CFix Automated Concurrency-Bug Fixing

Guoliang Jin, Wei Zhang, Dongdong Deng, Ben Liblit, and Shan Lu.

University of Wisconsin–Madison

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Bugs Need to be Fixed

- Buggy software is an unfortunate fact;
  - There are many bugs to be fixed.
- Software quality does not improve until bugs are fixed;
  - Human patches are often buggy.



Automated fixing is desired

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Automated Fixing is Difficult

- **What** is the correct behavior?
  - Usually requires developers' knowledge.
- **How** to get the correct behavior?
  - Change program states under bug-triggering inputs;
  - No change to program states under other inputs.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Concurrency Bugs

- Synchronization mistakes in multi-threaded programs.
- They are severe:
  - Caused real-world problems: 

IPO, etc;
  - Increasingly an issue in this multi-core era.
- They are difficult to reason about and fix manually:
  - Concurrency-bug fixing is time consuming[LuASPLOS'08];
  - Concurrency-bug patches are the most error-prone[YinFSE'11].

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Concurrency Bugs are Easier to Fix Automatically

- **What** is the correct behavior?
  - The program state is already correct
    as long as the buggy interleaving does not occur.

- **How** to get the correct behavior?
  - Only need to disable failure-inducing interleavings;
  - Can leverage well-defined synchronization operations.

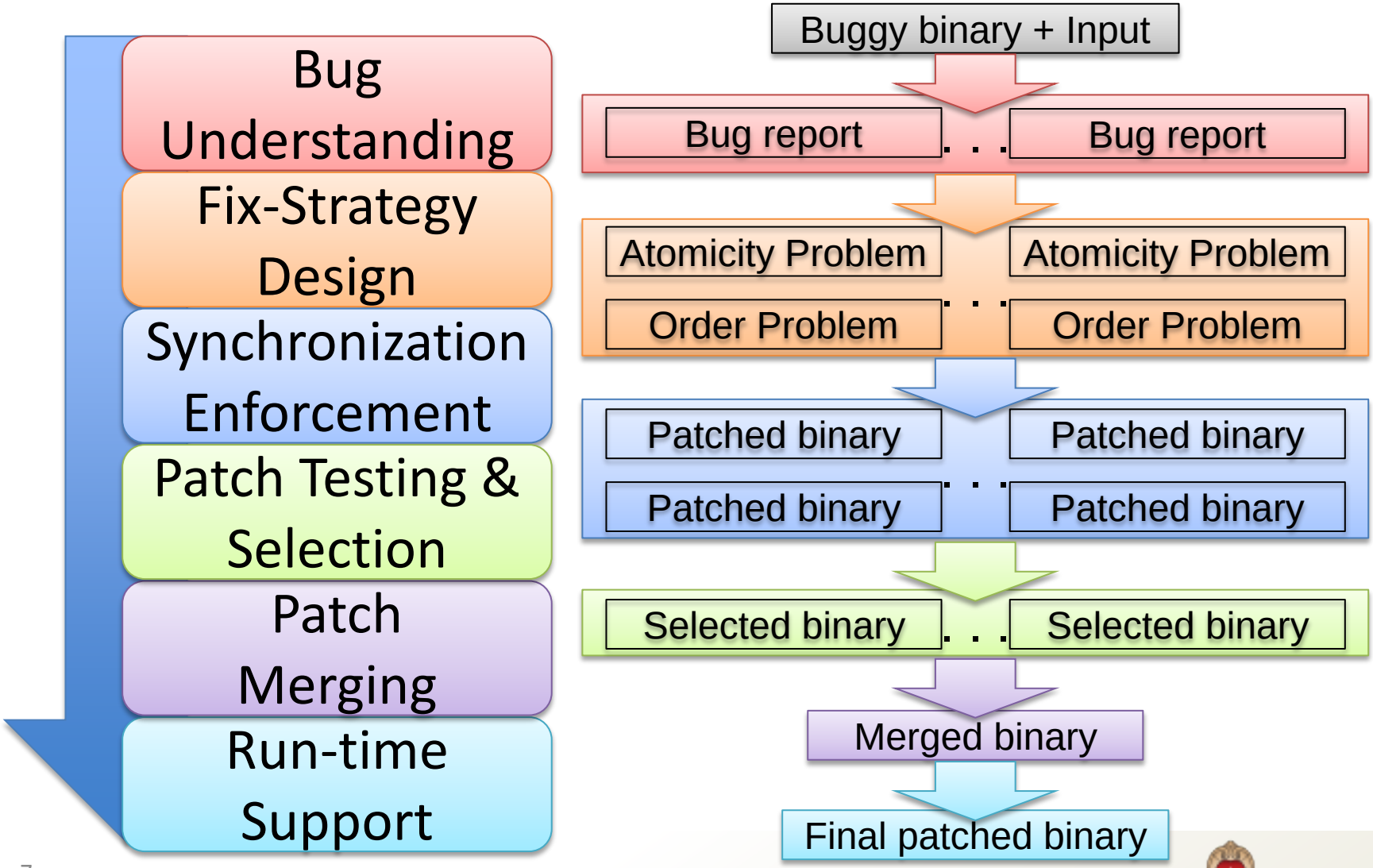
WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Challenges

- How to handle the huge variety of bugs?
- How to know the root cause of a bug?
- How to ensure correctness?
- How to maintain performance?
- How to achieve patch simplicity?
- How to effectively test the patches?

