

X-ray: Automating Root-Cause Diagnosis of Performance Anomalies in Production Software

Mona Attariyan

Michael Chow

Jason Flinn

Software is complex

- Troubleshooting complex systems is hard
- Performance issues are especially difficult



What tools are available?

- Developers
 - Profilers
 - Tracing tools
 - Logging
- Admins & Users
 - Look on the Internet
 - Stare at config file
 - Ask for support



Goal

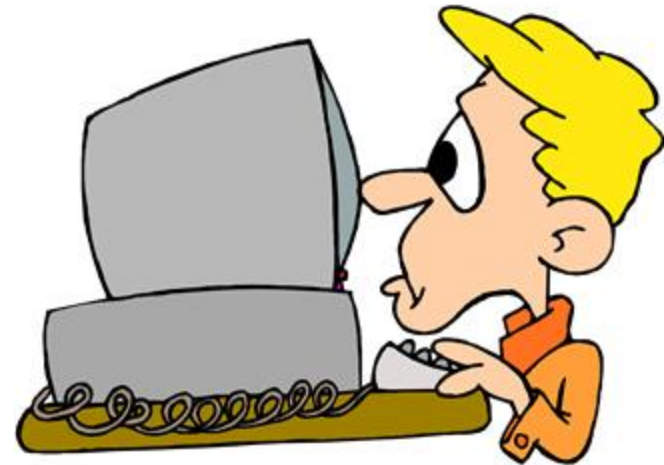
- Help diagnose performance issues without:
 - Source code
 - Error or log messages
 - Controlled workloads
 - Developer support



Example

- Postfix mail server user spends hours troubleshooting performance issue

```
top  
iotop  
wireshark  
/var/log/messages  
/etc/main.cf
```



Complex configuration files

```
queue_directory = /var/spool/postfix          ...
command_directory = /usr/sbin
daemon_directory = /usr/libexec/postfix
data_directory = /var/lib/postfix
mail_owner = postfix
default_privs = nobody
myhostname = host.domain.tld
mydomain = domain.tld
myorigin = $myhostname
inet_interfaces = all
mydestination = $myhostname, localhost.$mydomain, localhost
local_recipient_maps = unix:passwd.byname $alias_maps
unknown_local_recipient_reject_code = 550
mynetworks_style = class
mynetworks = hash:/etc/postfix/network_table
relay_domains = $mydestination
relayhost = $mydomain
relay_recipient_maps = hash:/etc/postfix/relay_recipients
in_flow_delay = 1s
alias_maps = dbm:/etc/aliases
alias_database = dbm:/etc/aliases
mail_spool_directory = /var/mail
mailbox_transport = lmtp:unix:/file/name
header_checks = regexp:/etc/postfix/header_checks
local_destination_concurrency_limit = 2
default_destination_concurrency_limit = 20
debug_peer_level =
debug_peer_list = mydomain.com
sendmail_path = /usr/sbin/sendmail
newaliases_path = /usr/bin/newaliases
mailq_path = /usr/bin/mailq
setgid_group = postdrop
html_directory = no
manpage_directory = /usr/local/man
sample_directory = /etc/postfix
readme_directory = no          ...
```



Complex configuration files

```
queue_directory = /var/spool/postfix
command_directory = /usr/sbin
daemon_directory = /usr/libexec/postfix
data_directory = /var/lib/postfix
mail_owner = postfix
default_privs = nobody
myhostname = host.domain.tld
mydomain = domain.tld
myorigin = $myhostname
inet_interfaces = all
mydestination = $myhostname, localhost.$mydomain, localhost
local_recipient_maps = unix:passwd.byname
unknown_local_recipient_reject_code = 451
mynetworks_style = class
mynetworks = hash:/etc/postfix/networks
relay_domains = $mydestination
relayhost = $mydomain
relay_recipient_maps = hash:/etc/postfix/relay_recipient_maps
in_flow_delay = 1s
alias_maps = dbm:/etc/aliases
alias_database = dbm:/etc/aliases
mail_spool_directory = /var/mail
mailbox_transport = lmtp:unix:/file/name
header_checks = regexp:/etc/postfix/header_checks
local_destination_concurrency_limit = 2
default_destination_concurrency_limit = 20
debug_peer_level = 3
debug_peer_list = mydomain.com
sendmail_path = /usr/sbin/sendmail
newaliases_path = /usr/bin/newaliases
mailq_path = /usr/bin/mailq
setgid_group = postdrop
html_directory = no
manpage_directory = /usr/local/man
sample_directory = /etc/postfix
readme_directory = no
```

```
local_destination_concurrency_limit = 2
default_destination_concurrency_limit = 20
debug_peer_level = 3
debug_peer_list = mydomain.com
sendmail_path = /usr/sbin/sendmail
newaliases_path = /usr/bin/newaliases
```



What's missing from current tools?

- Existing tools reveal “**what**” happened:
 - Require knowledge of how applications work
- End users need to infer “**why**” manually
 - Want to know the **root cause** of the problem
 - Configuration settings
 - Input



Insight

- Profilers use brute-force analysis
 - Attribute cost of every event to source code
- Performance Summarization
 - Combine this brute-force analysis with a causal analysis of every event
 - Event is instruction or system call
 - Attribute costs to root causes



Performance Summarization

1. Find all events within scope (“what?”)
2. Attribute costs to events
3. Find root causes of events (“why?”)
4. Aggregate costs by root cause
5. Output ordered list of root causes

