#### Towards Reproducible Evaluation of Large-Scale Distributed Systems

#### Miguel Matos INESC-ID & IST, U. Lisboa

http://www.gsd.inesc-id.pt/~mm

ApPLIED'18



### "Science moves forward by corroboration - when researchers verify others' results."

-Challenges in irreproducible research. Nature 2016



#### "Science advances faster when people waste less time pursuing false leads."

-Challenges in irreproducible research. Nature 2016



"No research paper can ever be considered to be the final word, but there are too many that do not stand up to further study."

-Challenges in irreproducible research. Nature 2016



# What leads to irreproducibility?



### What leads to irreproducibility?

- Incomplete description of
  - system
  - environment
  - benchmark
- Inherent system and environment variability
- Uncontrollable variables
- Poor metrics



# Irreproducibility factors

- Sometimes these factors are *knowingly* omitted
  - lack of space, time, proper language
- But often factors are *unknowingly* omitted
  - We don't know what we don't know

[BlackBurn et al, TOPLAS 2016]



# What can we do in distributed computing in particular?



"A distributed system is one in which the <u>failure</u> of a <u>computer</u> you didn't even <u>know existed</u> can render you own computer <u>unusable</u>."

-L. Lamport



# **Distributed System**

- Processes
- Network
- Fault patterns
- Monitor & Benchmarking





# **Distributed System**

- Processes
- Network
- Fault patterns
- Monitor & Benchmarking

# What leads to irreproducibility in each of these?

在于无效意义。这次意思这些是是不可能的意思的问题,这次意大学的意思。但是不是不是无法的情况,还是是不是是不是不能。





- Hardware specification?
- Operating System version/configuration?
- System libraries?



- Hardware specification?
- Operating System version/configuration?
- System libraries?

"we deploy our prototype on Amazon's EC2 using 1,000 m4.2xlarge virtual machines (VMs), each of which has 8 cores and up to 1 Gbps network throughput."

[Some systems paper]



Effect of environment variable size on speedup



[Mytkowicz, ASPLOS'09]

ANGAINO

nesc id

Effect of linking order variable size on speedup



[Mytkowicz, ASPLOS'09]

















Our novel approach, Blue, improves performance over Red by 15% ...





Our novel approach, Blue, improves performance over Red by 15% ...

... due to a lucky combination of environment variables size and linking order



"The first principle is that you must not fool yourself and you are the easiest person to fool." Richard P. Feynman

Our novel approach, Blue, improves performance over Red by 15% ...

... due to a lucky combination of environment variables size and linking order



### How do we precisely describe all the relevant factors?





- Bandwidth, latency, jitter, packet loss?
- Network topology?
- Contention with the other tenants?



Effect of buffer size and congestion algorithms on link utilization



Effect of buffer size and congestion algorithms on link utilization



Effect of buffer size and congestion algorithms on link utilization



How do we precisely describe (and enforce) all the relevant factors?



### "In the next experiment, we <u>failed</u> 5% of random processes every 10 minutes."

-Random systems paper



- Fault injection used to assess reliability
- Faults can target processes, the network or both
- What impacts reproducibility?
- Limitless fault types
  - kill processes
  - drop packets
  - hog resources
  - corrupt disk blocks



Effect of packet loss in TCP performance



Effect of packet loss in TCP performance



Effect of packet loss in TCP performance



Effect of packet loss in TCP performance



Effect of packet loss in TCP performance



- Benchmark results can will vary
  - Across independent runs
    - Do multiple runs, present summary statistics



#### What do these three images have in common?









#### What do these three images have in common?



#### Same summary statistics

x=54.26, y=47.83 sdx=16.76 sdy=26.93 Pearson's r=-0.06

[Matejka et al., CHI'17]



- Benchmark results can will vary
  - Across benchmarking/measurement tools



Results of method profiling with different JVM profilers

[Mytkowicz et al., PLDI'10]



Results of method profiling with different JVM profilers

[Mytkowicz et al., PLDI'10]



Results of method profiling with different JVM profilers

[Mytkowicz et al., PLDI'10]



# Variability Causes

- Processes and Network
  - Affected by many *known* and *unknown* factors
- Fault patterns
  - Affected by fault *implementation*
- Monitor & Benchmarking



• Affected by tool variability & data interpretation



# What can we do?





# Approach

- We need a systematic approach to describe what we know about the system and environment
- and also what we don't know (or don't think it matters)

29

- Build the tools to run the system and distribute the artifacts
  - Performance
  - Correctness
  - Debugging



# ANGAINOE

Angainor was constructed to hold Melkor, the primordial source of evil that sought to corrupt the people of Middle-Earth



