



MICRO-50, Boston, October 2017

Tutorial:

# Microarchitecture Level Reliability Assessment

Throughput and Accuracy

<http://micro50-tutorial.di.uoa.gr/>

Organizers/Presenters:

Athanasios Chatzidimitriou, Manolis Kaliorakis, Dimitris Gizopoulos



MICRO-50, Boston, October 2017

Part 3:

# GeFIN fast modes Accelerating a SFI campaign

# Opportunities for acceleration

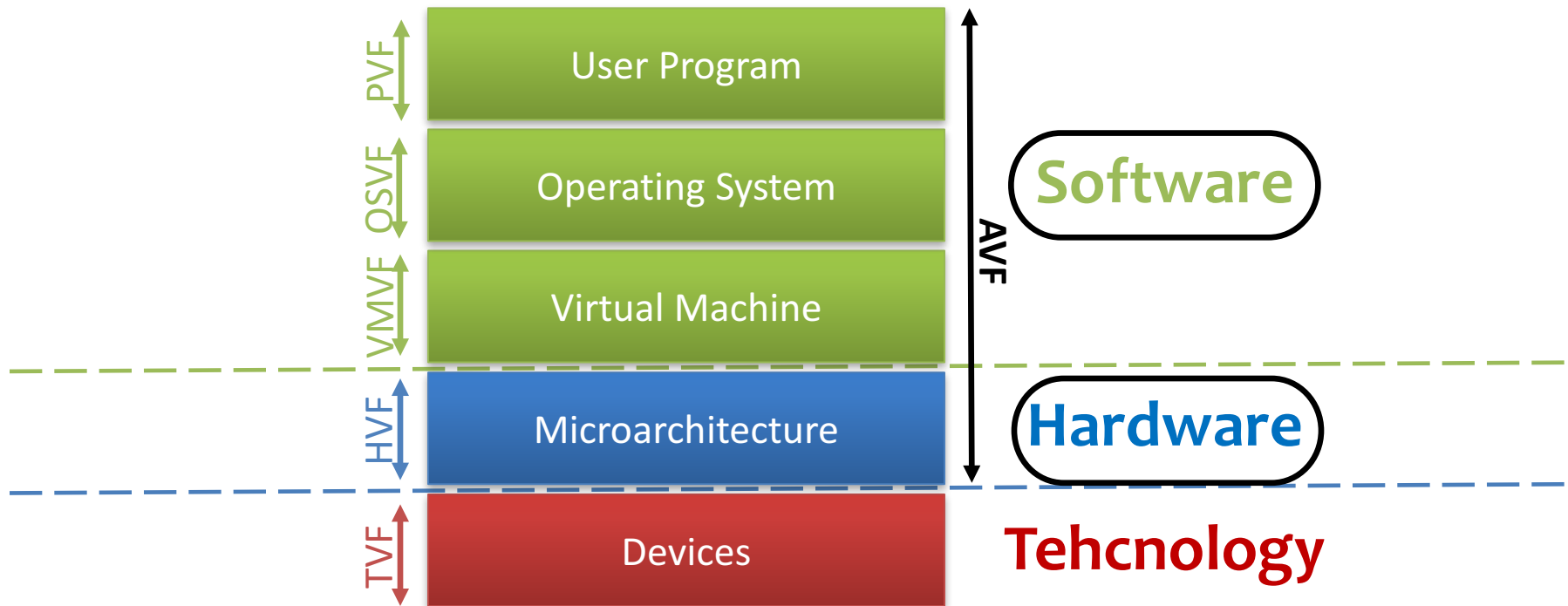
- **Fault propagation**
  - The **system** vulnerability **stack**
  - System **layers**
- **Fault injection simulation lifetime**
  - Fault simulation **epochs**
  - Time **distribution**



MICRO-50, Boston

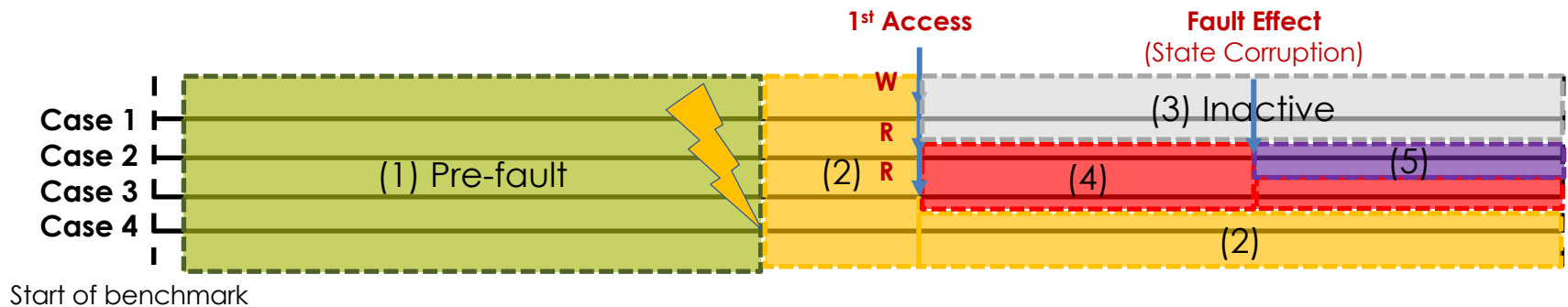


# System vulnerability stack\*



\* V.Sridharan, D.R.Kaeli, "Using hardware vulnerability factors to enhance AVF analysis", ISCA 2010

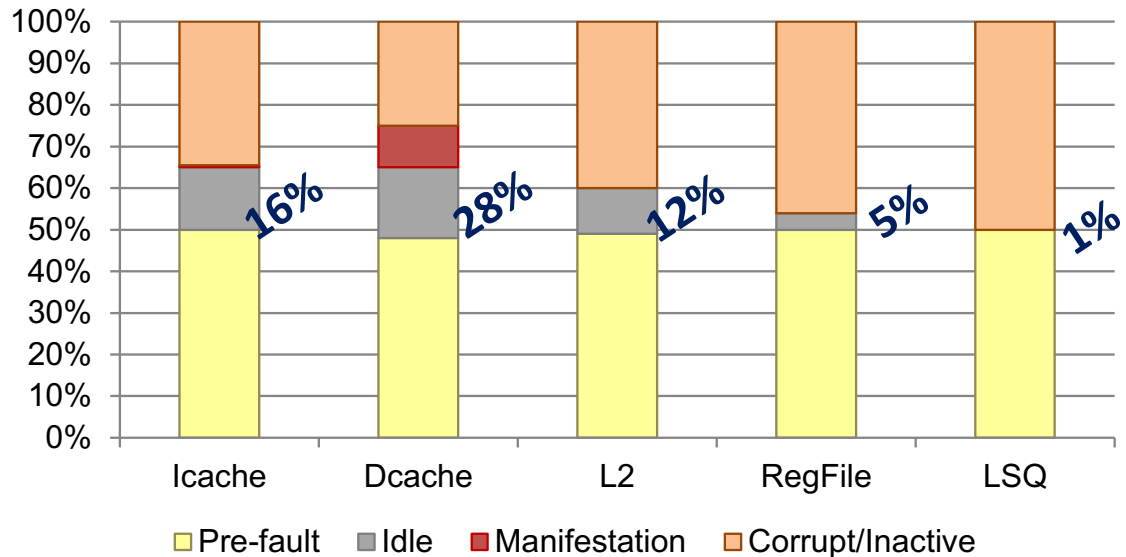
# Anatomy of a Fault injection simulation



