CodeQL

Finding retry locations & exceptions



Retry discovery

HDFS/DFSInputStream.java

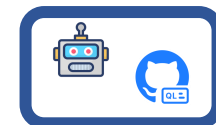
```
bool createBlockReader():  
1. while (true) {  
2.   try {  
3.     block = refreshBlock(block);  
4.     dnInfo = getDNInfoFor(block);  
5.     if (dnInfo == null)  
6.       break;  
7.   }  
8.   catch (IOException e) {  
9.     LOG("Failed to connect to "+  
        dnInfo.addr + "Retried " +  
        ++retryCount + " times");  
10.    addToDeadNodes(dnInfo.info);  
11.  }  
12. }
```

```
BlockStruct refreshBlock(...)  
    throws IOException;  
DnInfoStruct getDNInfoFor(...)  
    throws IOException;
```



CodeQL

Finding retry locations & exceptions



Retry discovery

HDFS/DFSInputStream.java

```
bool createBlockReader():  
1. while (true) {  
2.   try {  
3.     block = refreshBlock(block);  
4.     dnInfo = getDNInfoFor(block);  
5.     if (dnInfo == null)  
6.       break;  
7.   }  
8.   catch (IOException e) {  
9.     LOG("Failed to connect to "+  
        dnInfo.addr + "Retried " +  
        ++retryCount + " times");  
10.    addToDeadNodes(dnInfo.info);  
11.  }  
12. }
```

```
BlockStruct refreshBlock(...)  
    throws IOException;  
DnInfoStruct getDNInfoFor(...)  
    throws IOException;
```



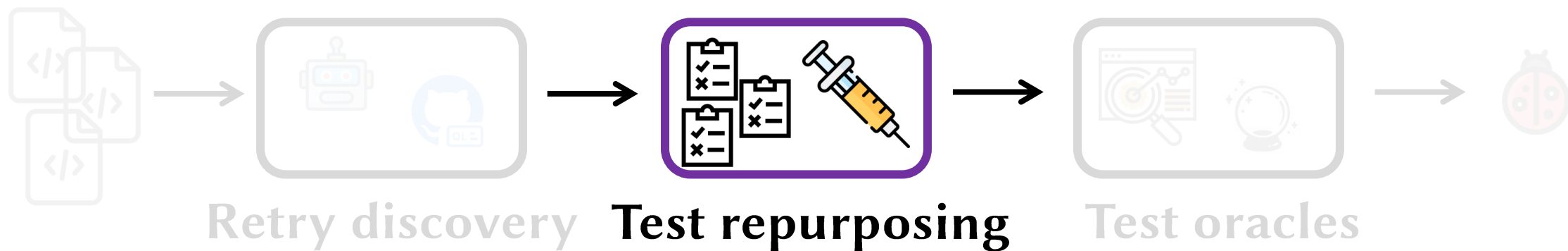
CodeQL



<DFSInputStream.java:3, IOException>

<DFSInputStream.java:4, IOException >

WASABI, a toolkit for finding retry bugs



Dynamic testing

Selecting which tests to run



Test repurposing



<DFSInputStream.java:3, IOException>



<DFSInputStream.java:4, IOException >



< ... , ... >

..



Selecting which tests to run



Test repurposing



⟨DFSInputStream.java:3, IOException⟩



Run test suite once, to collect coverage information



...