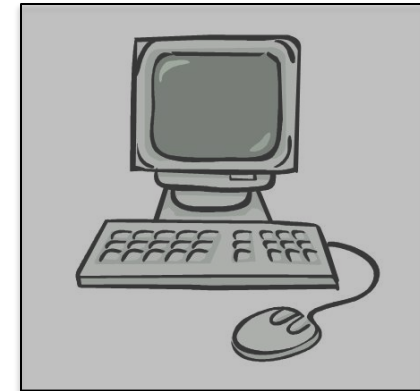
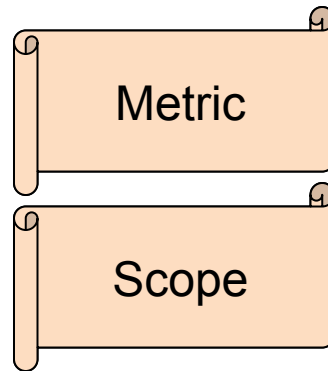


Outline

- *Motivation*
- How to use X-ray
- Building Blocks
- Performance Summarization
- Evaluation



How to use X-ray

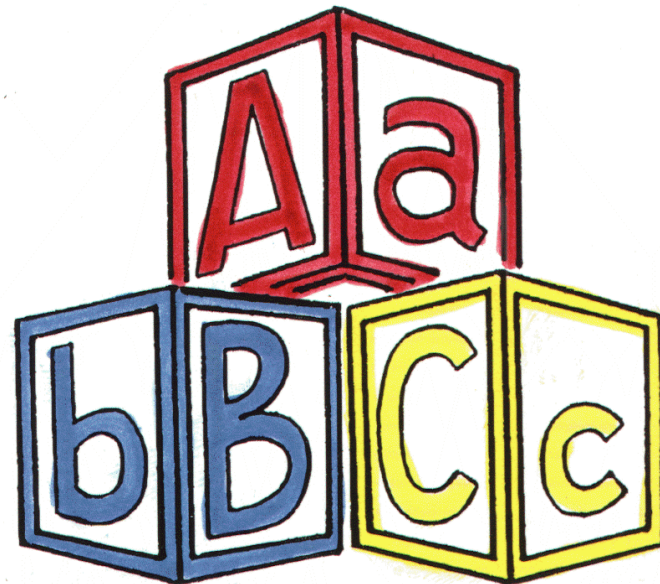


List of root causes:

- 1) ...
- 2) ...
- 3) ...

Building Blocks

- Deterministic Replay
- Causality Analysis



Deterministic Replay

- X-ray uses deterministic replay
 - Enables offline analysis
 - Minimizes perturbation due to analysis
- Requirements
 - Low online overhead
 - Add binary analysis (Pin) during replay



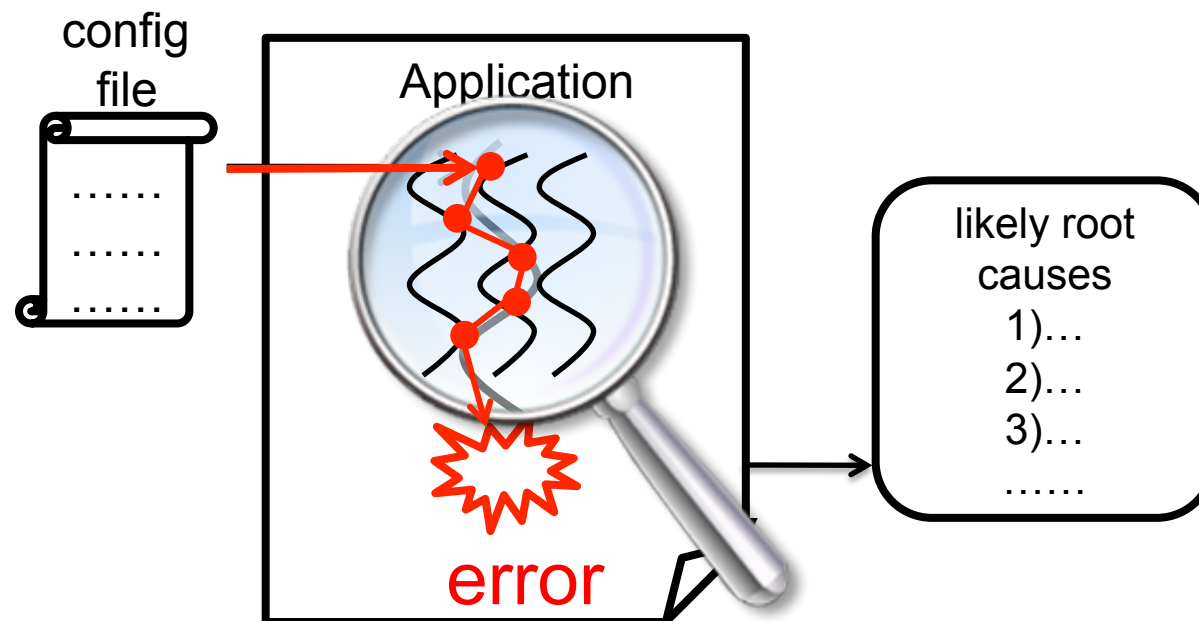
Replay with Instrumentation

- Challenge: Record & Replay executions differ
- Solution: Instrumentation awareness
- Compensate for differences caused by Pin:
 - System calls
 - Memory areas
 - Signals
 - Locking
- Adds overhead of 1–5%



ConfAid

- ConfAid reports root causes of a specific event
- Uses taint-tracking to determine causality



ConfAid Details

- Assign weights to taints to represent strength
 - Data flow taint > Control flow taint
 - Closer branches > Distant branches
 - Direct control flow > Indirect control flow
- Attributing root causes to all events is marginally more expensive than just one event

Outline

- *Motivation*
- *How to use X-ray*
- *Building Blocks*
- **Performance Summarization**
- **Evaluation**