- Fault injection
  - First access
  - First visible effect
- 1 Pre-fault epoch
  - 2 Idle epoch
  - 3 Inactive epoch
  - 4 Manifestation epoch
  - 5 Corruption epoch

# Time distribution

- **Idle & Manifestation** epochs are enough to **capture** the fault effect.
- Measured to consume only **13%** (avg) of total **simulation time**
- Our goal is to **focus** only on **critical** simulation portions

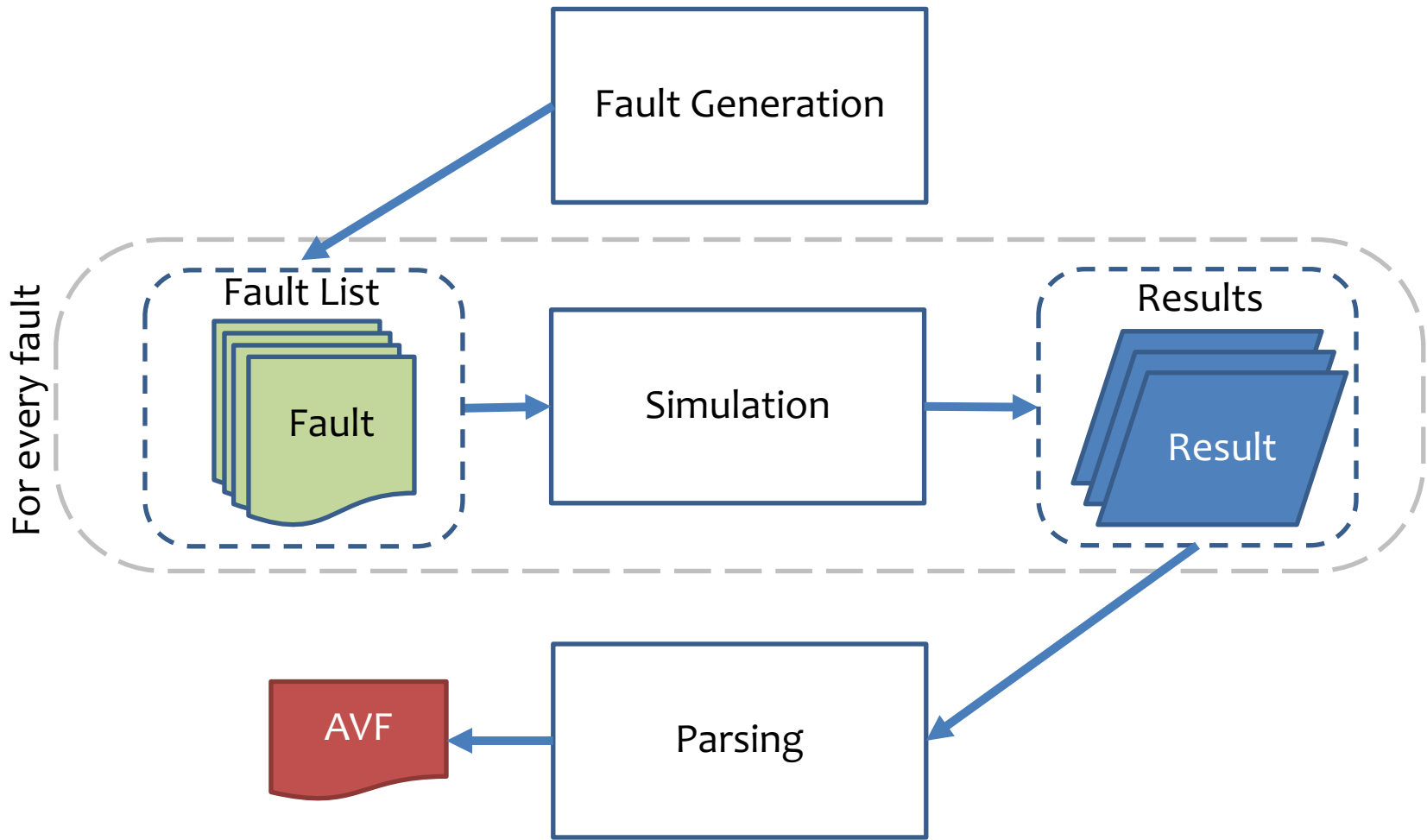


# Accelerating SFI campaign

- **Parallelization**

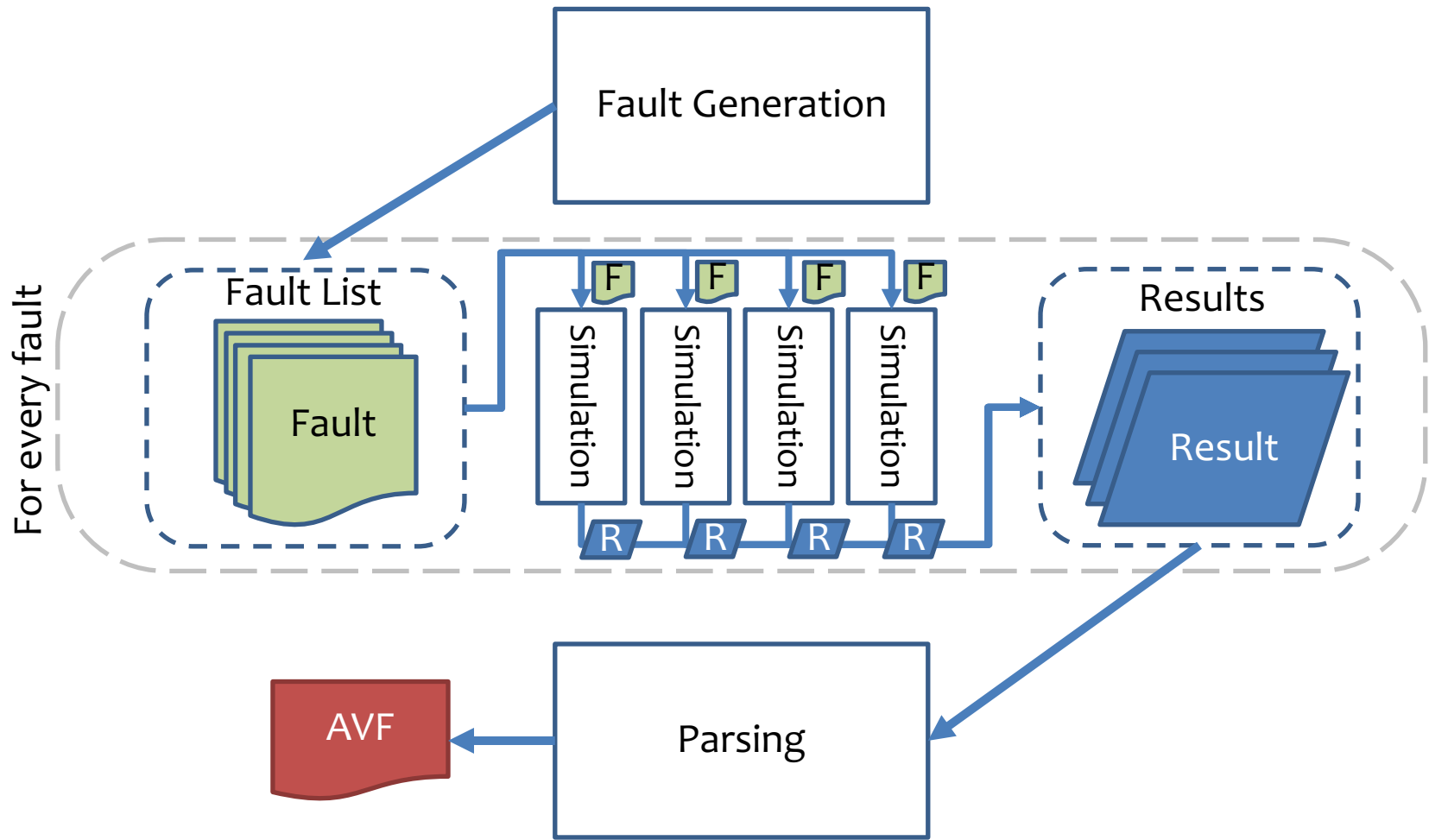
- Seemingly obvious, each fault injection is independent
- Exploit maximum CPU Utilization