Selecting which tests to run



Test repurposing



`<DFSInputStream.java:3, IOException>`



Run test suite once, to collect coverage information



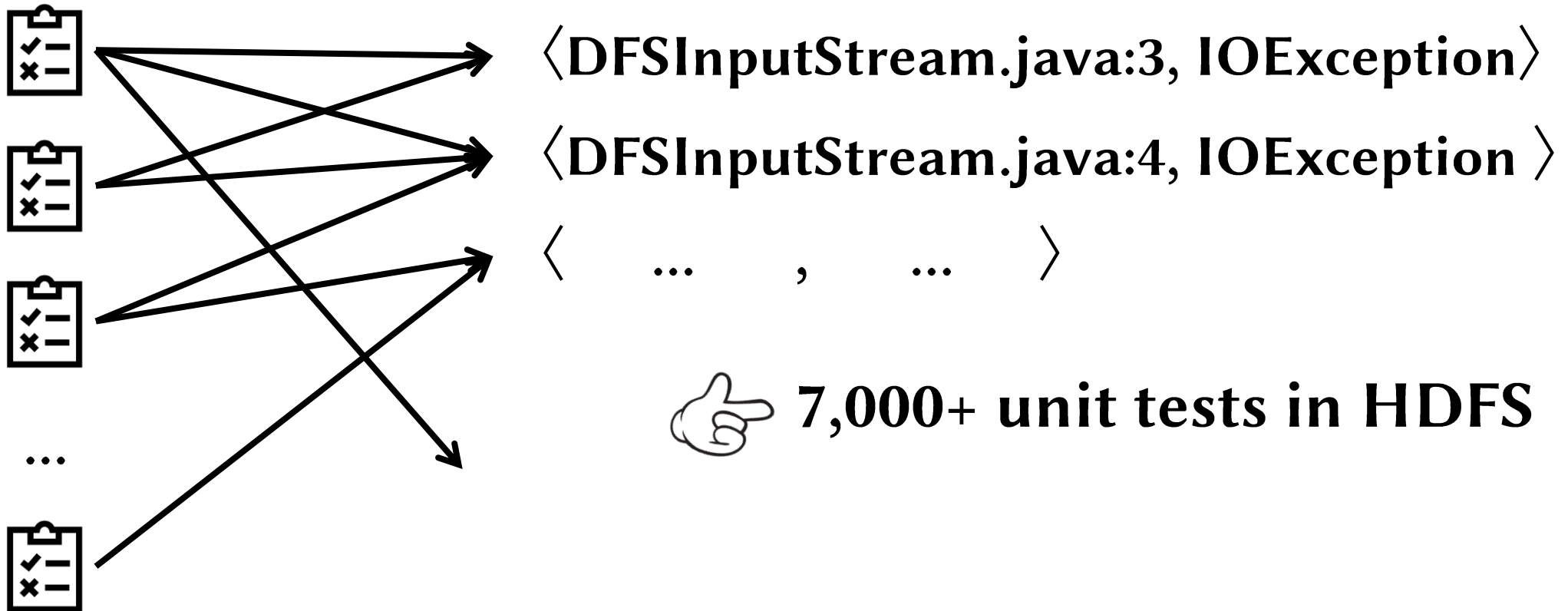
Use coverage info to match tests with retry locations