X-ray scoping

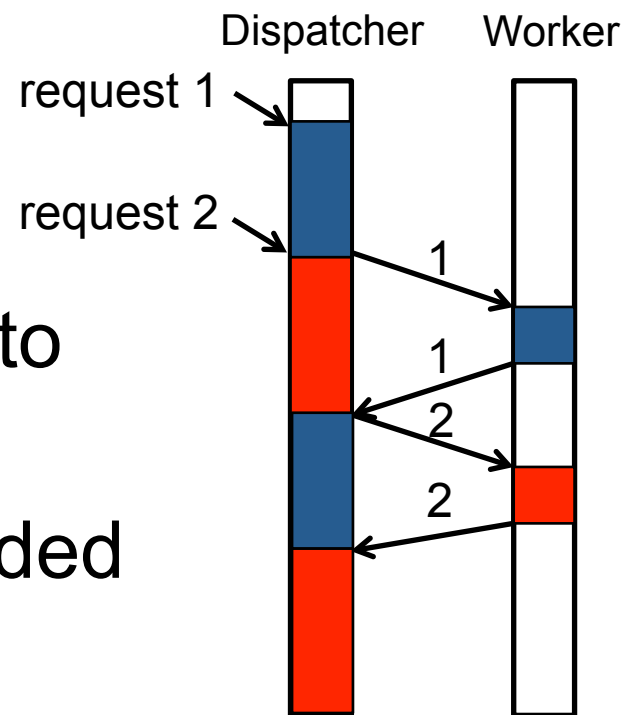
- Scope: portion of execution to analyze
 - Entire execution
 - Time period
 - Request
 - Multiple requests
- Request extraction
 - Identifies basic blocks that handle a request



Request Extraction: Method 1

- Track requests at process granularity

- Use external communication to understand requests
- Does not work for multi-threaded and event-based programs

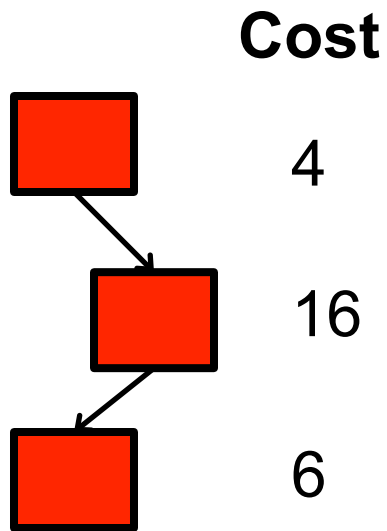


Request Extraction: Method 2

- Use taint-tracking to identify requests
- Propagate data and control flow taints
- Assign basic block to request with most weight




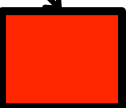

Performance Summarization




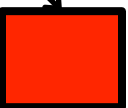

Cost can be:

- CPU usage
- Latency
- File system usage
- Network usage


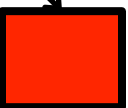

Performance Summarization

	Latency	Root Causes
	4	foo: 0.5
	16	foo: 0.25
	6	foo: 0.5, bar: 0.25

Performance Summarization

	Latency	Root Causes	foo	bar
	4	foo: 0.5	2	0
	16	foo: 0.25	4	0
	6	foo: 0.5, bar: 0.25	3	1.5

Performance Summarization

	Latency	Root Causes	foo	bar
	4	foo: 0.5	2	0
	16	foo: 0.25	4	0
	6	foo: 0.5, bar: 0.25	3	1.5

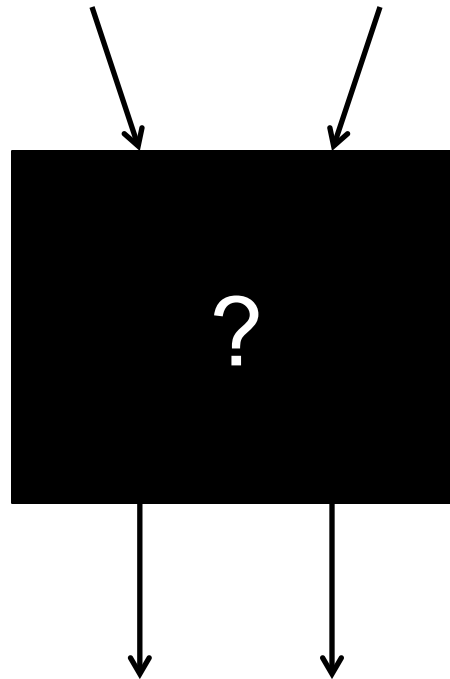
9 1.5

Output:

- 1) foo
- 2) bar

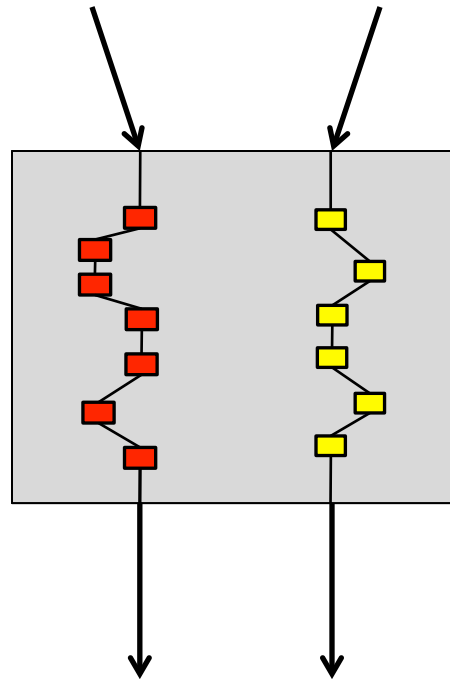
Differential Performance Summarization

- Identify why two requests differed in performance

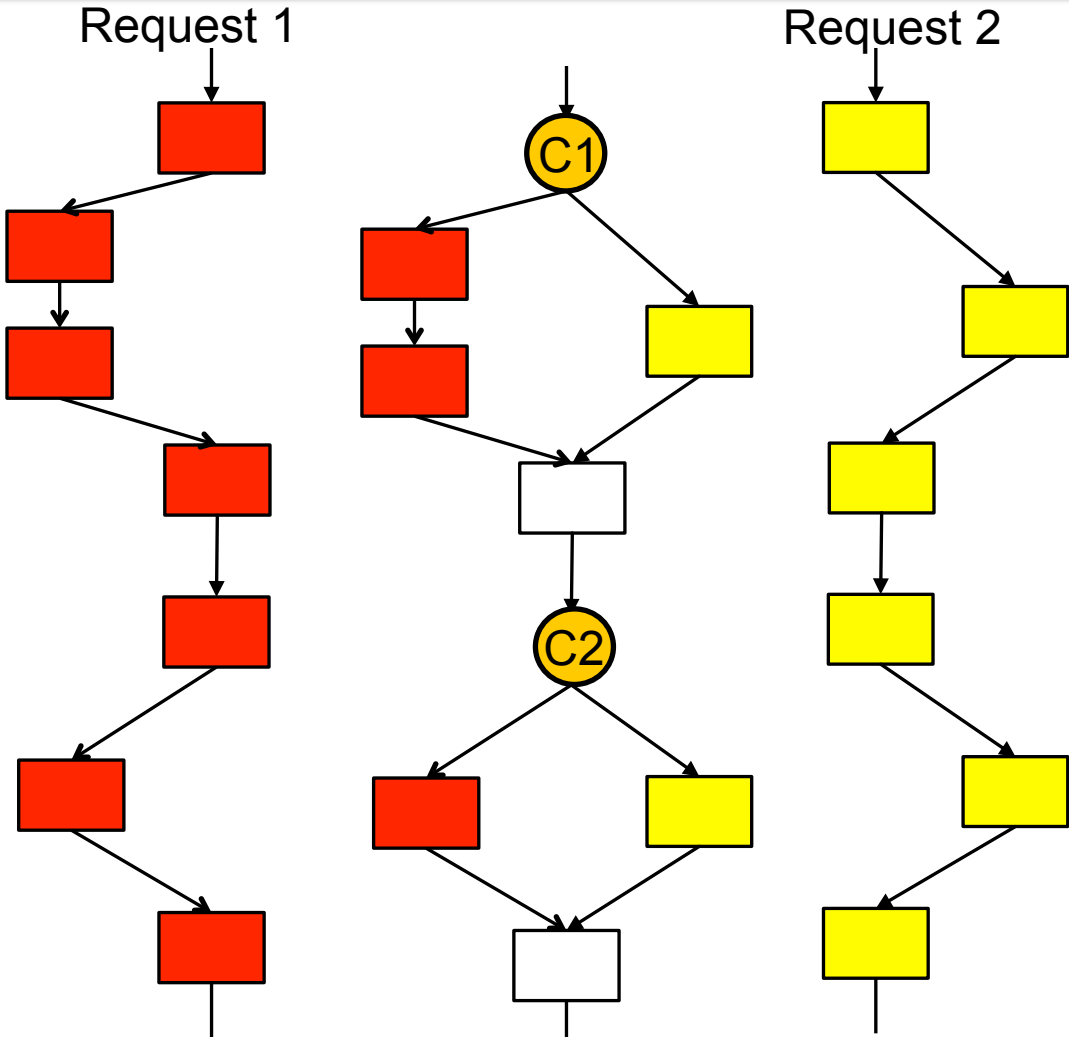


Differential Performance Summarization

- Identify why two requests differed in performance



Differential Performance Summarization

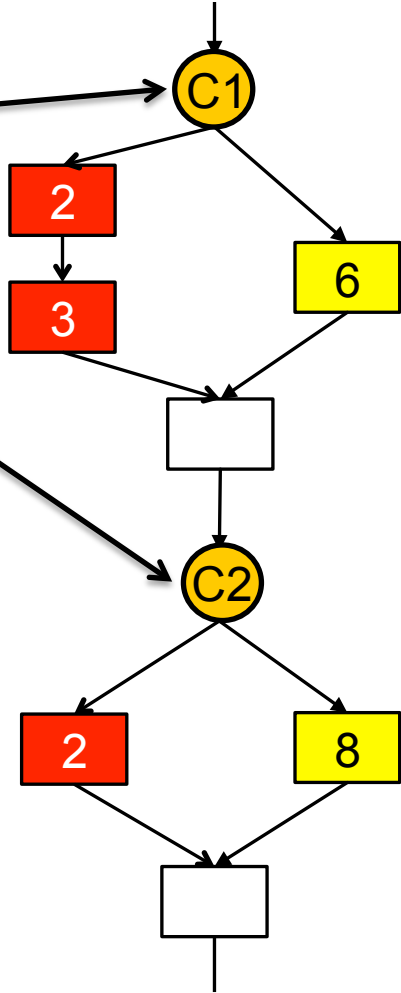


Michael Chow



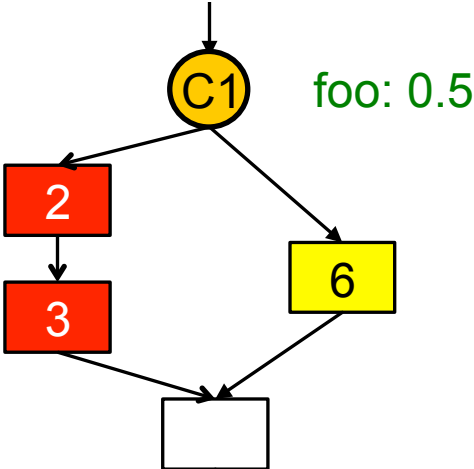