# Statistical Fault Injection





# Statistical Fault Injection



# Accelerating SFI campaign

- **Parallelization**
- **Fast Forwarding**
  - Skipping pre-fault epoch
  - Enhancing checkpoint functionality



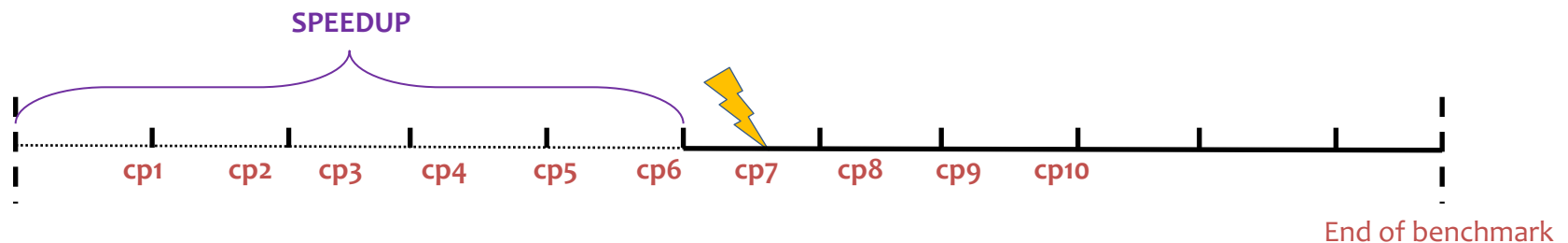
MICRO-50, Boston



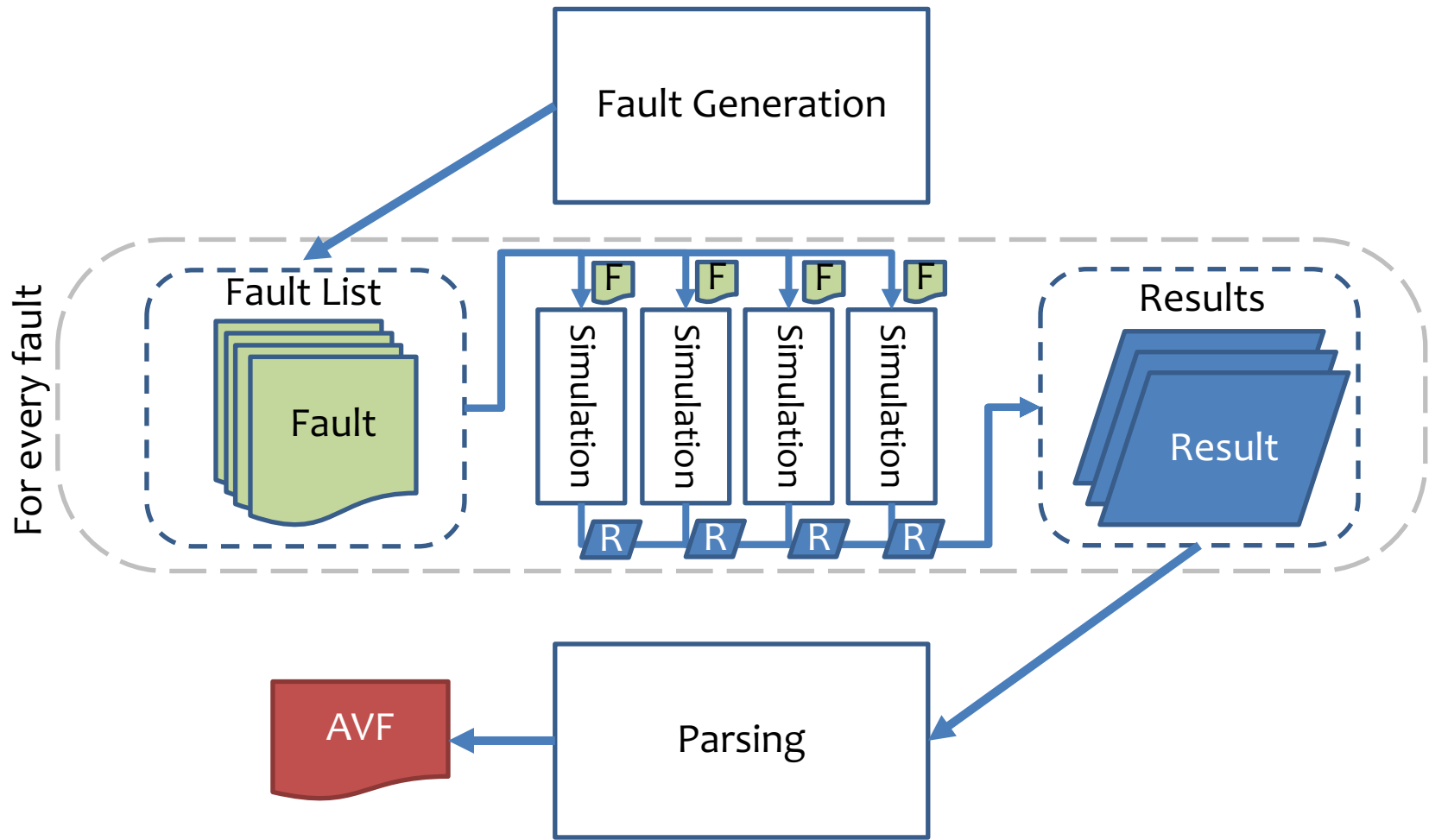
# Fast forwarding

## Enhanced checkpointing

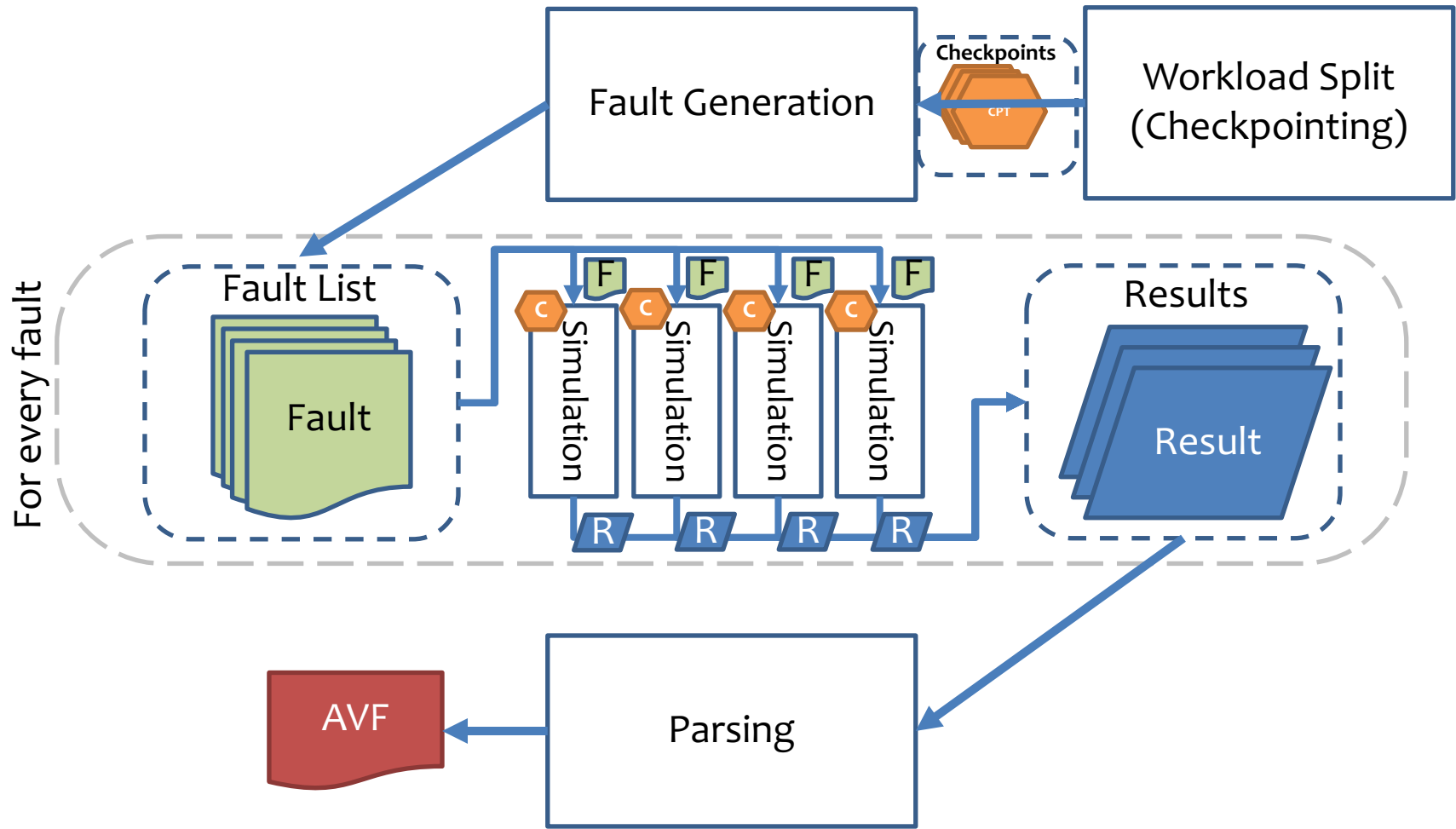
- Include **cache memories**
- Minimal loss of microarchitectural state
- Designed to skip the **pre-fault** epoch
- **<1p.u.** deviation
- Up to ~**45%** campaign **acceleration** compared to baseline