Selecting which tests to run



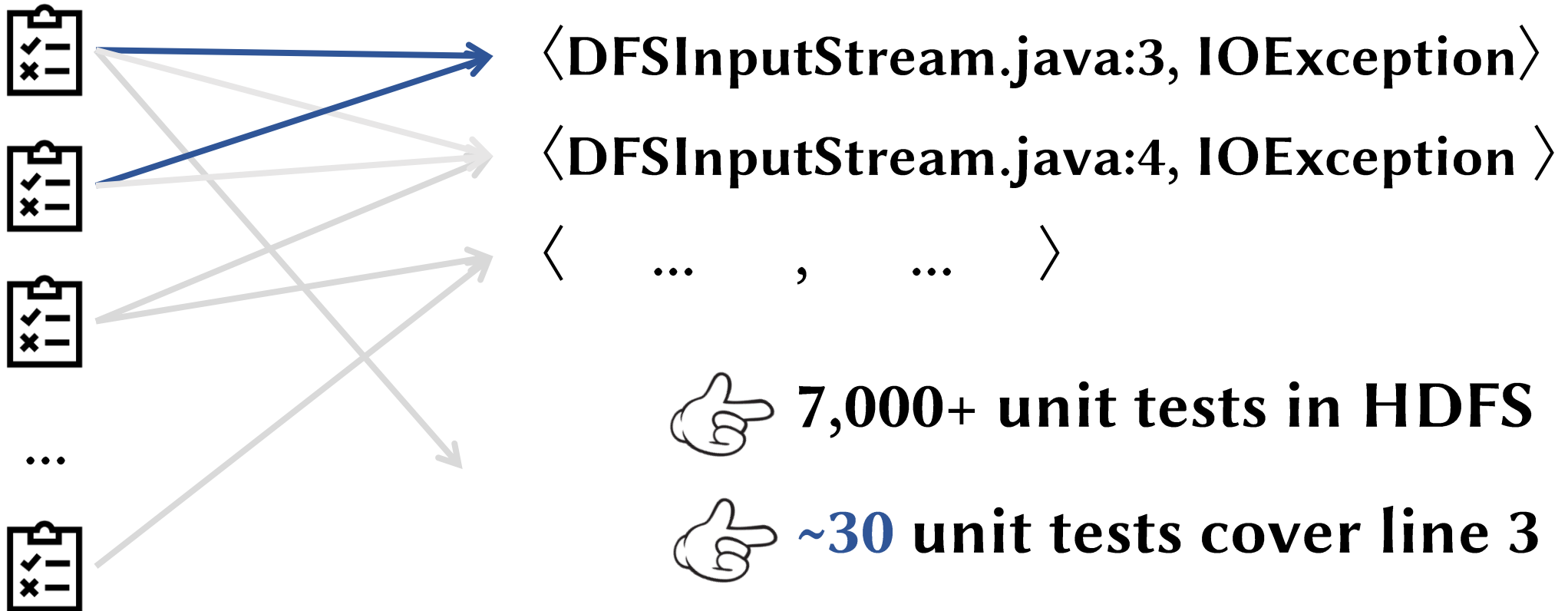
Test repurposing



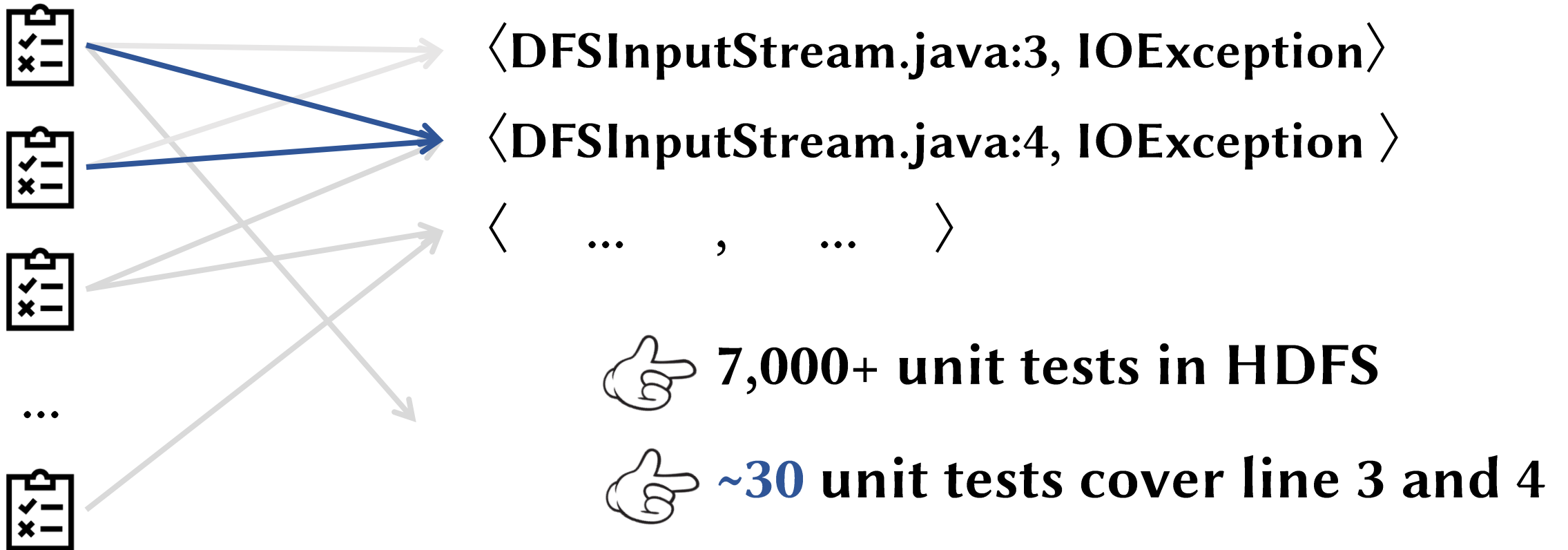
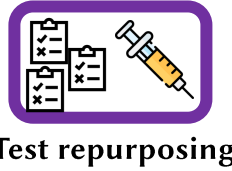
Selecting which tests to run