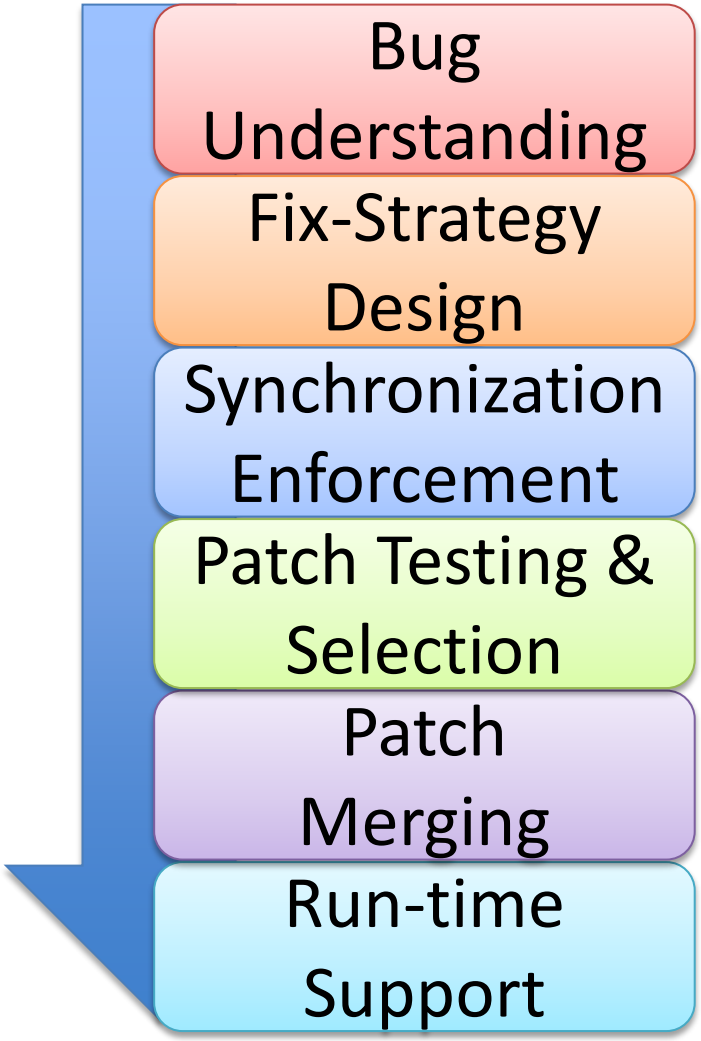
WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

CFix Overview



WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

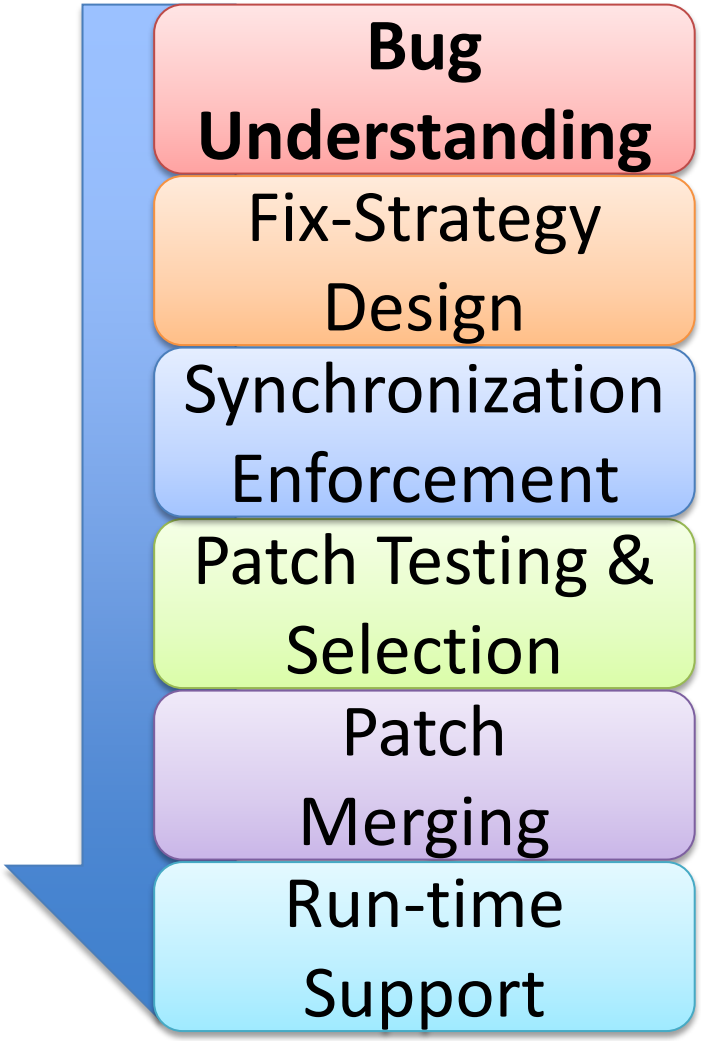
Contributions



- A technique that enforces order relationship: OFix;
- A framework that assembles a set of techniques to automate the whole bug-fixing process: CFix.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

CFix: Automated Concurrency-Bug Fixing



Challenge:

- Concurrency bugs are difficult to reason about.

Solution:

- Leverage existing bug detectors.

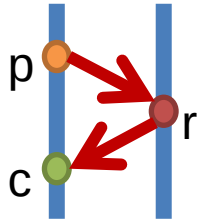
WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Bug Understanding

- Use four types of different detectors:

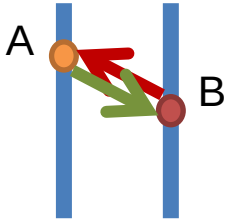
atomicity violation detectors

ParkASPLOS'09, FlanaganPOPL'04, LuASPLOS'06, ChewEuroSys'10



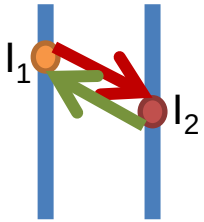
order violation detectors

ZhangASPLOS'10, LuciaMICRO'09, YuISCA'09, GaoASPLOS'11



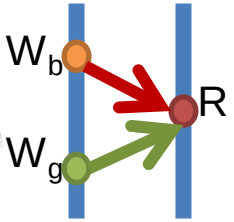
data race detectors

SenPLDI'08, SavageTOCS'97, YuSOSP'05, EricksonOSDI'10, KasikciASPLOS'10



abnormal data flow detectors

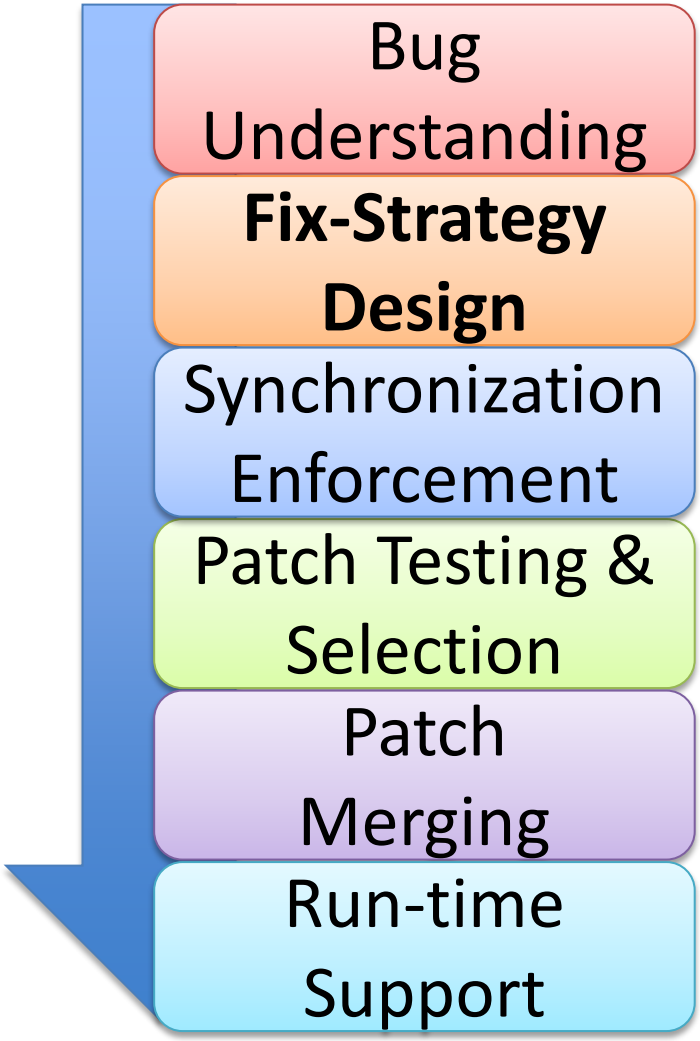
ZhangASPLOS'11, ShiOOPSLA'10



Bug report could be imperfect.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

CFix: Automated Concurrency-Bug Fixing



Challenges:

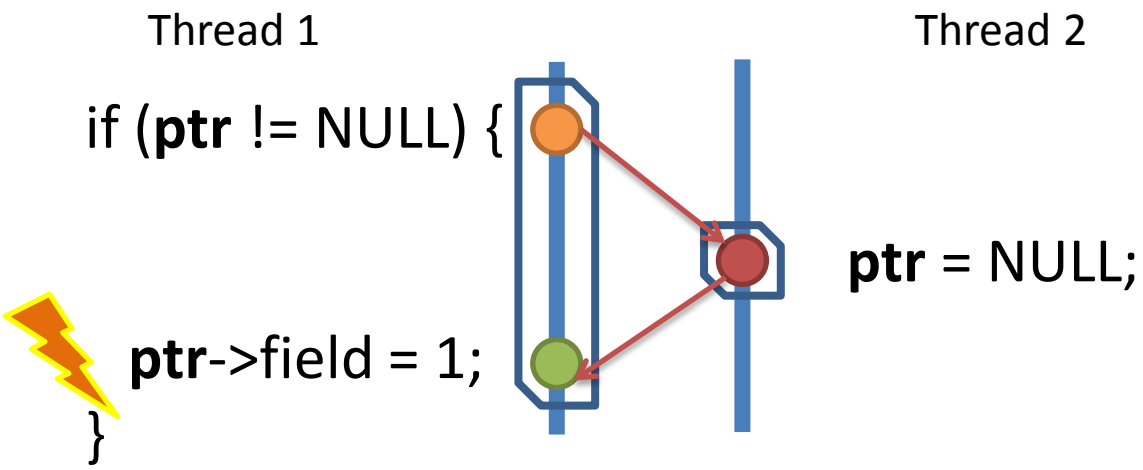
- Huge variety of bugs;
- Inaccurate root cause.

Solution:

- Design a set of fix strategies using a combination of mutual exclusion enforcement; order relationship enforcement.

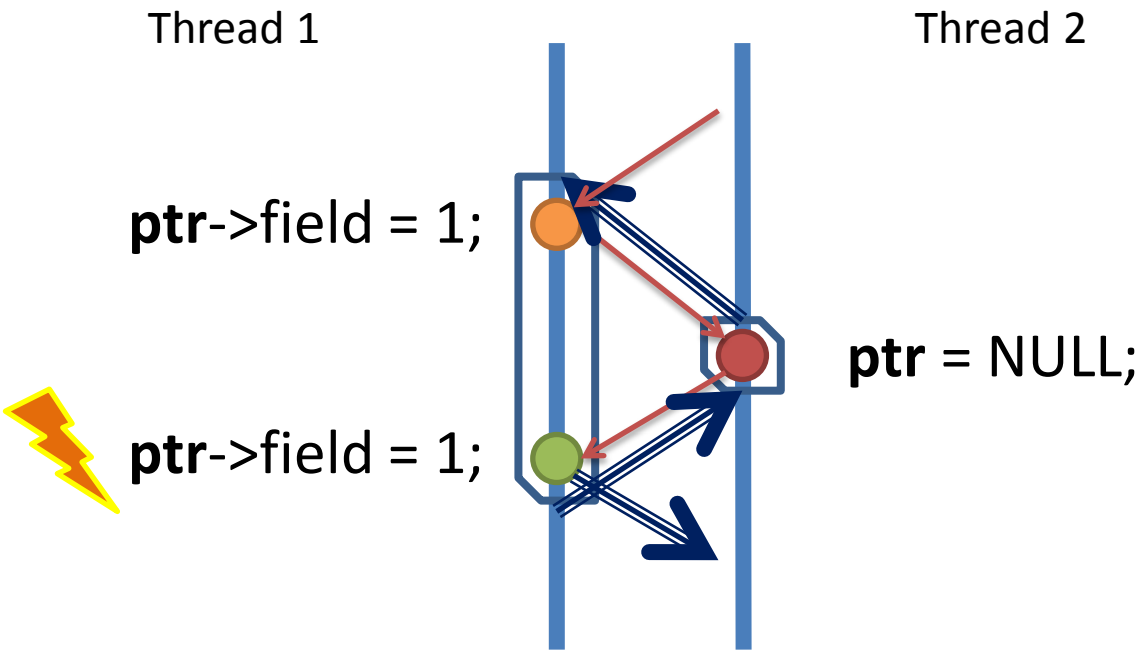
WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Fix-Strategy for Atomicity-Violation Detectors
Example 1