Differential Performance Summarization

Identify conditionals where paths diverge

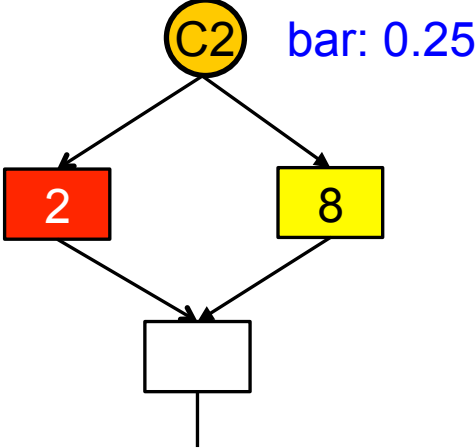


Differential Performance Summarization

Differential Cost of C1: 1

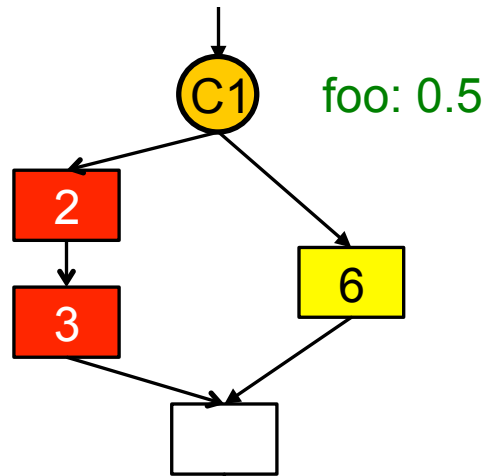


Differential Cost of C2: 6



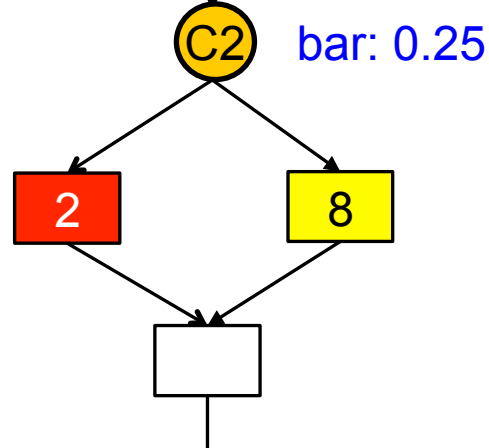
Differential Performance Summarization

Differential Cost of C1: 1



foo	bar
0.5	0

Differential Cost of C2: 6



0	1.5
<hr/>	
0.5	1.5

Output:

1) bar
2) foo



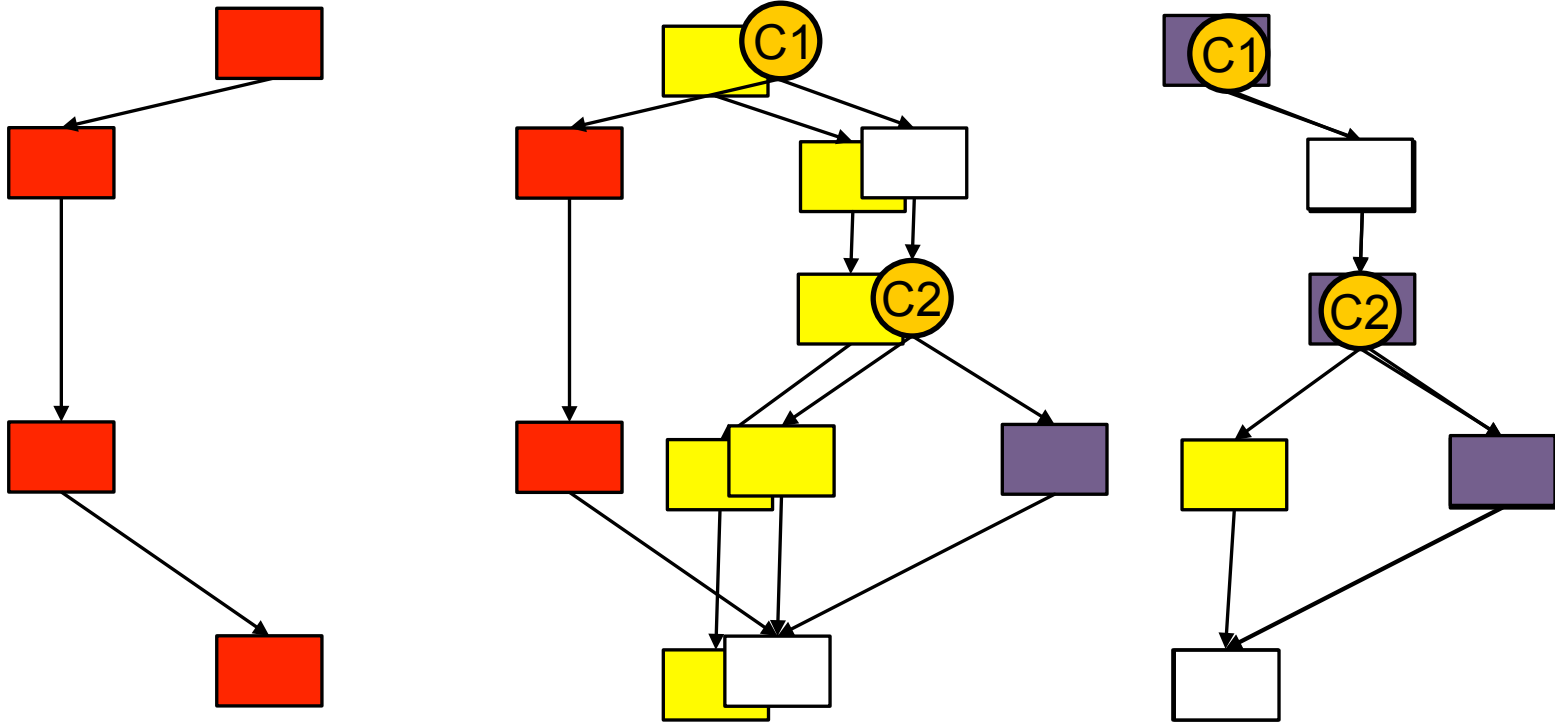
Multi-Input Performance Summarization

- Compare a large number of requests

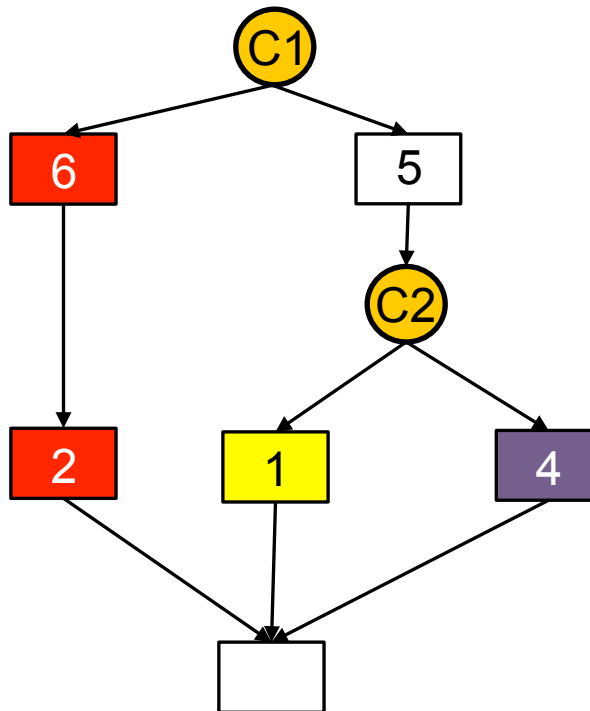
Insight: Help identify similar requests



Multi-Input Performance Summarization

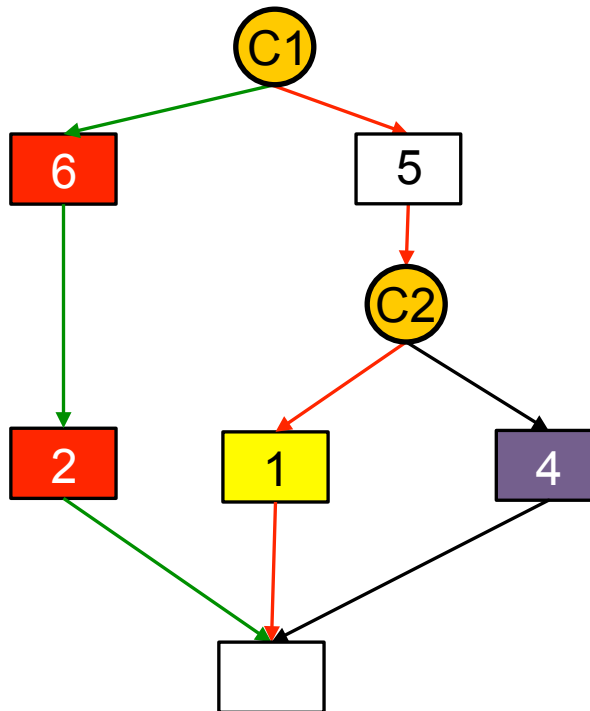


Multi-Input Performance Summarization



Multi-Input Performance Summarization

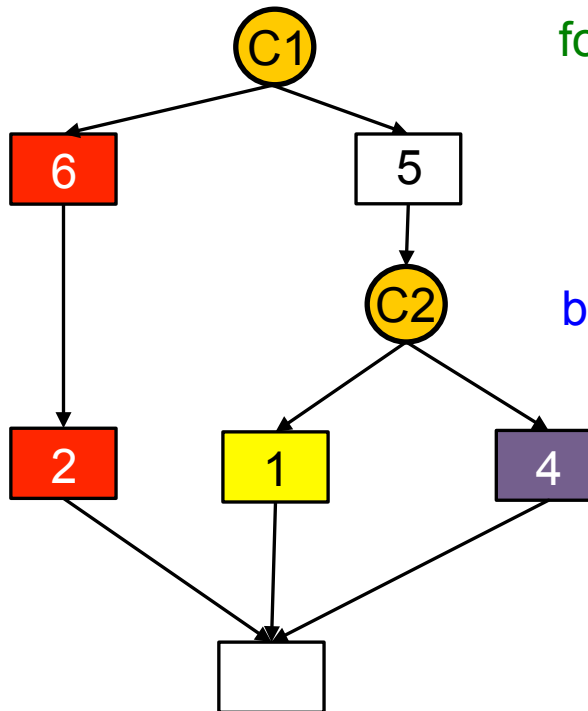
Cost of Conditional C1: 2



Multi-Input Performance Summarization

Cost of Conditional C1: 2

Cost of Conditional C2: 3



	foo	bar
foo: 0.2	0.4	
bar: 0.5		1.5
	<hr/>	<hr/>
	0.4	1.5

Output:

- 1) bar
 - 2) foo

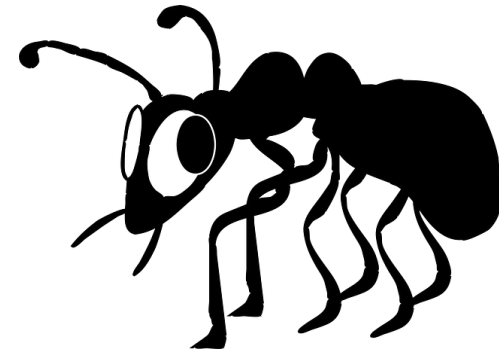


Limitations of X-ray

- Only considers configuration settings and inputs as root causes
- Only track causality at the user-level