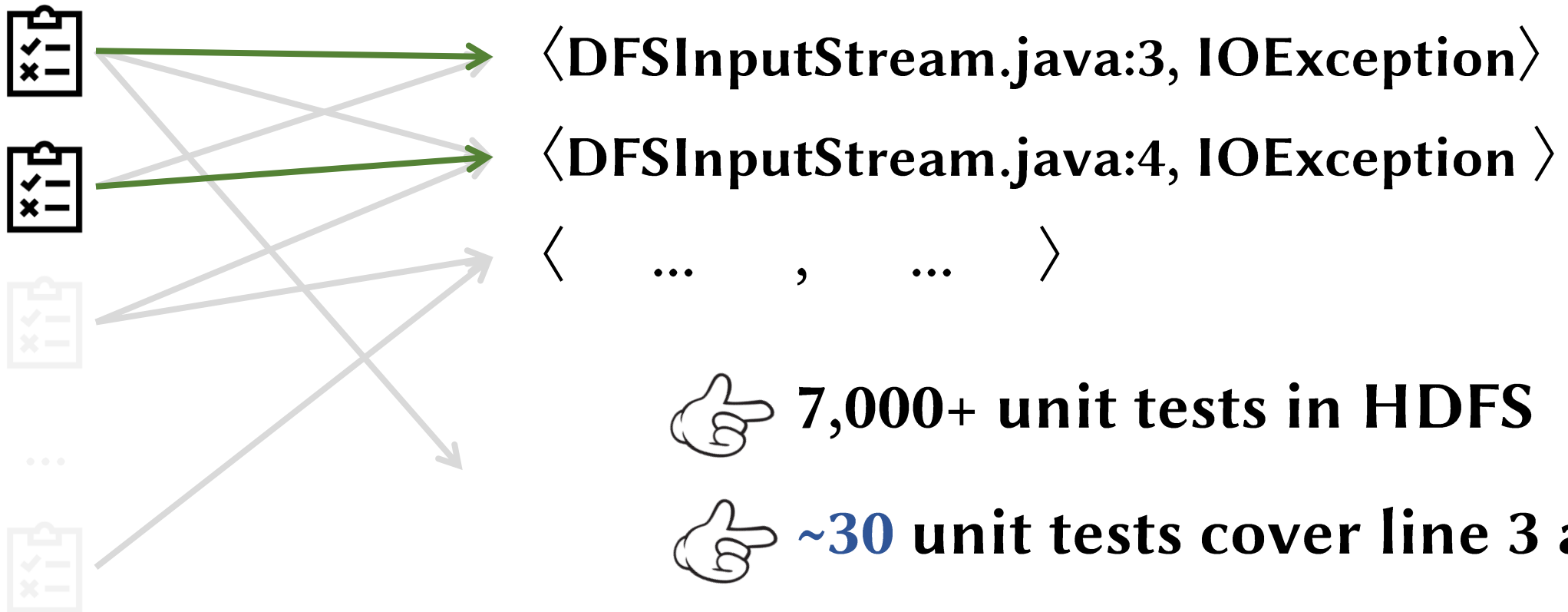
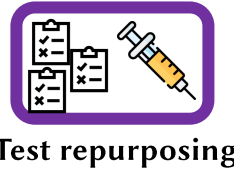
Test repurposing



Selecting which tests to run



Selecting which tests to run

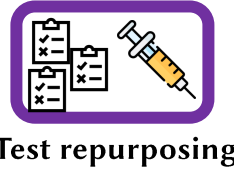


👉 **7,000+ unit tests in HDFS**

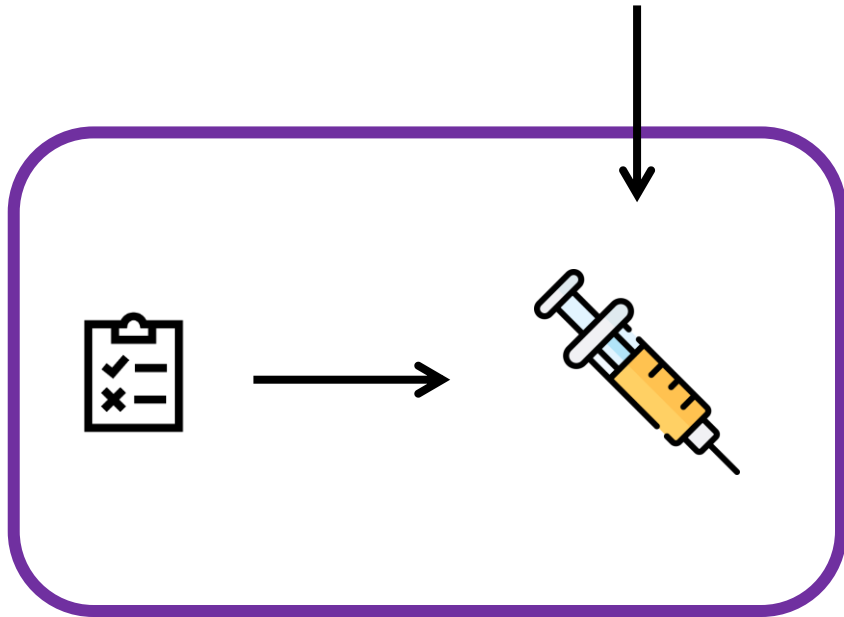
👉 **~30 unit tests cover line 3 and 4**

👉 **2 needed to test for retry bugs**

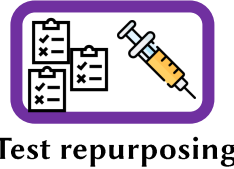
Repurposing tests to find retry bugs



⟨HDFSInputStream.java:3, IOException⟩



Injecting faults during test runs



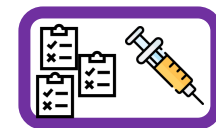
HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block); ←
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "
10.        + dnInfo.addr);
11.   }
12. }
```

Intercept using AspectJ &
inject **IOException**

Injecting faults during test runs



Test repurposing

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "
10.        + dnInfo.addr);
11.     addToDeadNodes(dnInfo.info);
12.   }
```

Jump to **catch block**

A black curved arrow originates from the line 'block = refreshBlock(block);' and points to the 'catch (IOException e) {' line, indicating a jump to the catch block.

Injecting faults during test runs



Test repurposing

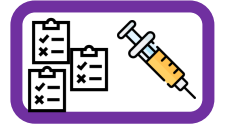
HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "
10.        + dnInfo.addr);
11.     addToDeadNodes(dnInfo.info);
12.   }
```

Renter the **while** loop

Injecting faults during test runs



Test repurposing

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {  
2.   try {  
3.     block = refreshBlock(block);  
4.     dnInfo = getDNInfoFor(block);
```

How do we know a retry bug happened?

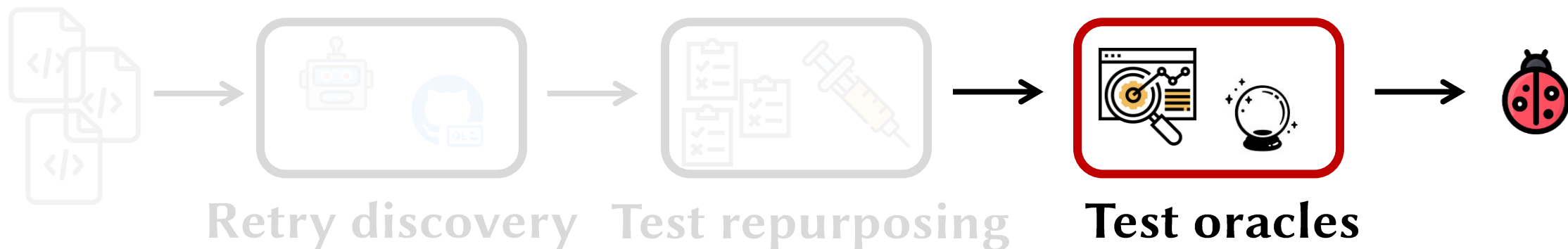
```
7.   }  
8.   catch (IOException e) {  
9.     LOG("Failed to connect to "  
        + dnInfo.addr);  
10.    addToDeadNodes(dnInfo.info);  
11.  }  
12. }
```