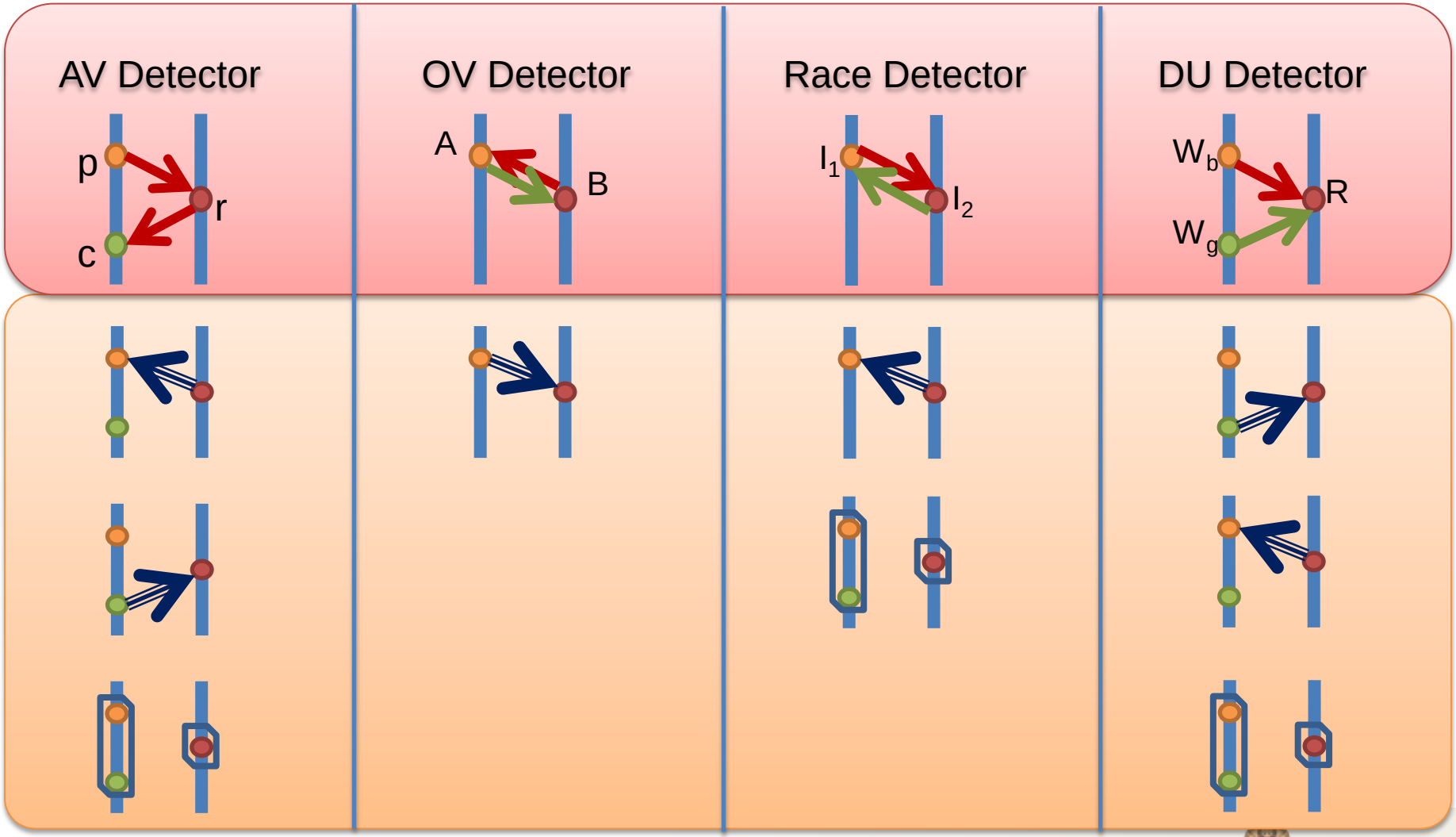


WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Fix-Strategy for Atomicity-Violation Detectors Example 2

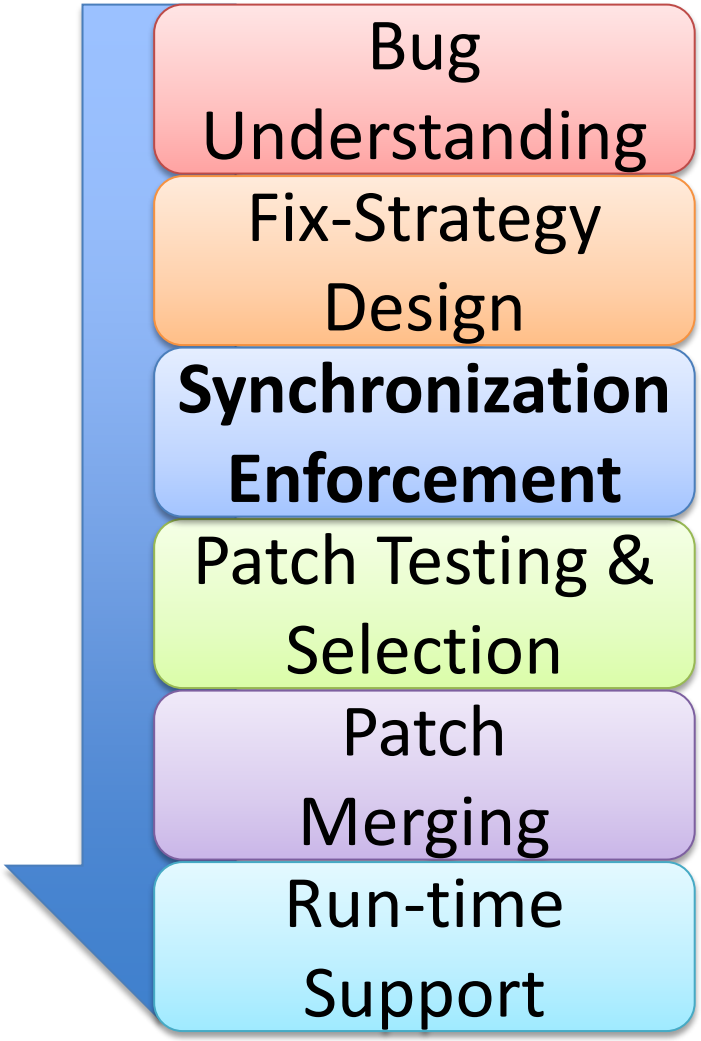


Fix-Strategies



WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

CFix: Automated Concurrency-Bug Fixing



Challenges:

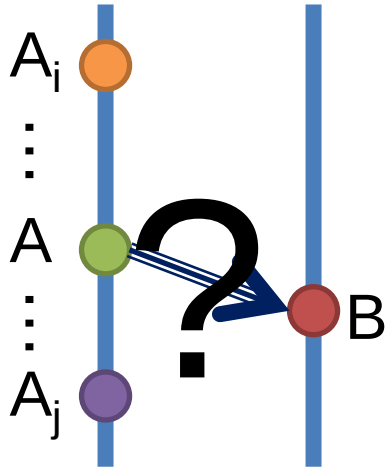
- Correctness, performance, simplicity.