Evaluation

- Looked at 4 apps, 17 performance bugs
 - Apache, Postfix, PostgreSQL, lighttpd
- Found bugs on mailing lists, forums, blogs, documentation, books



X-ray Results

Application	Test Case	Rank			
		1 st	2 nd	3 rd	4 th
Apache	1	MaxKeepAliveRequests	On	Directory	...
	2	Allow	domain	ServerRoot	...
	3	On	HostNameLookups	Directory	...
	4	AuthUserFile	Input	ServerRoot	...
	5	On	ContentDigest	Directory	...
	6	DocumentRoot	Input(eTag)	Listen	...
Postfix	1	debug_peer_list	domain	queue_directory	...
	2	body_checks	myhostname	mydestination	...
	3	reject_rbl_client	smtpd_client_restrictions	reject_rbl_client	...
PostgreSQL	1	timezone	datestyle	log_timezone	...
	2	shared_buffers	wal_sync_method	synchronous_commit	...
	3	checkpoint_timeout	shared_buffers	max_connections	...
	4	wal_writer_delay	shared_buffers	wal_level	...
	5	track_counts	log_timezone	lc_time	...
lighttpd	1	auth.backend.htpasswd.userfile	Input	server.use-ipv6	...
	2	document_root	Input(eTag)	server_root	...
	3	server.max-keep-alive-requests	server_root	state_dir	...



X-ray Results

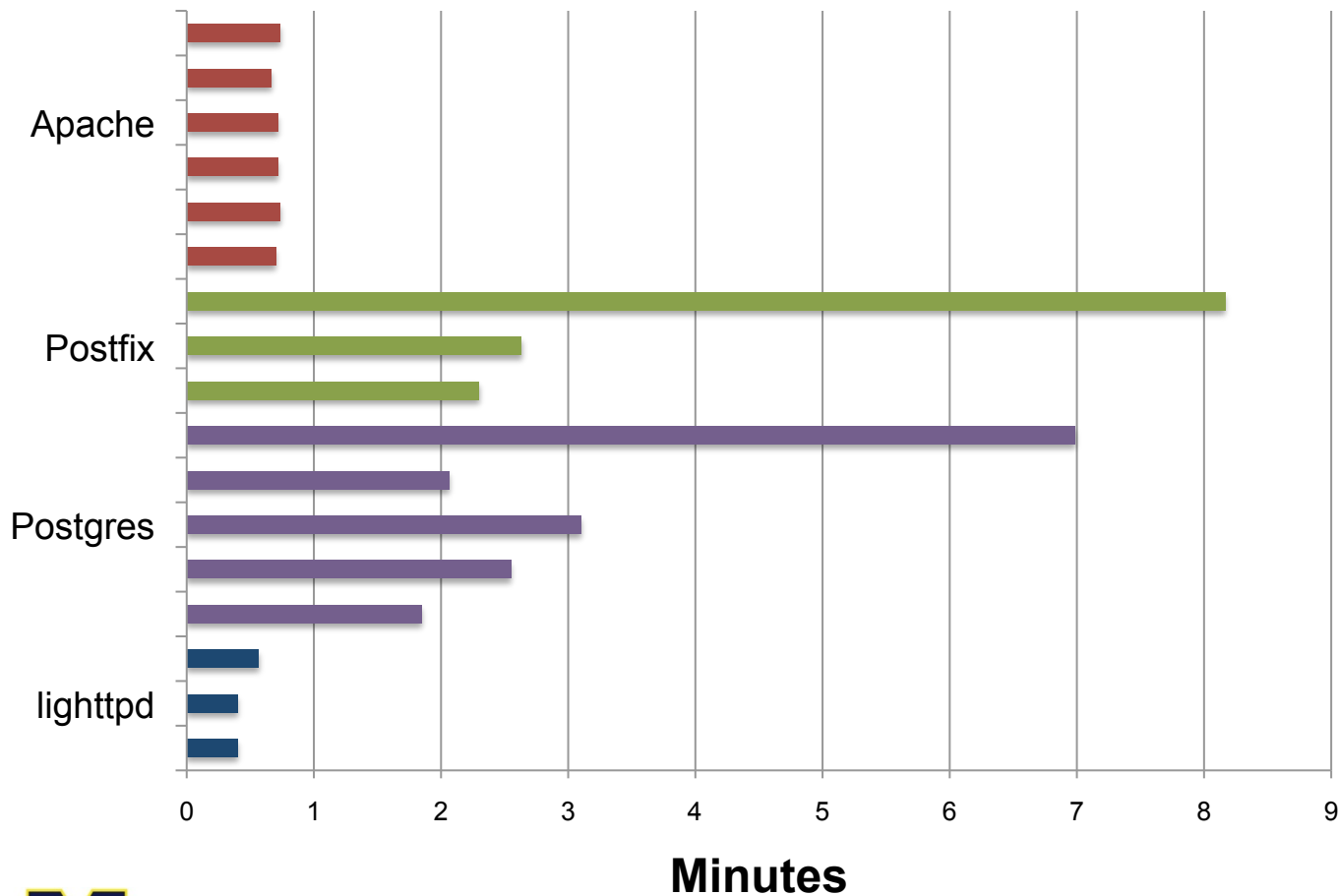
Application	Test Case	Rank			
		1 st	2 nd	3 rd	4 th
Apache	1	MaxKeepAliveRequests	On	Directory	...
	2	Allow	domain	ServerRoot	...
	3	On	HostNameLookups	Directory	...
	4	AuthUserFile	Input	ServerRoot	...
	5	On	ContentDigest	Directory	...
					...
Postfix					...
					...
					...
Postgr					...
					...
	4	wal_writer_delay	shared_buffers	wal_level	...
	5	track_counts	log_timezone	lc_time	...
lighttpd	1	auth.backend.htpasswd.userfile	Input	server.use-ipv6	...
	2	document_root	Input(eTag)	server_root	...
	3	server.max-keep-alive-requests	server_root	state_dir	...

Correct root cause(s) ranked or tied for first in 16 out of 17



How fast is X-ray?

X-ray Analysis Time



Online
overhead of
only 1-5%

Average
Analysis Time
2:04



Conclusion

- X-ray helps diagnose performance issues using **performance summarization**
- X-ray requires no source code
- X-ray correctly ranked the true root cause first or tied for first in 16 out of 17 issues

Questions



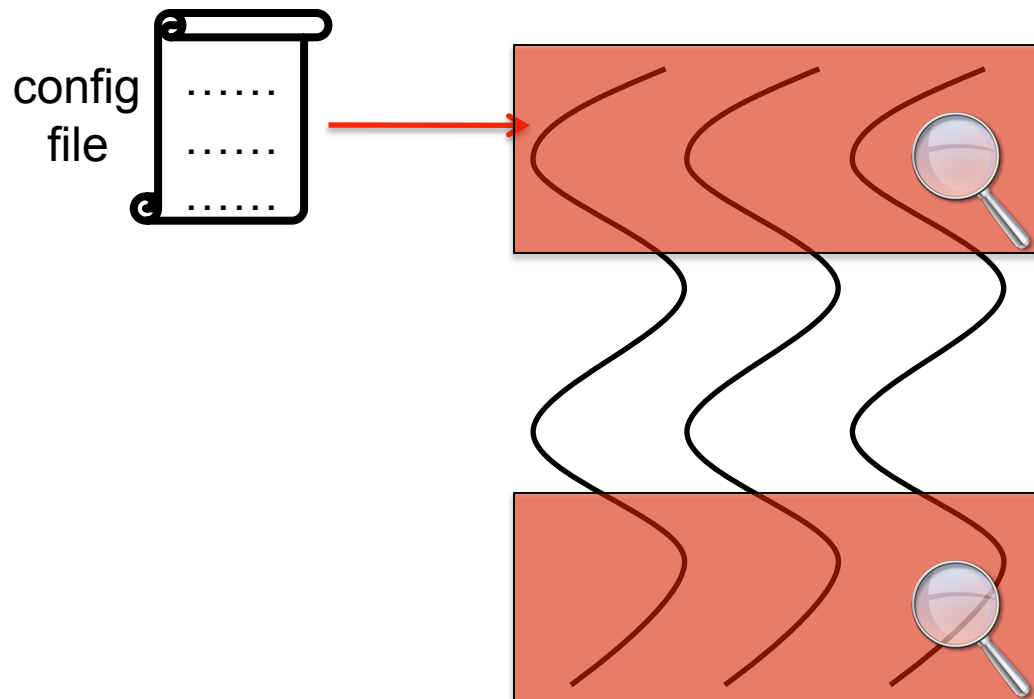


Backup Slides



Fast Forwarding

- Problem: What about long runs?
- Solution: Fast-forward taints



Related Work

- Detecting the problem
 - Profilers, DTrace, Event Tracing for Windows (ETW), VTune
- Identifying the root cause(s)
 - X-trace, SNAP, Pinpoint, Magpie, Spectroscope, Aguilera *et. al*
- Solving the problem
 - PeerPressure, Strider, Chronus



Selection of Bugs

- Selected performance issues with root cause(s) of configuration options or input
- Required full range of X-ray to solve
 - Scope: Request, Time Interval, Differential
 - Metric: Latency, CPU, File System, Network
- Bugs had mix of binary and variable options



Attribution Strategies

- Strategies to attribute costs to root causes

Strategy	Number of False Positives ranked higher				True root cause unranked
	0	1	2	3+	
Absolute	21	2	0	0	0
Normalized	20	0	3	0	0
Winner-take-all	15	3	1	2	2
Learning	20	2	1	0	0



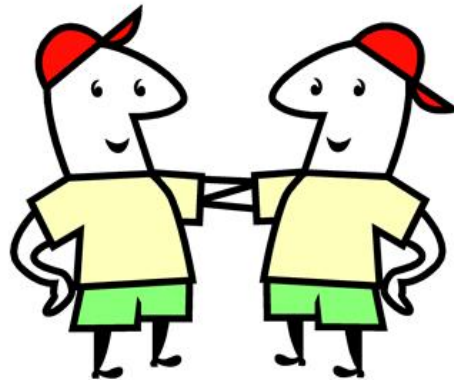
Sensitivity to ConfAid Heuristics

- Confaid ages control flow taint

Age Weight	Number of False Positives ranked higher				True root cause unranked
	0	1	2	3+	
0	15	3	1	2	2
0.125	19	2	2	0	0
0.25	20	3	0	0	0
0.5	21	2	0	0	0
0.75	20	0	1	2	0
0.875	20	0	1	2	0
1	8	3	2	10	0

Case Study: Differential Summarization

- How similar do requests have to be?



- In-depth look at a performance issue
- Sent a variety of different requests

Case Study: Differential Summarization

HTTP Requests	Ranking			
	1st	2nd	3rd	4th
GET/GET	MaxKeepAliveRequests	On	Directory	...
GET/POST	MaxKeepAliveRequests	On	Directory	...
POST/GET	Input	DocumentRoot	MaxKeepAliveRequests	...
Hundred	Input	ServerRoot	MaxKeepAliveRequests	...