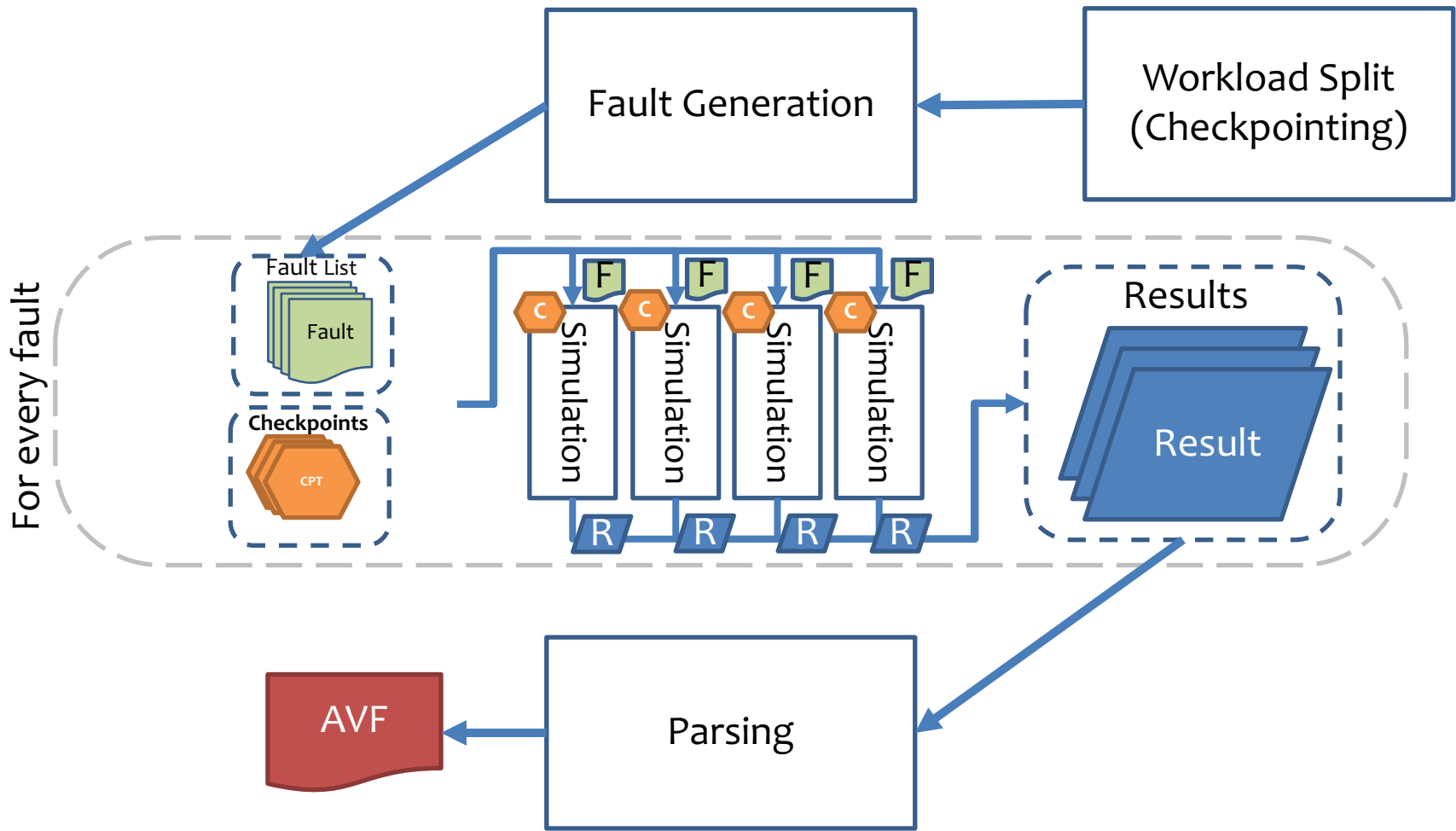
# Statistical Fault Injection



# Statistical Fault Injection



# Statistical Fault Injection



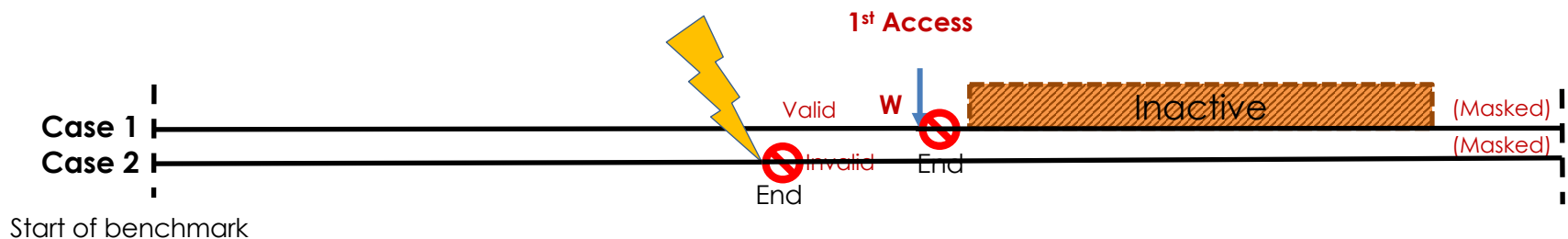
# Accelerating SFI campaign

- Parallelization
- Fast Forwarding
- Early Stop
  - Stopping simulation on overwrite
  - Stopping on invalid/unused entries

# Early stop

## Early stop on overwrite/invalid

- Stopping simulation when fault gets **overwritten** – Targets **Inactive** period
- **No** loss of accuracy
- **14-47%** campaign acceleration compared to Baseline (depends on the component)





# Accelerating SFI campaign

- **Parallelization**
- **Fast Forwarding**
- **Early Stop**
- **HVF estimation**
  - Identify fault propagation to software layer



MICRO-50, Boston



# HVF estimation

## Early stop on program corruption

- Stopping simulation when a state corruption is detected - **Software Corruption** period
- Up to **12%** campaign acceleration compared to mode2
- Exactly between **Hardware** & **Software vulnerability**
- Achieved through Gem5 Exec Tracing features – can cause **overhead**