WASABI, a toolkit for finding retry bugs



Dynamic testing

WASABI, a toolkit for finding retry bugs



Dynamic testing

- 👉 Crashes with different exception (**HOW**)
- 👉 Large # of retry attempts or prolonged retry (**WHEN**)
- 👉 Not “pausing” between retry attempts (**WHEN**)

Determining if a retry bug occurred



Test oracles

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "
          + dnInfo.addr);
10.    addToDeadNodes(dnInfo.info);
11.  }
12. }
```

Inject **IOException**

A black curved arrow pointing from the text 'Inject IOException' to the catch block in the code snippet.

Determining if a retry bug occurred



Test oracles

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "
10.        + dnInfo.addr);
11.   }
12. }
```

NullPointerException



Determining if a retry bug occurred



Test oracles

HDFS/DFSInputStream.java

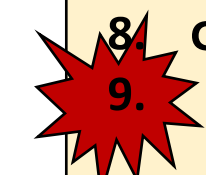
```
bool createBlockReader():
```

```
1. while (true) {
2.   try {
3.     block = refreshBlock(block);
4.     dnInfo = getDNInfoFor(block);
5.     if (dnInfo == null)
6.       break;
7.   }
8.   catch (IOException e) {
9.     LOG("Failed to connect to "
10.        + dnInfo.addr);
11.     addToDeadNodes(dnInfo.info);
12.   }
}
```

IOException

≠

NullPointerException



Determining if a retry bug occurred



Test oracles

HDFS/DFSInputStream.java

```
bool createBlockReader():
```

```
1. while (true) {  
2. try {  
3.   block = refreshBlock(block); ←
```

IOException

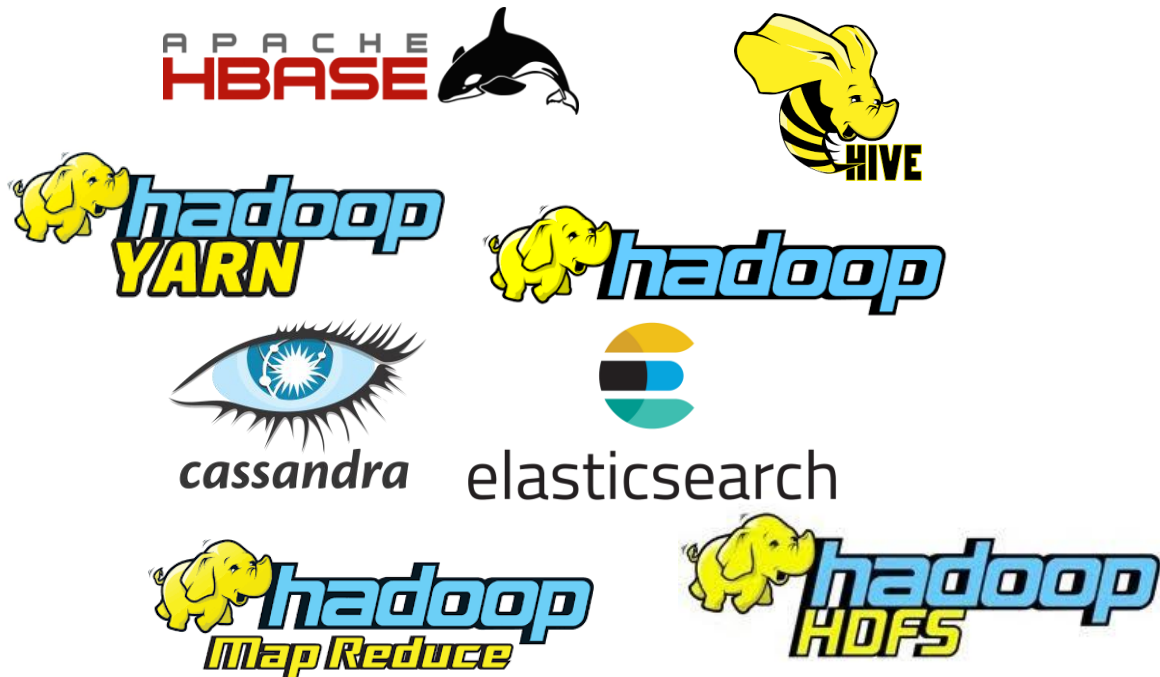
New bug, HDFS-17590 found by WASABI

```
7. }  
8. catch (IOException e) {  
9.   LOG("Failed to connect to "  
        + dnInfo.addr); ←  
10.  addToDeadNodes(dnInfo.info);  
11. }  
12. }
```

NullPointerException

Benchmarks and key results

8 open-source systems



Benchmarks and key results

8 open-source systems



109 new bug found



cassandra



elasticsearch



Benchmarks and key results

8 open-source systems

👉 109 new bug found

👉 largely <12h per system



cassandra



elasticsearch



Benchmarks and key results

8 open-source systems

👉 109 new bug found

👉 largely <12h per system

👉 \$8 median GPT cost per system



Benchmarks and key results

8 open-source systems



109 new bug found



largely **<12h** per system



\$8 median GPT cost per system



for **2** bugs found, **1** false positive



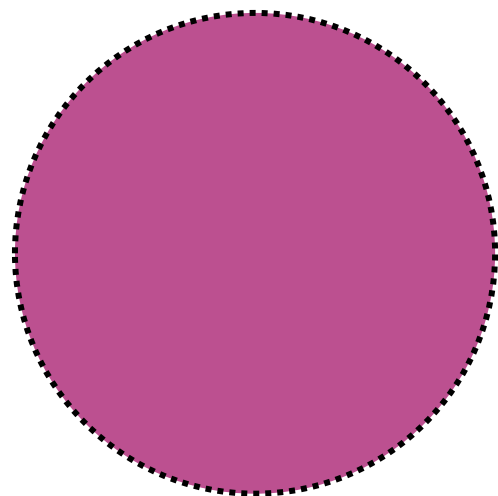
cassandra

elasticsearch



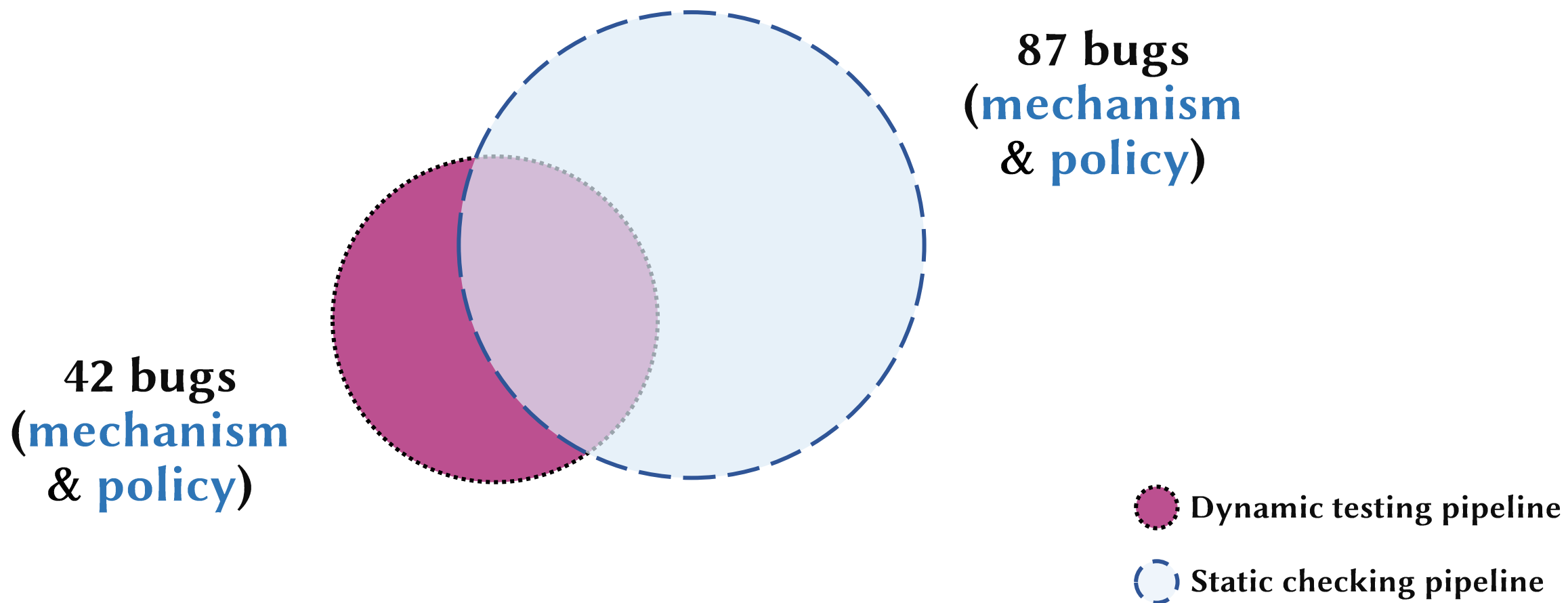
New retry bugs found by WASABI

42 bugs
(mechanism
& policy)

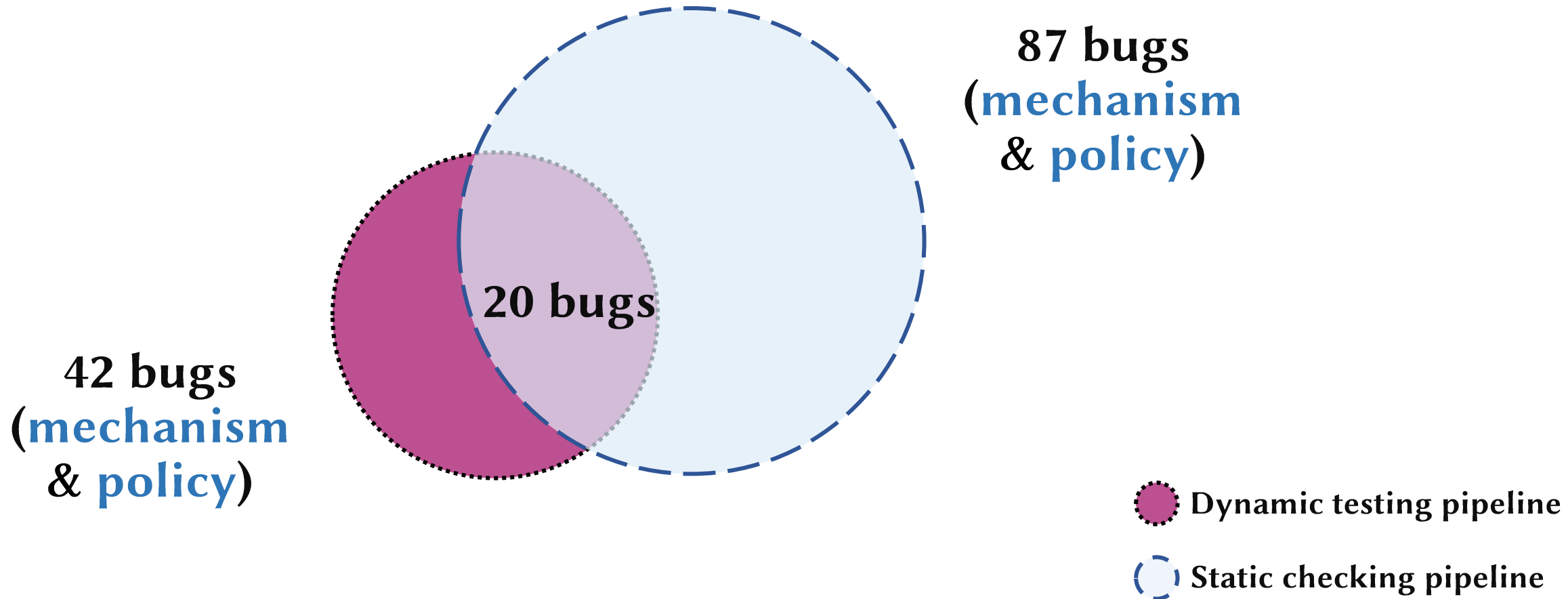


 Dynamic testing pipeline

New retry bugs found by WASABI



New retry bugs found by WASABI



Key takeaways

Key takeaways



Resilience mechanisms, crucial in large-scale systems

Key takeaways



Resilience mechanisms, crucial in large-scale systems



Bug-finding at mechanism level is needed

Key takeaways



Resilience mechanisms, crucial in large-scale systems



Bug-finding at mechanism level is needed



LLMs + traditional program analysis =



Thank you!