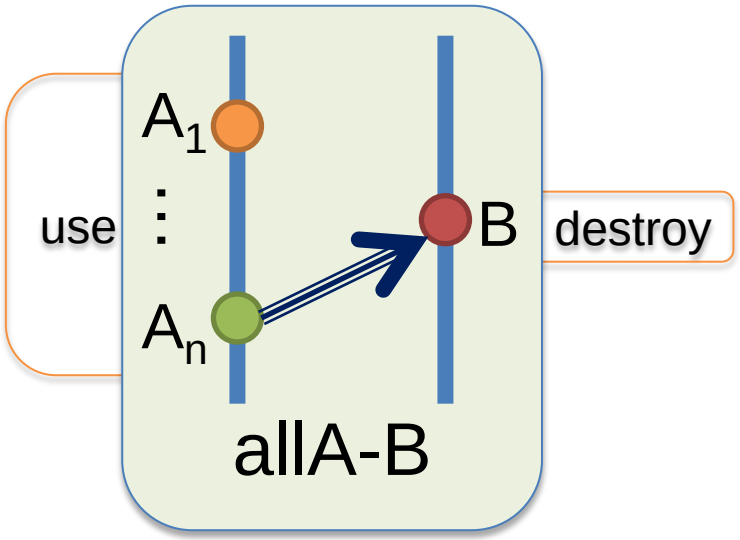
Solution:

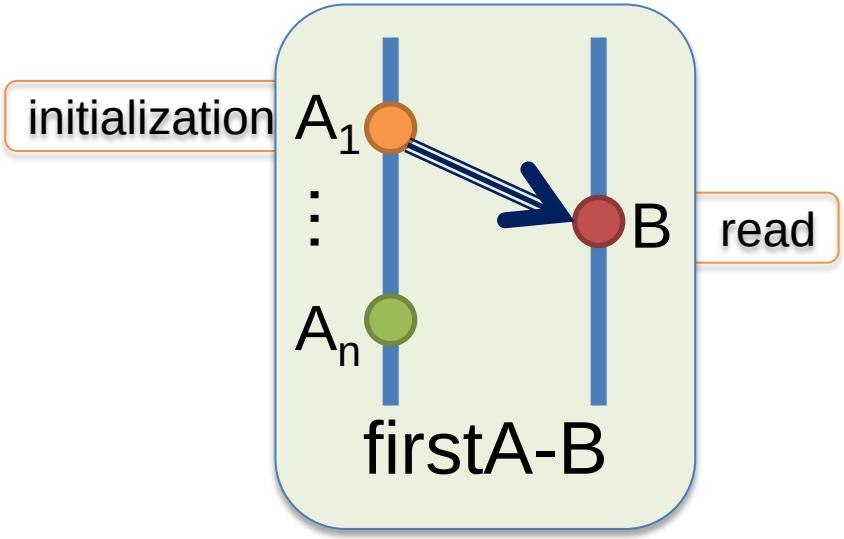
- Mutual exclusion enforcement: AFix [JinPLDI'11]
- Order relationship enforcement: OFix.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Order Relationship Enforcement: OFix







WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

OFix allA-B Enforcement: Principles

- *signal* in A-threads:
  - A-thread signals when it will not execute more A;
  - Each A-thread signals only once;
  - Each A-thread signals as soon as possible.

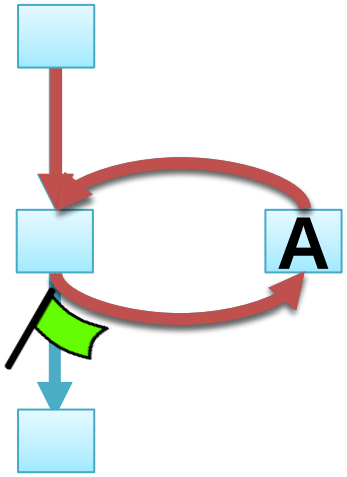
- *wait* before B:
  - B Proceeds when each A-thread has signaled.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

OFix allA-B Enforcement: A side

How to identify the last A instance in one thread

```
. . .;
for (. . .)
    . . . ; // A
. . .;
```



- Each thread that execute A:
  -  exactly once as soon as it can execute no more A.

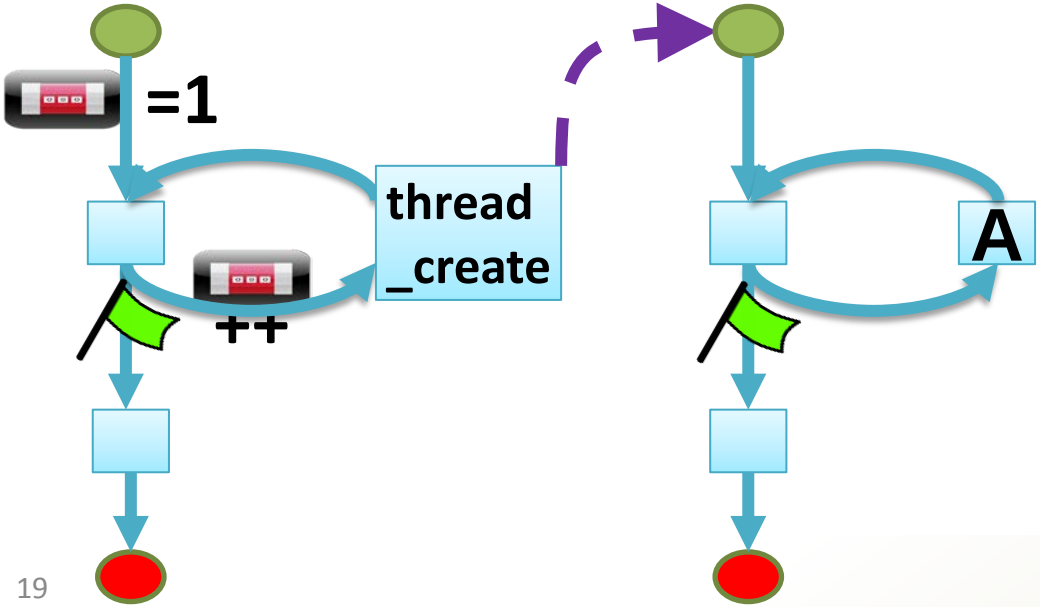
WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

OFix allA-B Enforcement: A side

How to identify the last thread that executes A

```
void main() {
  for (. . .)
    thread_create(thr_main);
  . . .;
}
```

```
void thr_main() {
  for (. . .)
    . . . ; // A
  . . .;
}
```