### Process

- Today easier than before
- Containers
  - concise description of runtime and configurations
- Glue still being heavily developed
  - deployment
  - interactions with network and faults



### Process

- Specify
  - image
  - network
  - additional commands

```
services:
   nginx:
    image: "nginx:alpine"
    labels:
            org.lsdsuite.stats:
     networks:
            - nginx-net
    siege:
       image: "yokogawa/siege"
       command: ["siege" , "-
              concurrent=3",
              "--delay=3" ,
              "http://tasks.nginx/" ]
       networks:
              - nginx-net
networks:
       nginx-net:
              driver: overlay
       ipam:
       config:
              - subnet: 10.22.0.0/16
```

### Process

- Specify
  - image
  - network
  - additional commands



```
services:
   nginx:
    image: "nginx:alpine"
    labels:
            org.lsdsuite.stats:
     networks:
            - nginx-net
    siege:
       image: "yokogawa/siege"
       command: ["siege" , "-
               concurrent=3" ,
              "--delay=3" ,
              "http://tasks.nginx/" ]
       networks:
              - nginx-net
networks:
       nginx-net:
              driver: overlay
       ipam:
       config:
              - subnet: 10.22.0.0/16
```

### Network

- Network has a huge range of knobs
  - topology
  - bandwidth, latency, jitter
  - packet loss
- Specify and emulate network constraints for reproducibility
- Evaluate impact of network properties in the system

ANGA

### Network

- Specify network
  - topology
  - bandwidth
  - latency
  - jitter
  - packet loss

```
- network
   - nginx-net
- bridges
   - s1
   - s2
-links
  -link:
      origin: nginx
      dest: s1
      latency: 10
      upload: 10Mbps
      download: 10Mbps
      jitter: 0.25
  -link:
      origin: s1
      dest: s2
      latency: 20
      upload: 100Mbps
      download: 200Mbps
      jitter: 0.25
      drop: 0.001
```

### Network

- Specify network
  - topology
  - bandwidth
  - latency
  - jitter
  - packet loss
- network - nginx-net - bridges **-** s1 **-** s2 -links -link: origin: nginx dest: s1 latency: 10 upload: 10Mbps download: 10Mbps jitter: 0.25 -link: origin: s1 dest: s2 latency: 20 upload: 100Mbps download: 200Mbps jitter: 0.25 drop: 0.001

### Faults

- Precise description of fault injection
  - which, when, what pattern
- Extensible fault *library* 
  - under the same language as before
  - integrated with the process and network description



### Faults

#### • Specify

- process addition
- process removal
- Type of fault
- when to inject

- Fault patterns
  - synthetic
  - real traces

services: siege: churn: synthetic: - start: 10 - time: 120 add: 500 - time: 240 add: 500 - time: 360 kill: 100 type: KILL - time: 480 kill: 10% type: TERM - end: 600

### Faults



- process addition
- process removal
- Type of fault
- when to inject



- synthetic
- real traces



services: siege: churn: synthetic: - start: 10 - time: 120 add: 500 - time: 240 add: 500 - time: 360 kill: 100 type: KILL - time: 480 kill: 10% type: TERM - end: 600

- Benchmarking is yet another application/service
- Specified exactly as yet another process and network



- Monitor
  - Automated logging
    - system metrics
    - environment events (e.g.: faults)
  - Warn about
    - bottlenecks
    - reproducibility in peril
  - Derive data representation automatically





- Scenario
  - Evaluate latency of Byzantine Fault Tolerant library
    - subject to churn
    - varying number of clients



Number of clients over time clients 401 server killed 1 server started Clients 20

Time



0





# **Reproducibility Results**

BFT library running in a geo-distributed EC2 deployment



**Our reproduction** 



# **Reproducibility Results**

BFT library running in a geo-distributed EC2 deployment

# The latter can now run 'on any day' and yields the same results

B= BFT-SMaRt, W= WHEAT

(spu

600

Oregon (L) Ireland Sydney São Paulo Virginia





50th 🖂

**Miguel Matos** 

# **Reproducibility Results**

BFT library running in a geo-distributed EC2 deployment

# The latter can now run 'on any day' and yields the same results

B= BFT-SMaRt, W= WHEAT

(spr



50th 🖂

## Discussion

- Reproducibility is a hard problem
  - Many know and unknown factors affect results
- In Angainor, we are proposing a systematic approach to reduce these variability factors
- Ability to properly evaluate
  - Performance evaluation
  - Correctness
  - Debugging



### We are hiring

#### **Open Post-doc and PhD positions at INESC-ID Lisboa**

#### http://angainor.science



In collaboration with S. Brulhart, H. Mercier, I. Senhaji, J. Neves, V. Schiavoni, J Thode