MICRO-50, Boston



# Accelerating SFI campaign

- Parallelization
- Fast Forwarding
- Early Stop
- HVF estimation
- Early limit + early switch
  - Stop simulation before the end
  - Switch to emulation



MICRO-50, Boston



# Early limit

## Early switch to functional emulation

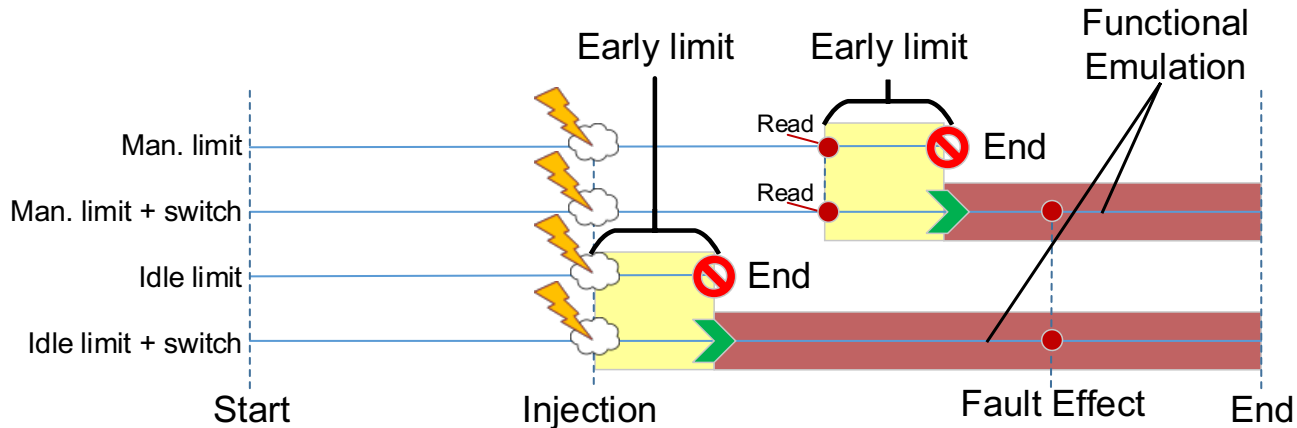
- Switch to atomic (functional) core
- Up to **20x** times faster (or 1000x with **KVM** support)
- Covers **software vulnerability**

## Early limit

- **Stops simulation** after pre-defined amount of time
- Supports limit for **idle** and **manifestation** epochs
- **Accuracy loss**



# Early limit



- Early limit can be used **individually** or **combined** with early switch.
- Adjusting accuracy loss
- **How** should it be used?

# Capturing behavior patterns

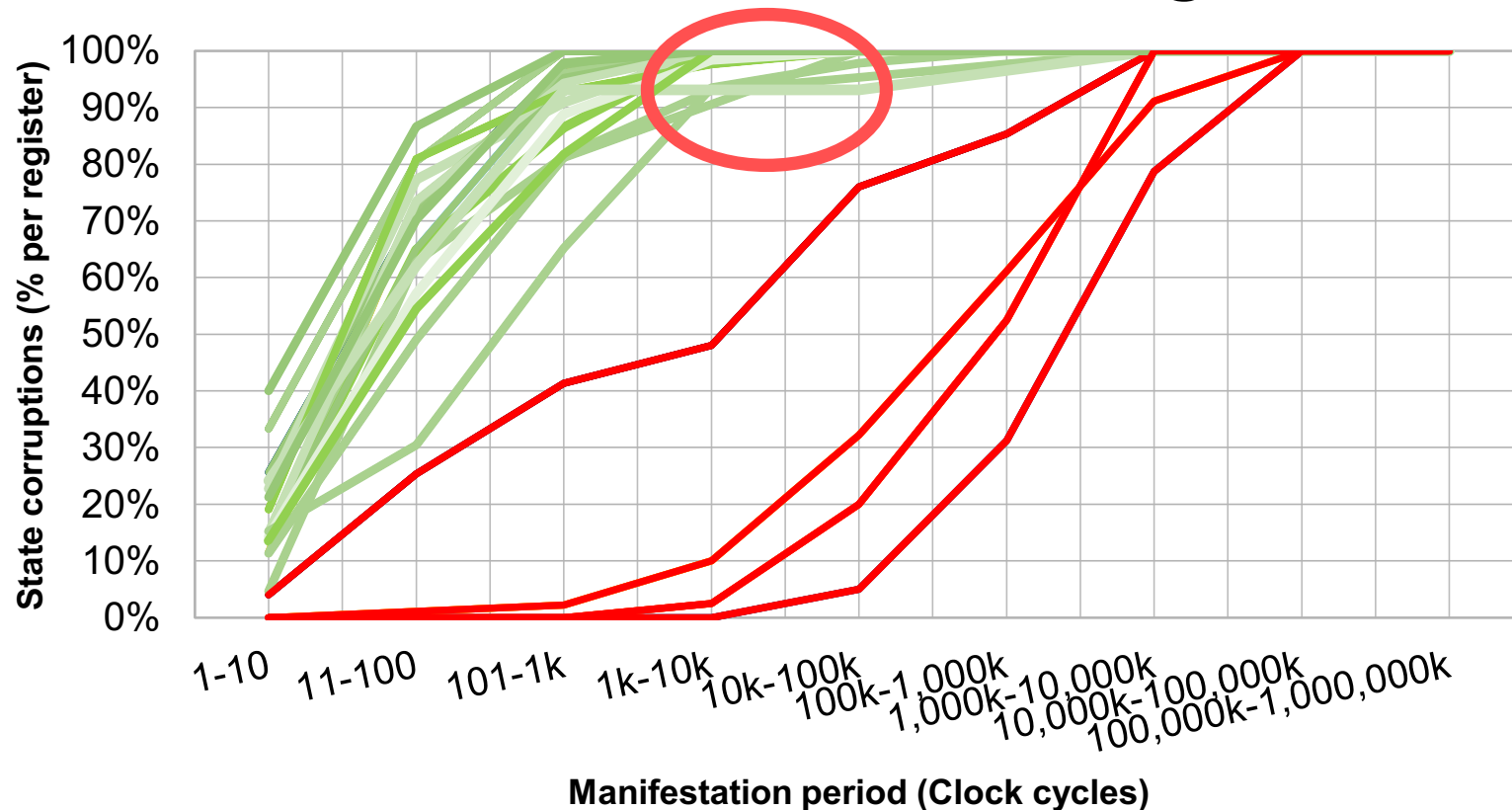
- Monitoring entry attributes to further **accelerate** fault simulation
- **Inner-pipeline** components
  - Short residency time
  - **Small** idle epochs
  - Register file, LSQ
- **Outer-pipeline** structures
  - **Long** idle and manifestation epochs
  - Cache memories



# Pipeline components

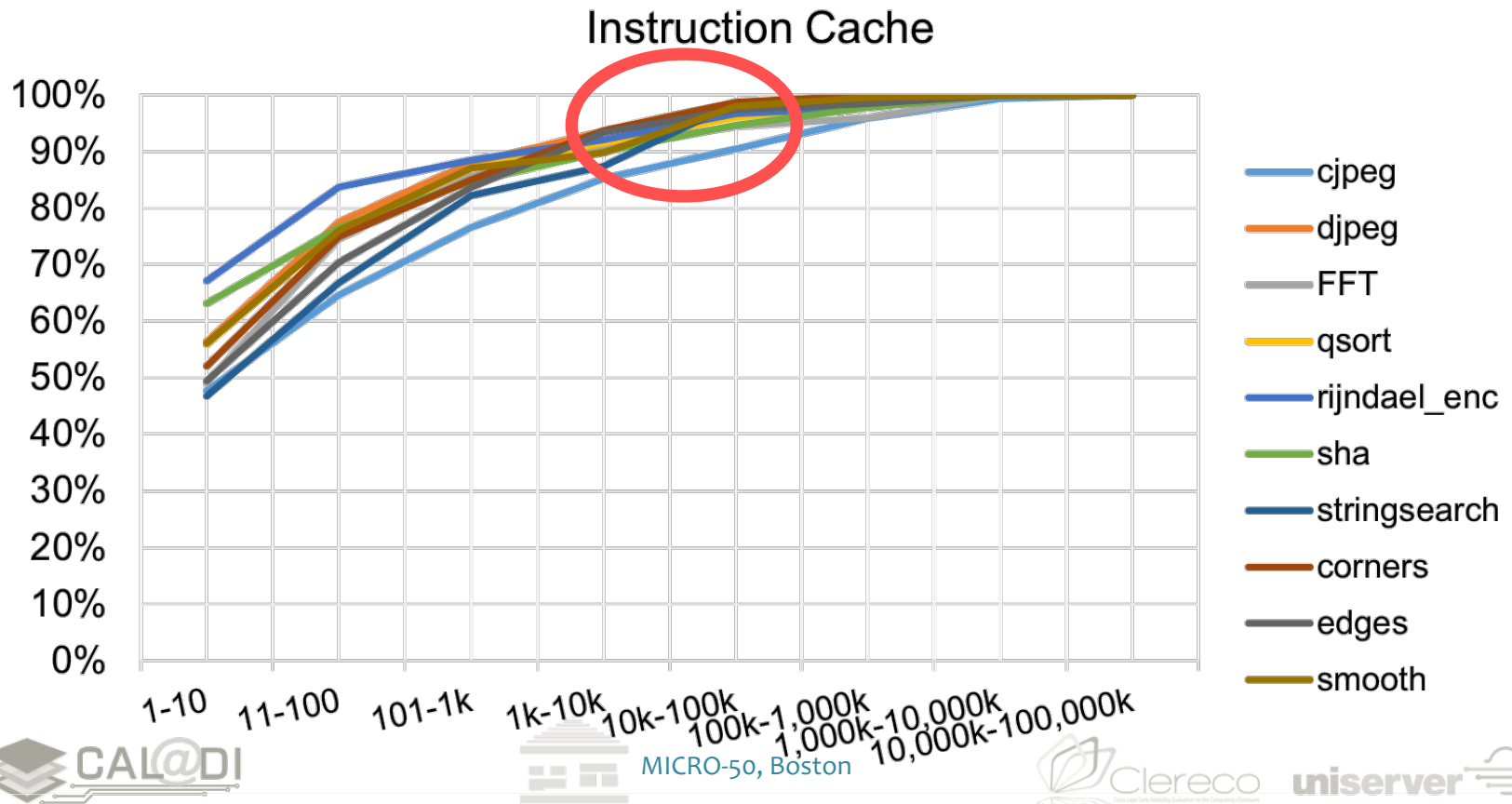
- Register file
  - Architecturally mapped
  - Dynamic registers

## ARM ISA architectural registers



# Cache memories

- Instruction cache
  - **Steady** manifestation epoch behavior
  - **Prediction-friendly**

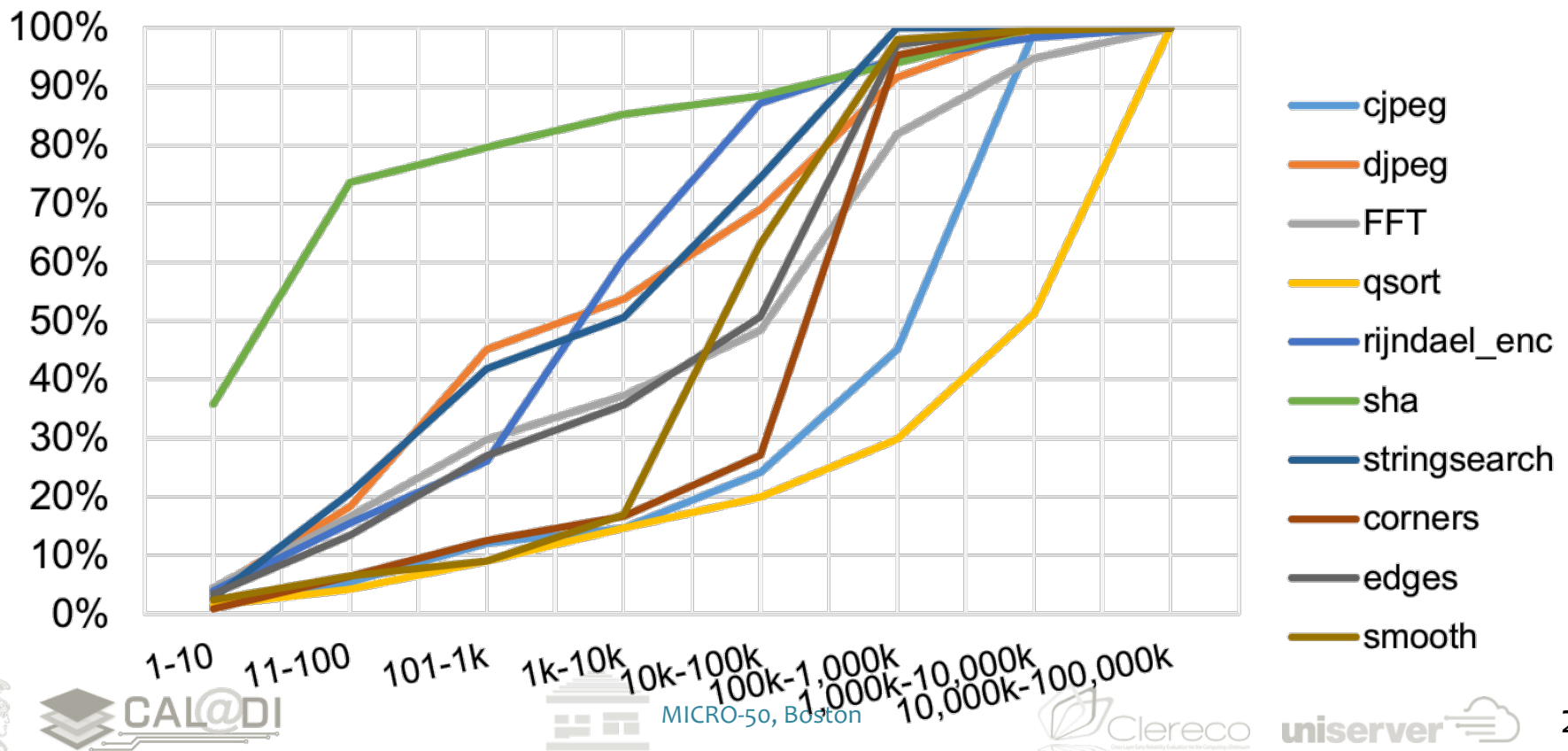




# Cache memories

- Data cache
  - **No patterns**
- L2 has similar behavior

Data Cache



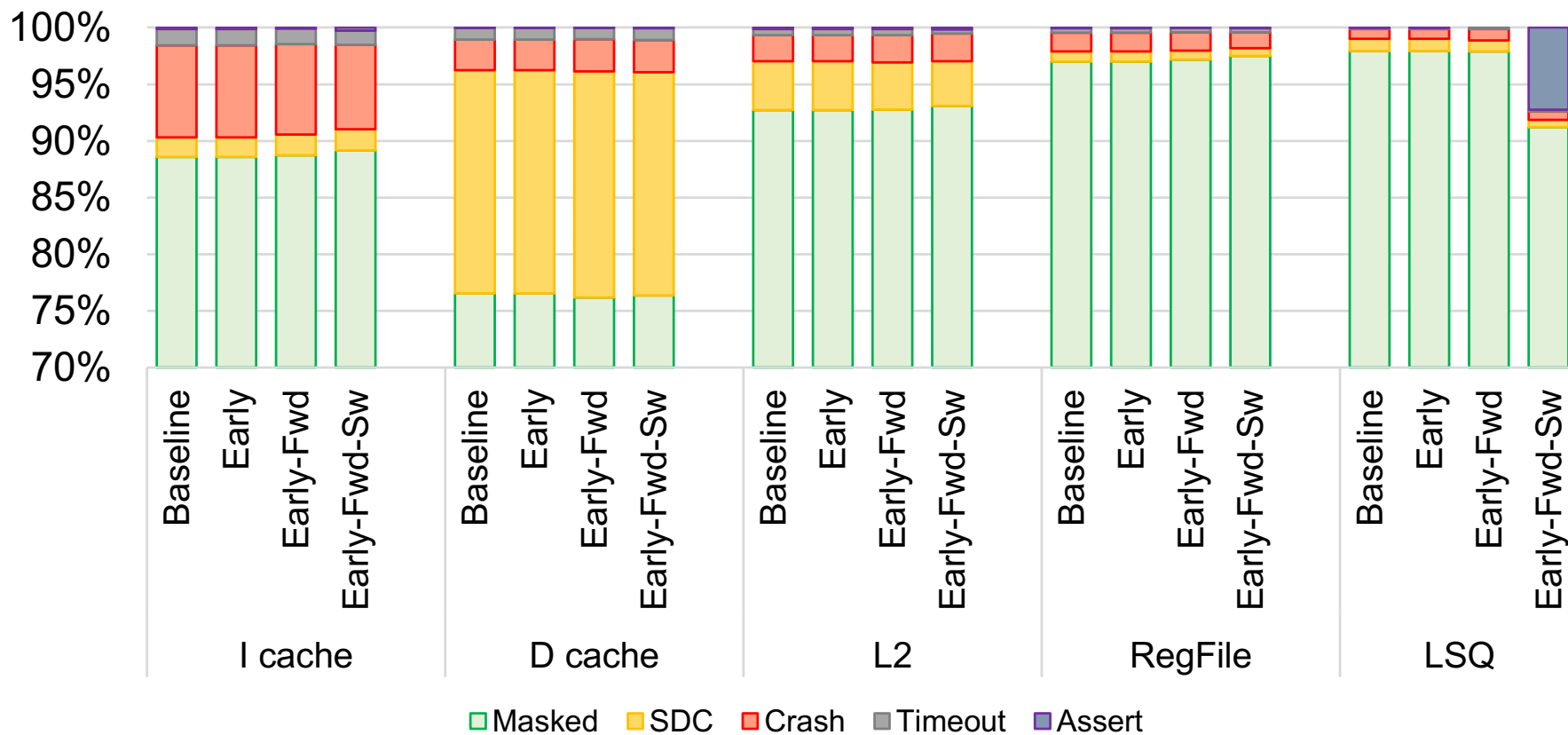
# Experimental results

- MiBench suite
- Configuration presets

Acronym	AVF measurement	Hardware vulnerability measurement
Baseline	<ul style="list-style-type: none"><li>▪ Baseline fault inject</li></ul>	<ul style="list-style-type: none"><li>▪ Early stop on program corruption</li></ul>
Early	<ul style="list-style-type: none"><li>▪ Early stop on overwrite</li></ul>	<ul style="list-style-type: none"><li>▪ Early stop on overwrite</li></ul>
Early-Fwd	<ul style="list-style-type: none"><li>▪ Fast forwarding</li><li>▪ Early stop on overwrite</li></ul>	<ul style="list-style-type: none"><li>▪ Fast forwarding</li><li>▪ Early stop on overwrite</li></ul>
Early-Fwd-Sw	<ul style="list-style-type: none"><li>▪ Fast forwarding</li><li>▪ Early stop on overwrite</li><li>▪ Early switch after 100,000 cycles</li></ul>	<ul style="list-style-type: none"><li>▪ Fast forwarding</li><li>▪ Early stop on overwrite</li><li>▪ Early stop after 100,000 cycles</li></ul>

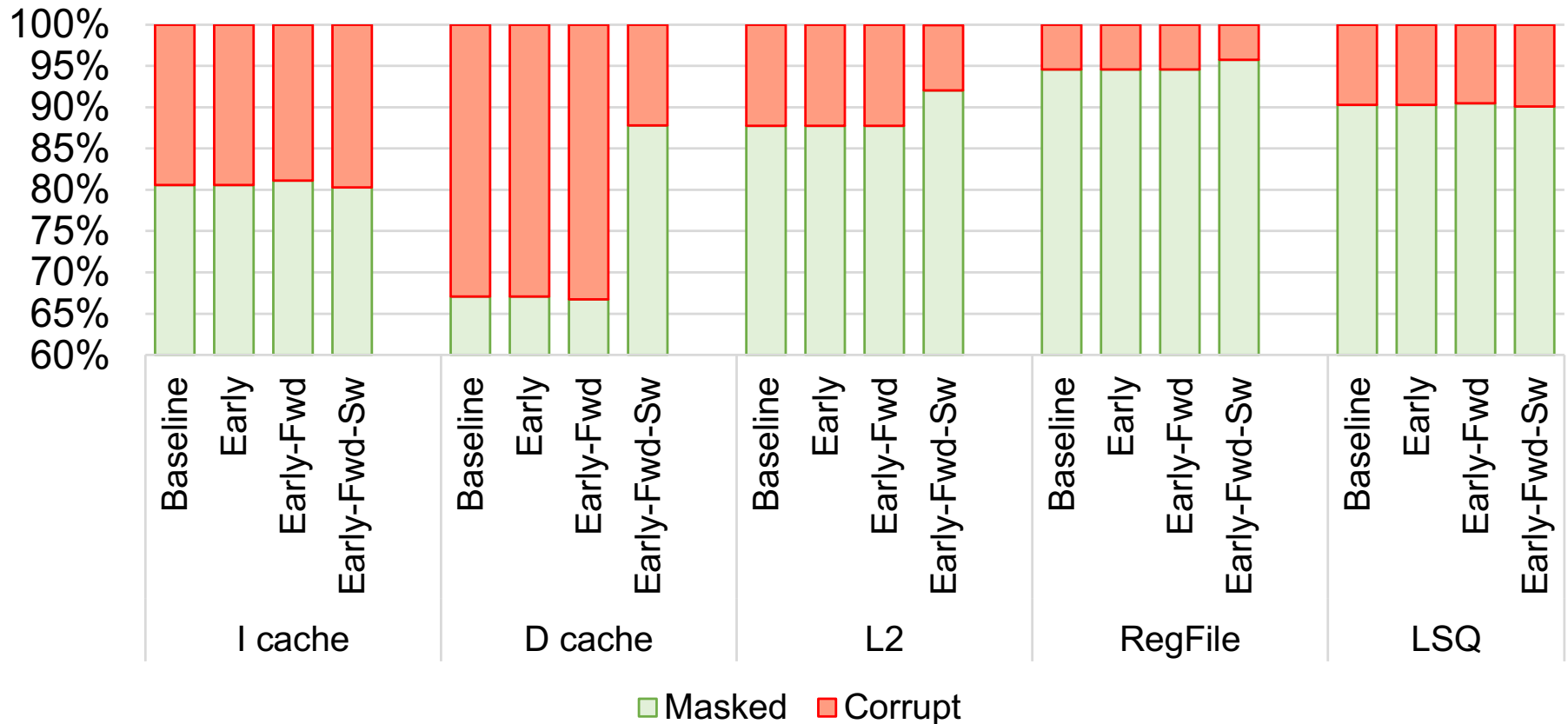
# Experimental results

Average ARM AVF per component



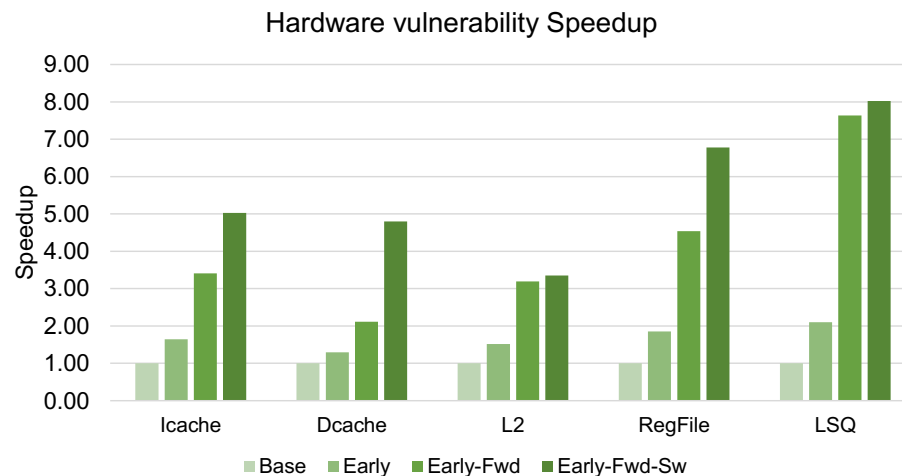
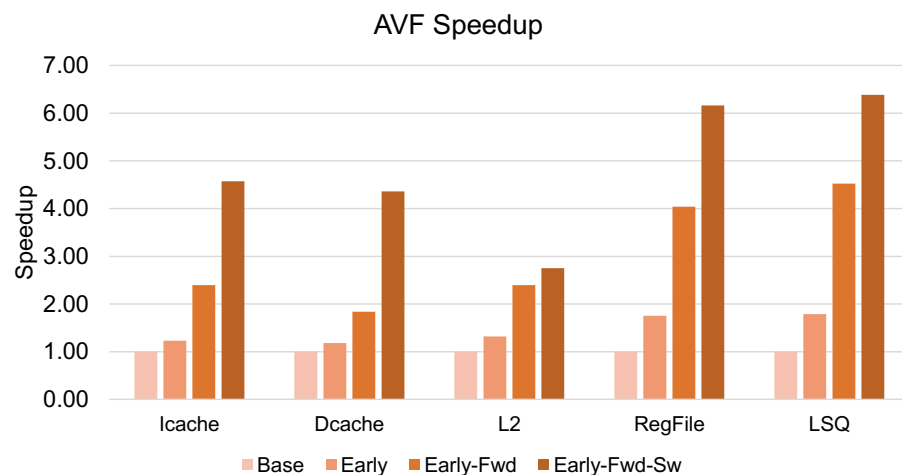
# Experimental results

Average ARM hardware vulnerability



# Experimental results

- Speedup is **proportional** to target **epoch duration**
- Fast-forwarding offers most of the reported acceleration
- **Prediction-friendly** structures report significant **speedup** with **minor accuracy losses**
- Low **L2** utilization reports minor speedup on early limits due to **unused** resources and long idle periods





MICRO-50, Boston, October 2017

Next ...  
**Part 4**

Tutorial:

# Microarchitecture Level Reliability Assessment

Throughput and Accuracy

<http://micro50-tutorial.di.uoa.gr/>

Organizers/Presenters:

Athanasios Chatzidimitriou, Manolis Kaliorakis, Dimitris Gizopoulos