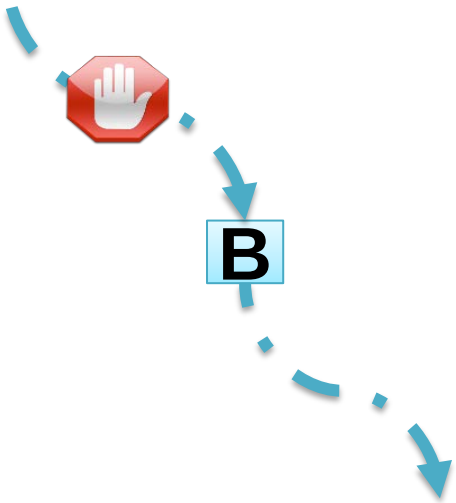


counter for signal threads

```
void ofix_signal() {
  mutex_lock(L);
  --;
  if ( == 0)
    cond_broadcast(con);
  mutex_unlock(L);
}
```

OFix allA-B Enforcement: B side

- Safe to execute only when  is 0.




```
void ofix_wait() {
  mutex_lock(L);
  if (  != 0)
    cond_timedwait(con, L, t);
  mutex_unlock(L);
}
```



- Give up if OFix knows that it introduces new deadlock.
- Timed wait-operation to mask potential deadlocks.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

OFix allA-B Enforcement: Simplification

-  is not needed if

  B does not execute in the same run as .

```
void main() {
    if (. . .) {
        OFixSignal;
        exit(1);
    }
    thread_create(thr_main, . . .); // A
    . . .; // B
}
```

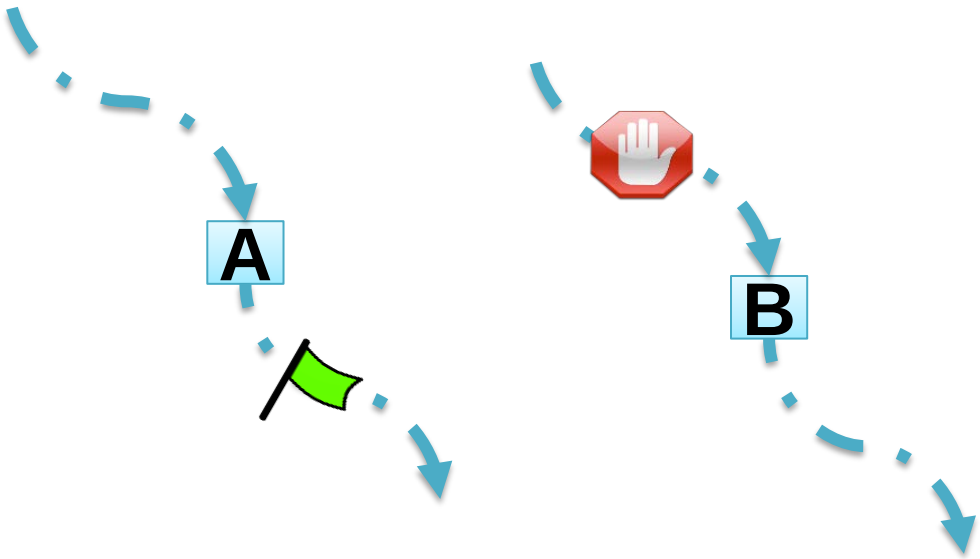


- Significantly improve patch simplicity.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

OFix firstA-B

- Basic enforcement

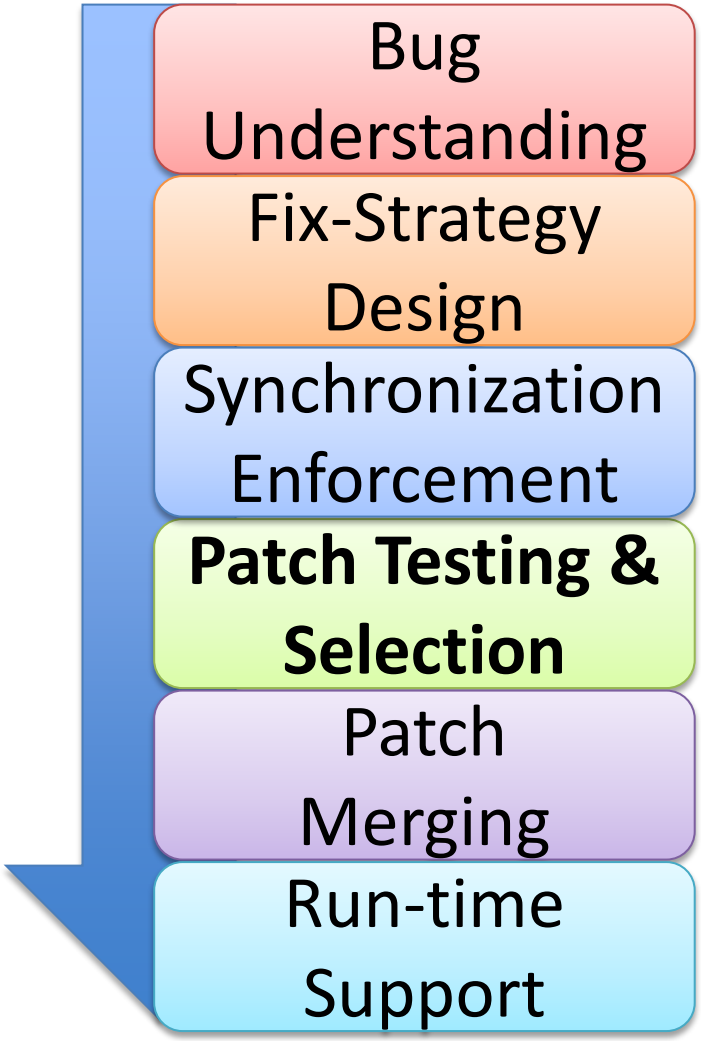


- When A may not execute:
  - Add a safety-net of signal with allA-B algorithm.

```
void ofix_signal_b() {
  if (flag != true) {
    flag = true;
    mutex_lock(L);
    cond_broadcast(con);
    mutex_unlock(L);
  }
}
```

```
void ofix_wait_b() {
  mutex_lock(L);
  if (flag != true)
    cond_timedwait(con, L, t);
  mutex_unlock(L);
}
```

CFix: Automated Concurrency-Bug Fixing



Challenge:

- Effective multi-thread software testing

Solution:

- CFix-patch oriented testing

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Patch Testing Principles

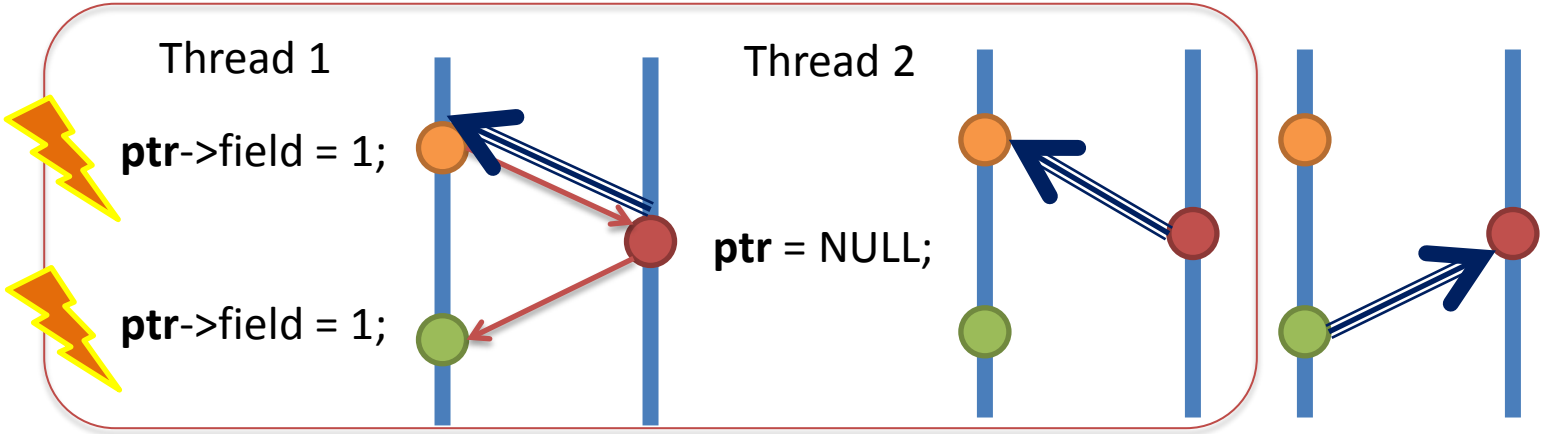
- Prune incorrect patches:
  - Patches causing failures due to incorrect root causes, etc.
- Prune slow patches
- Prune complicated patches

- Not exhaustive testing, but patch oriented testing.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Run Once without External Perturbation

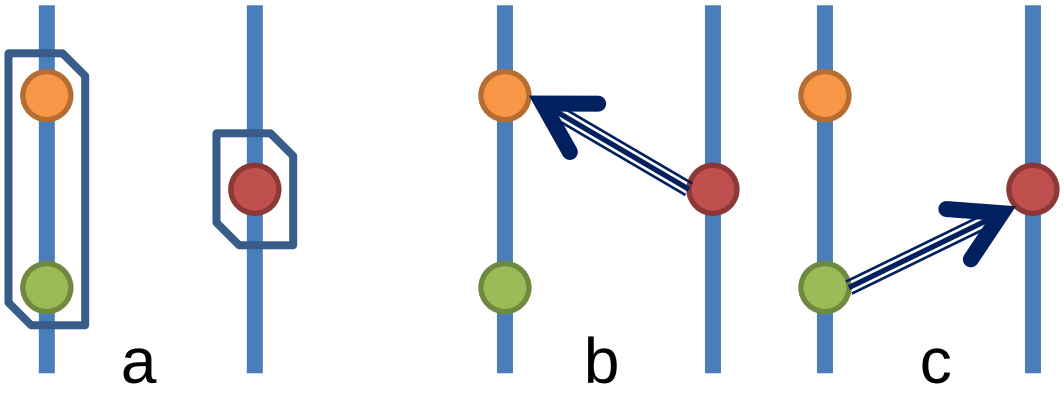
- Reject if there is a time-out or failure.
- Patches fixing wrong root cause:
  - Make software to fail deterministically.



WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

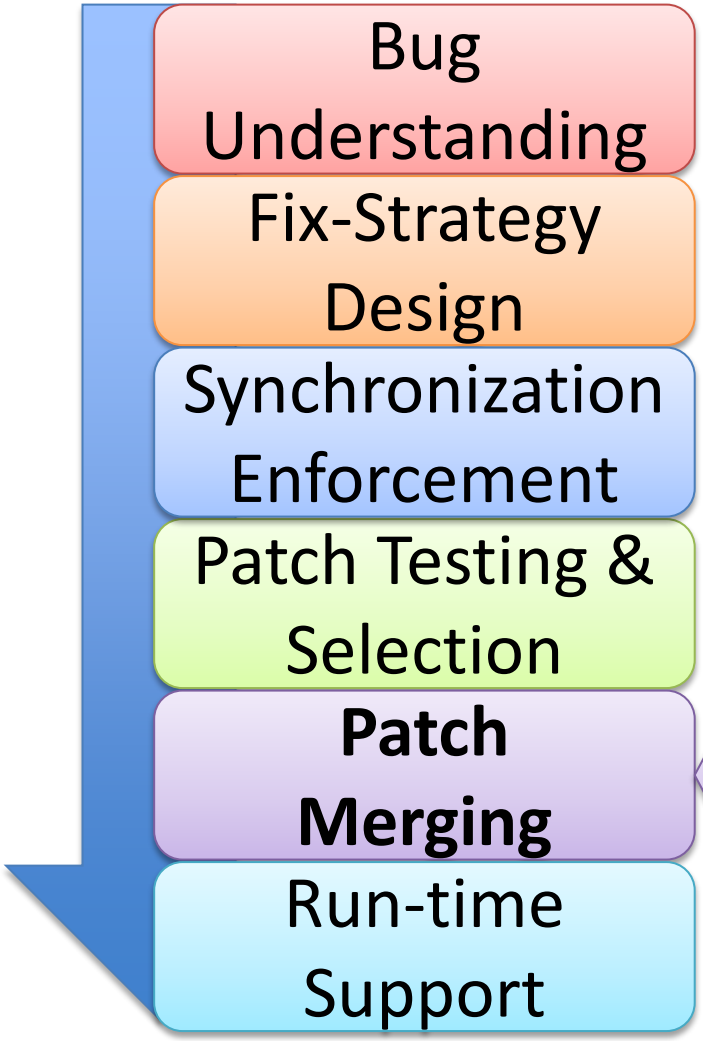
Implicit Bad Patch

- A failure in patch_b implies a failure in patch_a:
  - If patch_a is less restrictive than patch_b.



- Helpful to prune patch_a:
  - Traditional testing may not find the failure in patch_a.

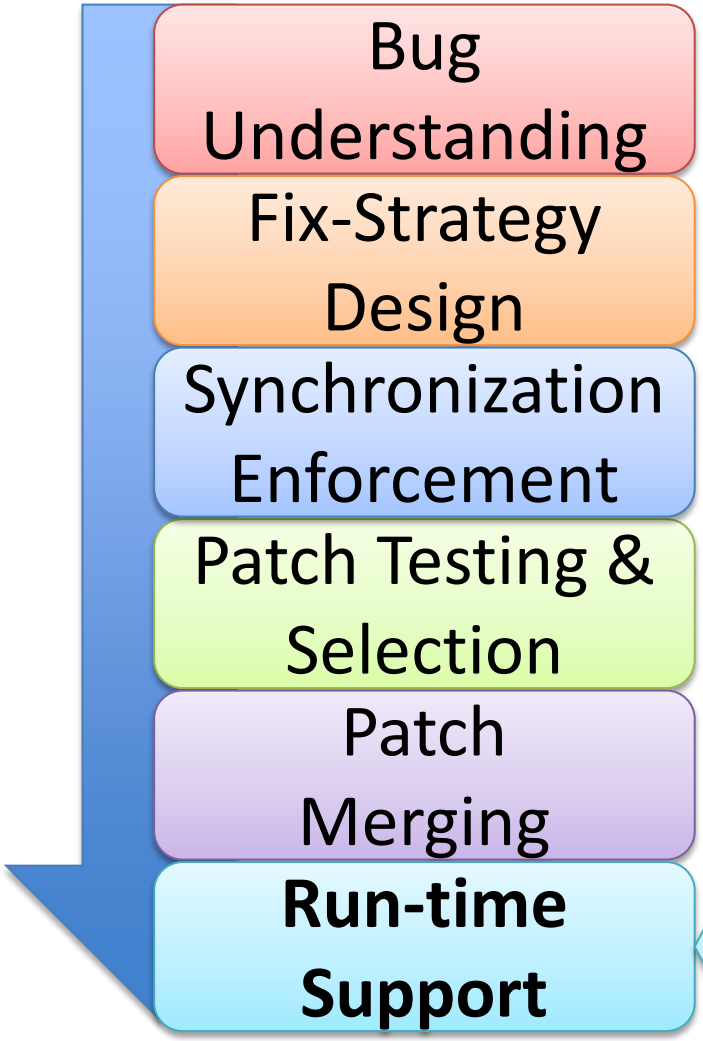
CFix: Automated Concurrency-Bug Fixing



- One programming mistake usually leads to multiple bugs.
- Heuristics to merge patches for related bugs.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

CFix: Automated Concurrency-Bug Fixing



- To understand whether there is a deadlock underlying time-out.
- Low-overhead, and suitable for production runs.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Methodology

APP.	AV Detector	OV Detector	RA Detector	DU Detector
PBZIP2				
x264				
FFT				
HTTrack				
Mozilla-1				
transmission				
ZSNES				
Apache				
MySQL-1				
MySQL-2				
Mozilla-2				
Cherokee				
Mozilla-3				

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Evaluation

APP.	AV Detector	OV Detector	RA Detector	DU Detector	# of Ops
PBZIP2	✓	✓	✓		5
x264		✓			7
FFT	✓	✓	✓	✓	5
HTTrack	✓	✓	✓		2
Mozilla-1	✓		✓		2
transmission	✓	✓	✓	✓	2
ZSNES		✓			3
Apache	✓		✓		3
MySQL-1	✓		✓	✓	5
MySQL-2	✓		✓	×	9
Mozilla-2	✓		✓		3
Cherokee	✓		✓	✓	2
Mozilla-3	✓		✓	✓	5

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

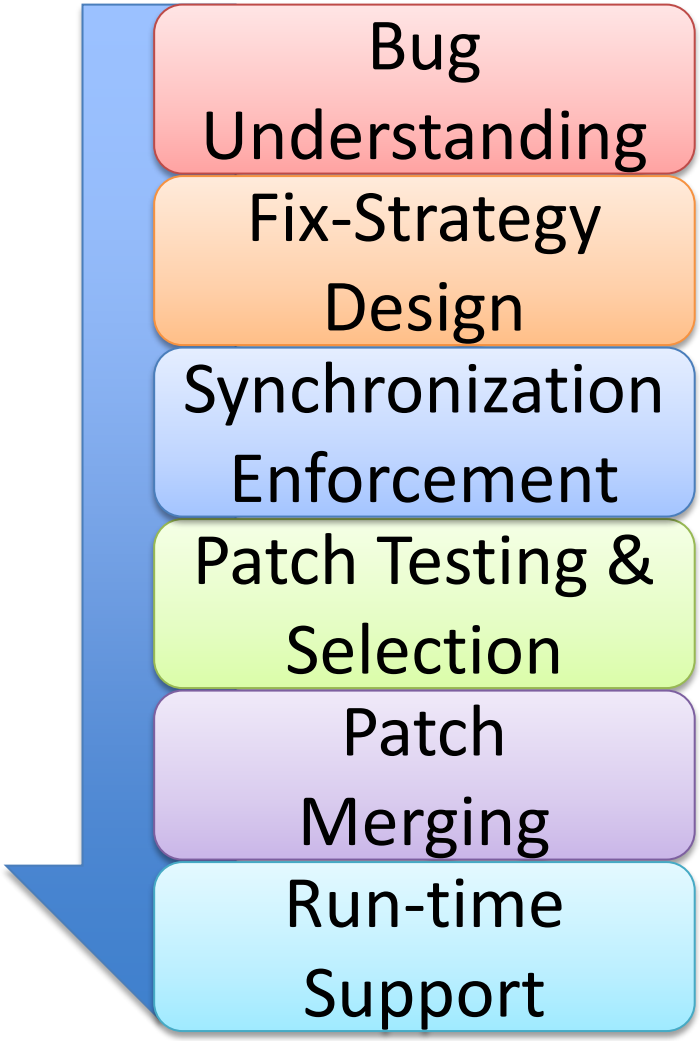
Conclusion

- CFix uses some heuristics, with good results in practice:
  - Fixing root cause without requiring detectors to report it;
  - Small overhead;
  - Good simplicity.

- Concurrency bugs are feasible to be fixed automatically:
  - By removing bad interleavings;
  - Must be careful in the details.

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Questions on CFix?



It's 2012.
Is this the end of my world?



WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON