Dependability Benchmarking of Off-the-Shelf OS Kernels

Karama Kanoun

45th Meeting of IFIP Working Group 10.4, Moorea, French Polynesia, March 5-9, 2004

Partially financed by the European Commission

DBench
DependabilityBenchmarking

DBench

❑ Objective of DBench

- ✦ Conceptual framework & experimental environment for benchmarking the dependability of (C)OTS and COTS-based systems
  - ➨ Concepts, specifications and guidelines for dependability benchmarking
  - ➨ Dependability benchmark prototypes

❑ Current / final results

- ✦ A framework for dependability benchmarking
- ✦ A set of benchmark specifications and associated prototypes
  - ➨ User point of view: robustness benchmarks wrt external errors
  - ➨ Emphasis on representativeness and validation

DBench
DependabilityBenchmarking

DBench Framework

Categorization

Benchmark Target - BT
(System nature
Application area
Operating environment)

Benchmarking context
(Life-cycle phase
Benchmark user
Benchmark performer
Benchmark purpose)

Measures

Measure nature
(qualitative or quantitative)

Measure type
(dependability-
or performance-related)

Measure extent
(comprehensive or specific)

Assessment method
(experimentation
or modeling & experimentation)

Experimentation

System Under Benchmark - SUB

Workload

Faultload

Measurements

Procedures & rules

DBench
DependabilityBenchmarking

Benchmark developed

- General-purpose operating systems
  - Robustness and timing measures, TPC-C Client, faulty application
- Real-Time Kernels in onboard space systems
  - Predictability of the kernel response time, faulty application
- Engine control applications in automotive systems
  - Robustness of the control application, transient hardware faults
- On-line transaction processing (OLTP) systems,
  - TPC-C-based, Operator, software & hardware faults
    - TPC-C like measures
    - Measures based on modeling & experimentation: availability, cost
- Web servers

DBench
DependabilityBenchmarking

Properties

- Representativeness
- Repeatability
- Reproducibility
- Portability
- Non-intrusiveness
- Scalability
- Cost effective
  - Set-up
  - Execution duration

DBench
DependabilityBenchmarking

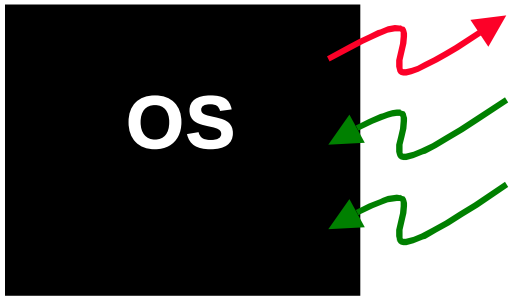
OS Benchmarking

- Integrator of a system including an operating system (OS)
  - Information on OS dependability
  - Select the most appropriate OS / system characteristics
  - Publishable results
- Objectives of OS dependability benchmarking
  - Provide generic and reproducible methods
    - Characterize the OS behavior in the presence of faults
    - Compare alternative solutions

DBench
DependabilityBenchmarking

OS Benchmarking Context

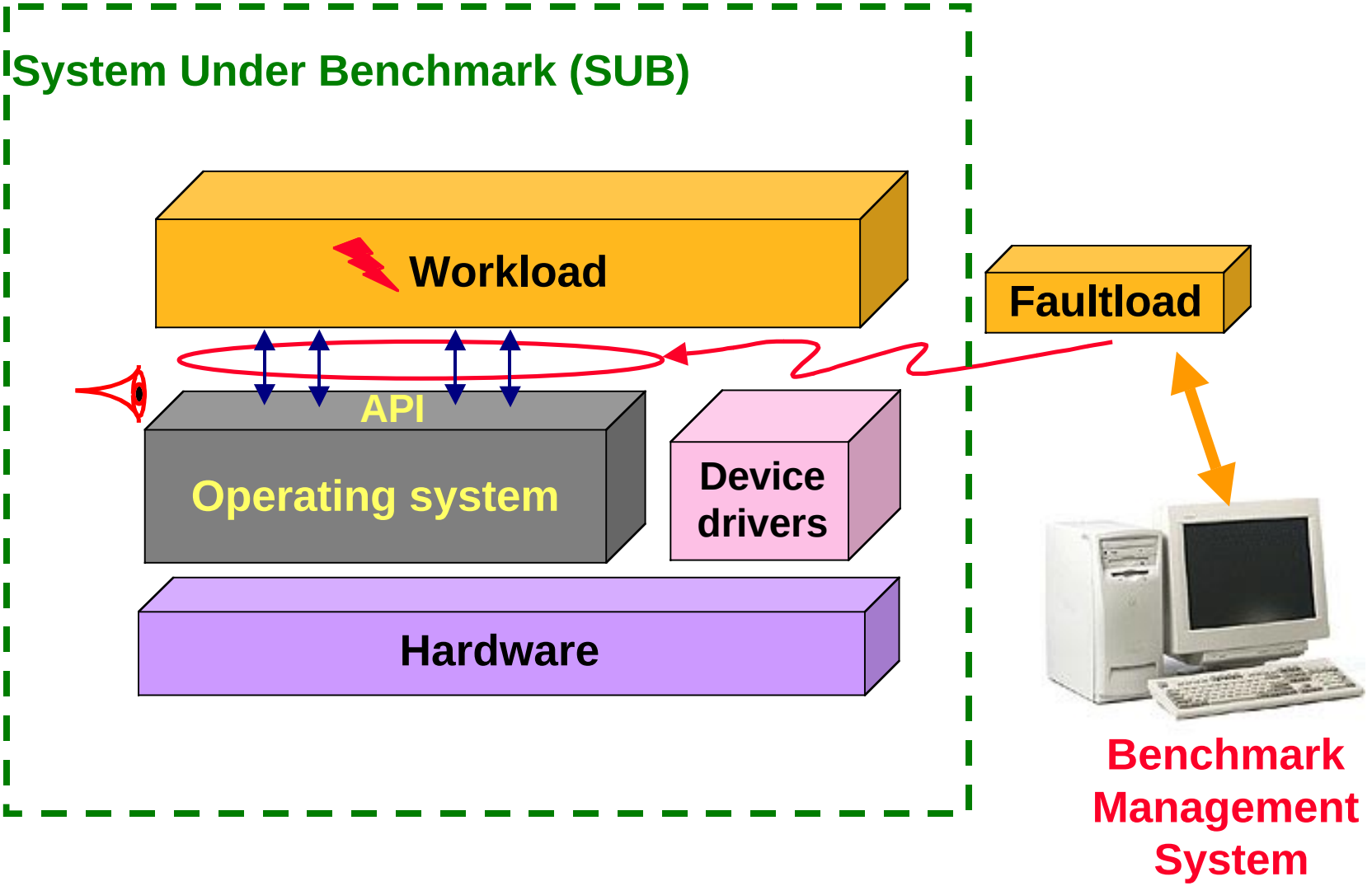
- Limited knowledge about the OS



- Functional description of the OS
- Non-intrusiveness
  - Faults injected outside the OS
  - Accessibility and observability

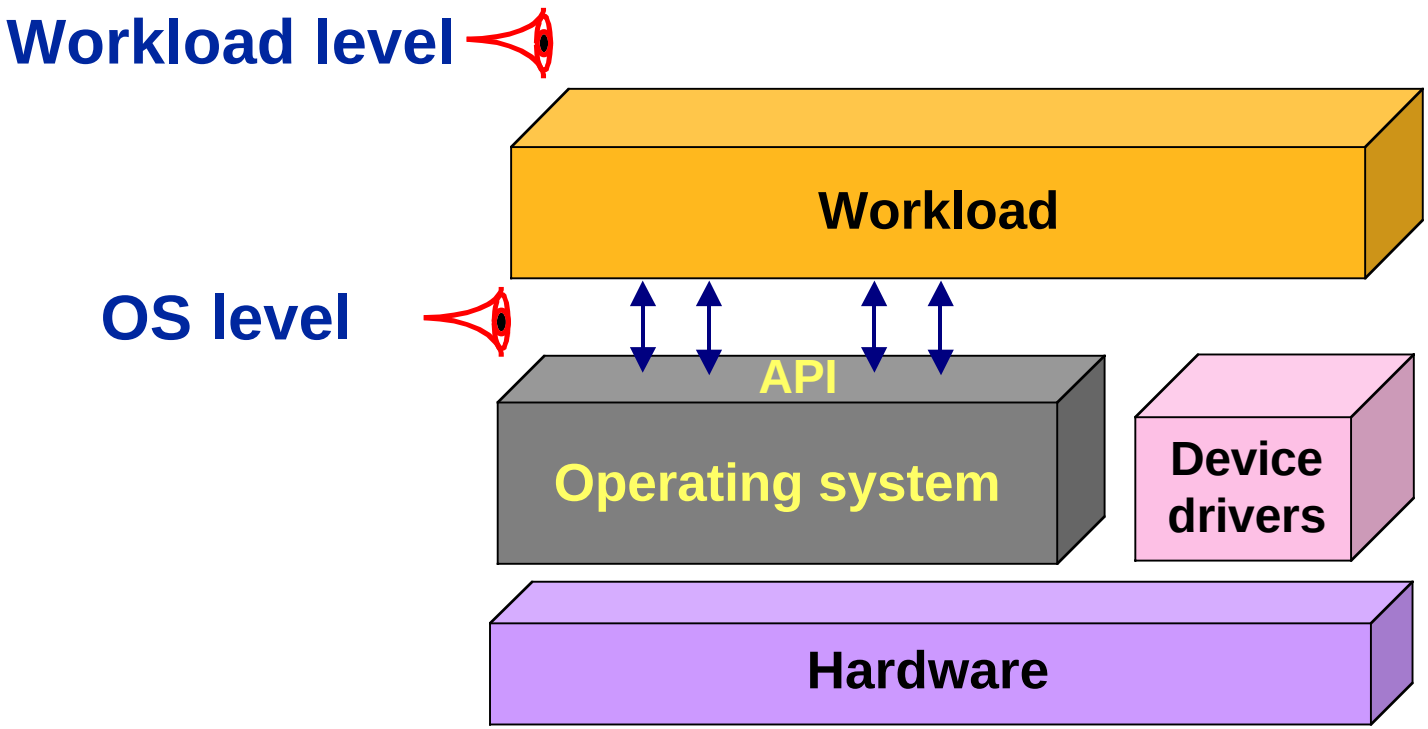
DBench
DependabilityBenchmarking

Benchmark Target & SUB



DBench
Dependability Benchmarking

Benchmark Measures



OS Level Measures

OS Outcomes

SEr	Error code
SXp	Exception
SPc	Panic
SHg	Hang
SNS	No signaling

Measures

- POS: OS Robustness (outcome distribution)
- Texec: OS reaction time in the presence of faults
- Tres: OS Restart time in the presence of faults

DBench
DependabilityBenchmarking

Workload Level Measures

Workload Outcomes

- WCC Correct completion
- WEC Erroneous completion
- WAb Abort
- WHg Hang

Workload Level Measures

Combined states

OS / Workload	Error code	Exception	Panic	Hang	No signalling
Correct completion	SEr-WCC	SXp-WCC	—	—	SNS-WCC
Erroneous completion	SEr-WEC	SXp-WEC	—	—	SNS-WEC
Abort	SEr-WAb	SXp-WAb	SPc-WAb	—	SNS-WAb
Hang	SEr-WHg	SXp-WHg	SPc-WHg	SHg-WHg	SNS-WHg

Workload Measures

- PSNS: WL Robustness (WL outcome distribution)
- TWL: WL completion time in the presence of faults

DBench DependabilityBenchmarking

Measure Summary

❑OS Measures

- ✦POS: OS Robustness
- ✦Texec: reaction time in the presence of fault (τexec: in absence of faults)
- ✦Tres: restart time in the presence of faults (τres: in absence of faults)

❑Workload Measures

- ✦PSNS: WL Robustness, when OS in SNS
- ✦TWL: WL correct completion time in the presence of faults

  (τWL: in absence of faults)

DBench
DependabilityBenchmarking

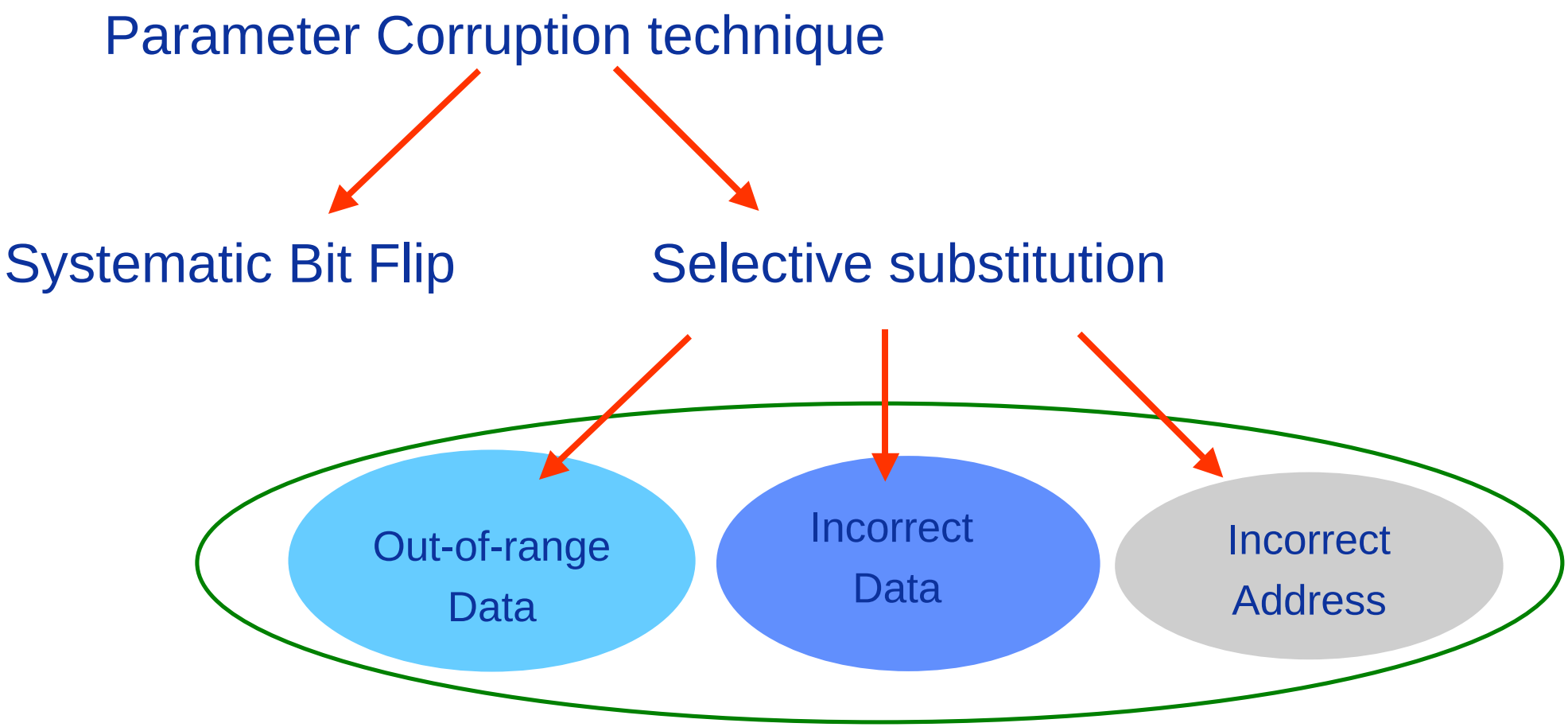
Execution profile

- Workload: TPC-C Client in the current prototype
- Faultload
  - Selection of system calls to be corrupted
    - Ideally: all system calls with parameters
    - In practice: most critical OS functional components

      Processes and Threads, File Input/output,

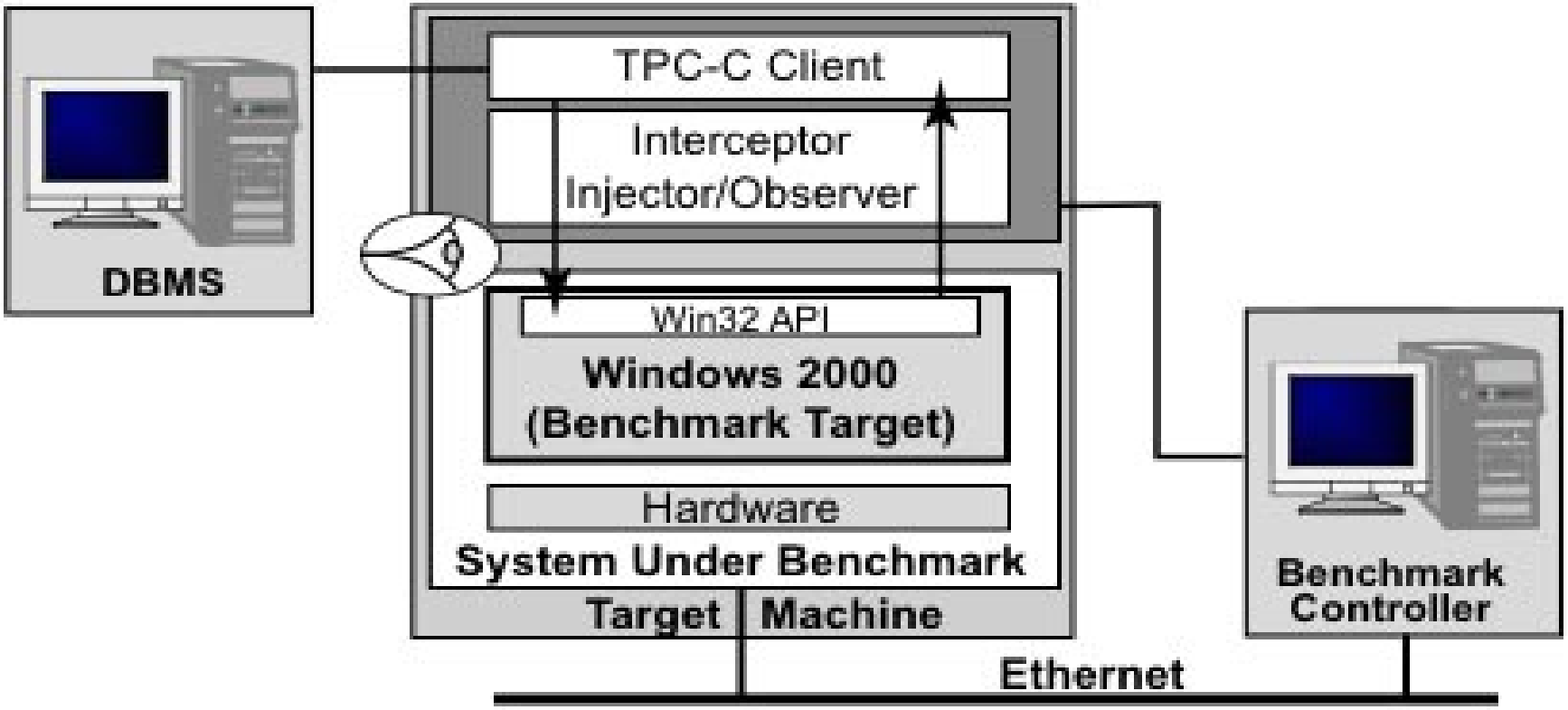
      Memory management, Configuration Management
    - 28 system calls, 75 parameters, 502 corrupted values
  - Interception of the selected system calls
  - Parameter corruption technique: selective substitution

DBench
DependabilityBenchmarking



DBench
DependabilityBenchmarking

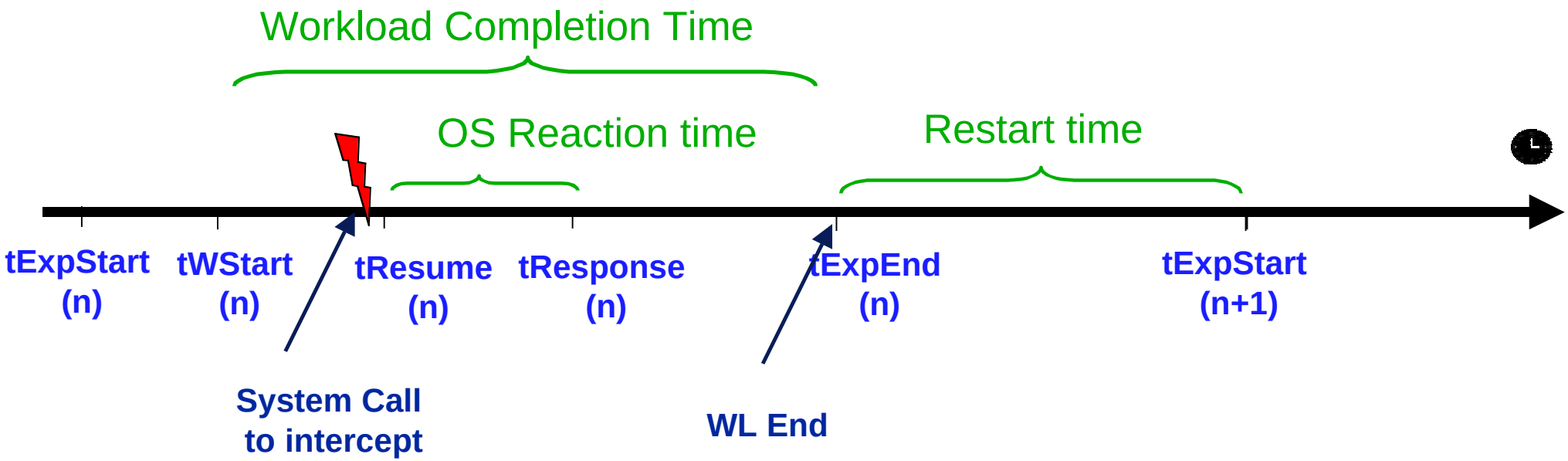
Experimental Set-up



DBench
DependabilityBenchmarking

Measurements

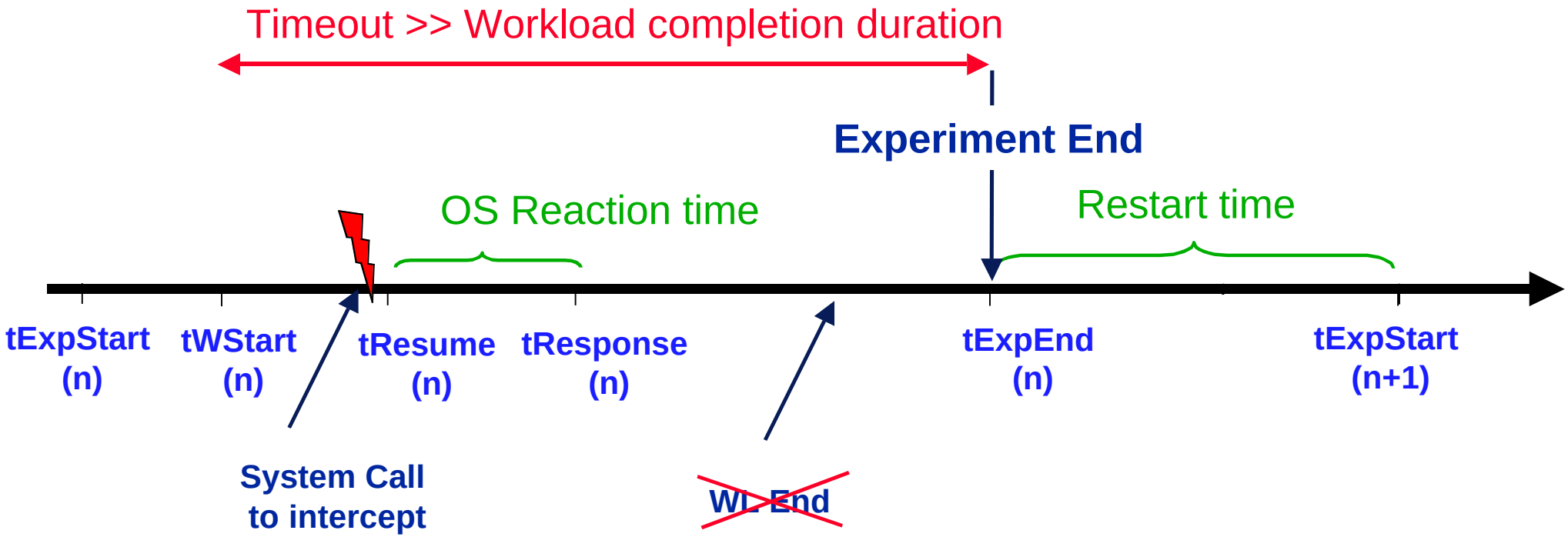
Experiments with Workload (WL) completion



DBench
DependabilityBenchmarking

Measurements

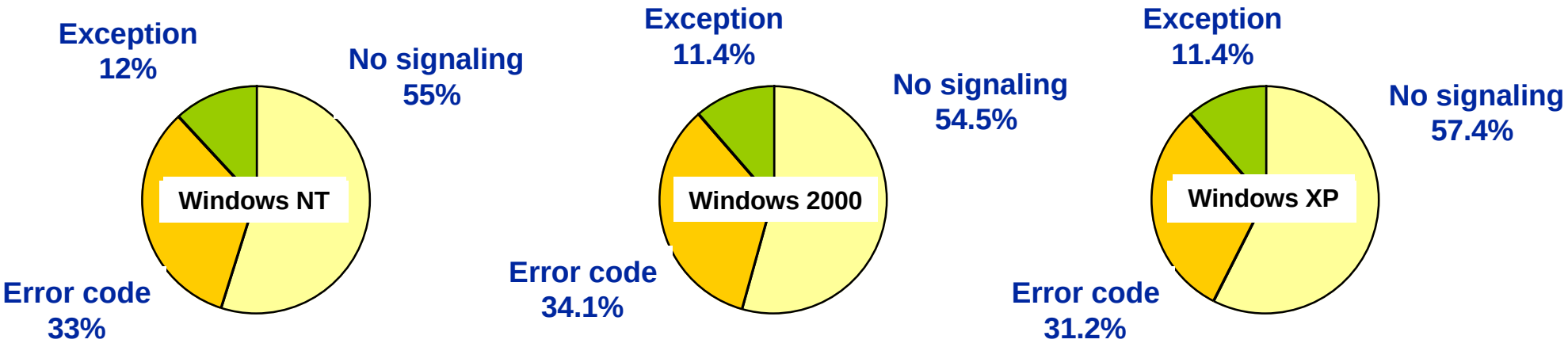
Experiments without Workload (WL) completion



DBench DependabilityBenchmarking

Results: OS Robustness

POS



28 system calls intercepted, 552 experiments

DBench DependabilityBenchmarking

Sensitivity Analysis wrt Faultload

	Incorrect data	Incorrect address	Out-of-range data	Systematic Bit-Flip	# System calls	# experiments
FL0	x	x	x		28	552
FL1		x	x		28	325
FL2			x		28	113
FL3				x	28	2400
FL4			x		All (132)	353

DBench
DependabilityBenchmarking

Workload States

Windows NT ↓WL	
Completion	(451)
Abort / Hang	(101)

Windows 2000 ↓WL	
Completion	(445)
Abort / Hang	(107)

Windows XP ↓WL	
Completion	(424)
Abort / Hang	(128)

DBench
DependabilityBenchmarking

Refinement of Workload States

↓PSNS

Windows NT ↓WL	Error code (182)	Exception (66)	No signaling (304)
Completion (451)	136	58	257
Abort / Hang (101)	46	8	47

Windows 2000 ↓WL	Error code (188)	Exception (63)	No signaling (301)
Completion (445)	136	57	252
Abort / Hang (107)	52	6	49

Windows XP ↓WL	Error code (172)	Exception (63)	No signaling (317)
Completion (424)	99	57	268
Abort / Hang (128)	73	6	49

DBench
DependabilityBenchmarking

OS Reaction Time

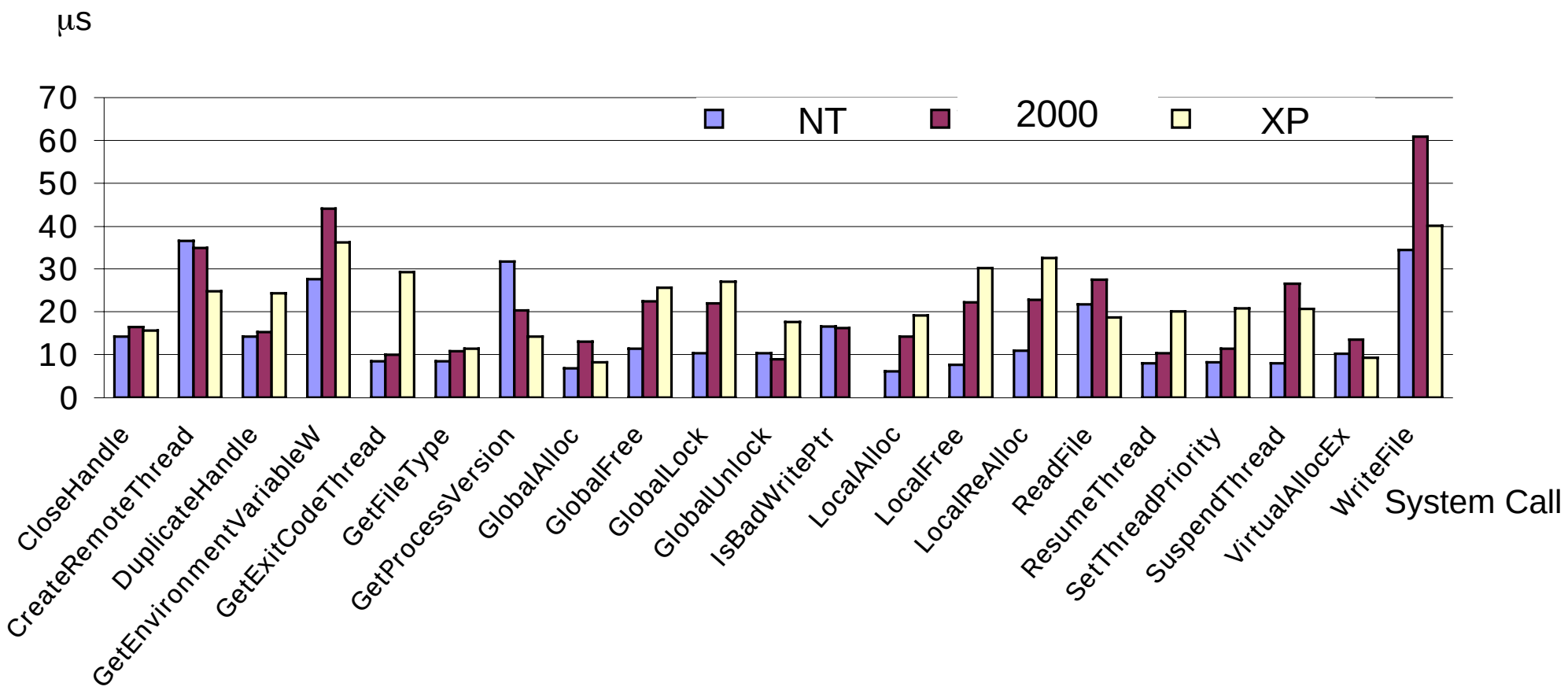
	τexec	Texec (Std dev.)
Windows NT	344 μs	128 μs (230 μs)
Windows 2000	1782 μs	1241 μs (3359 μs)
Windows XP	111 μs	114 μs (176 μs)

Texec Error code	Texec Exception	Texec No-signaling
17 μs (18 μs)	86 μs (138 μs)	203 μs (281)
22 μs (28 μs)	973 μs (2978 μs)	2013 μs (4147)
23 μs (17 μs)	108 μs (162 μs)	165 μs (204 μs)

DBench
DependabilityBenchmarking

Detailed OS Reaction Time

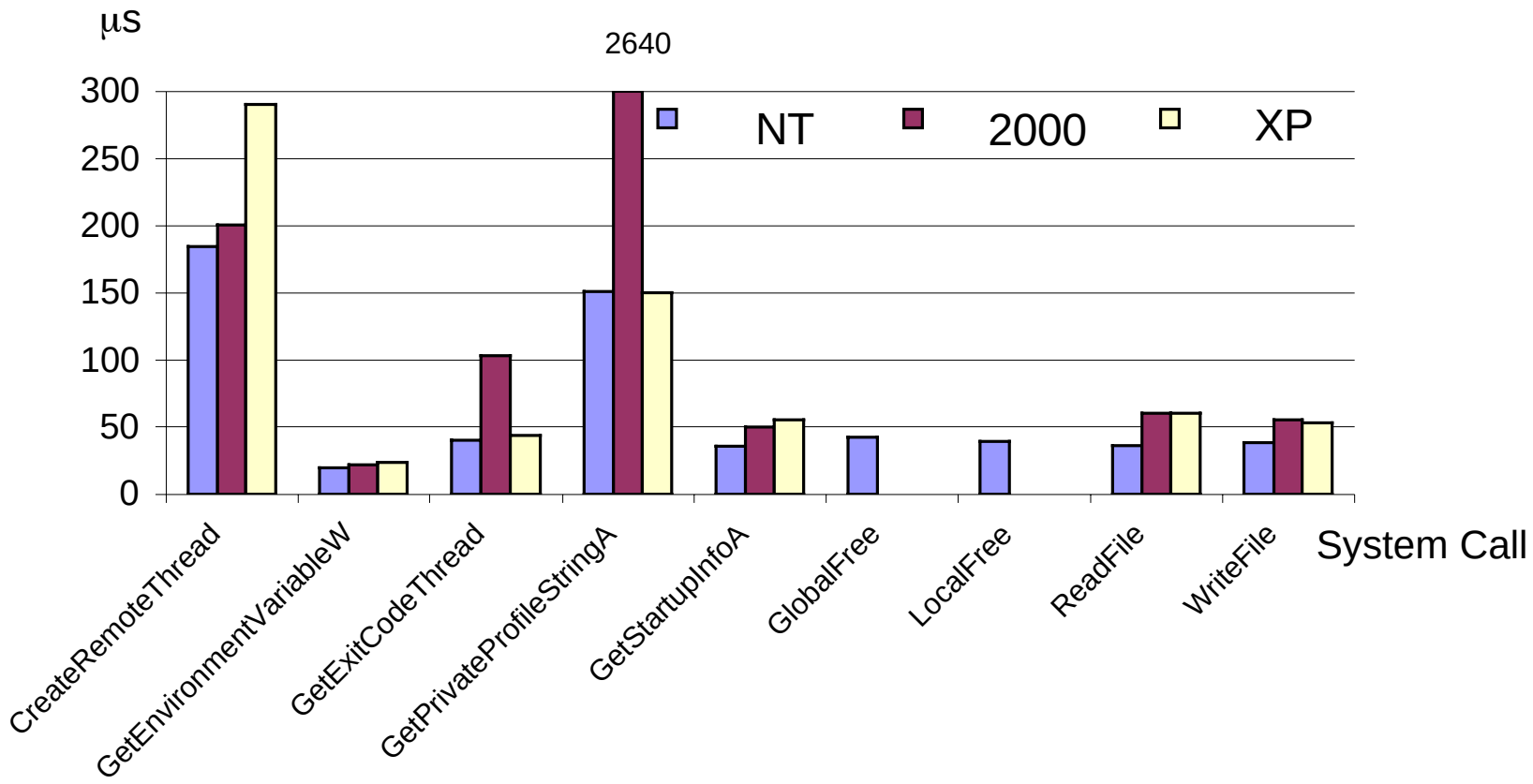
Error Code return



DBench
DependabilityBenchmarking

Detailed OS Reaction Time

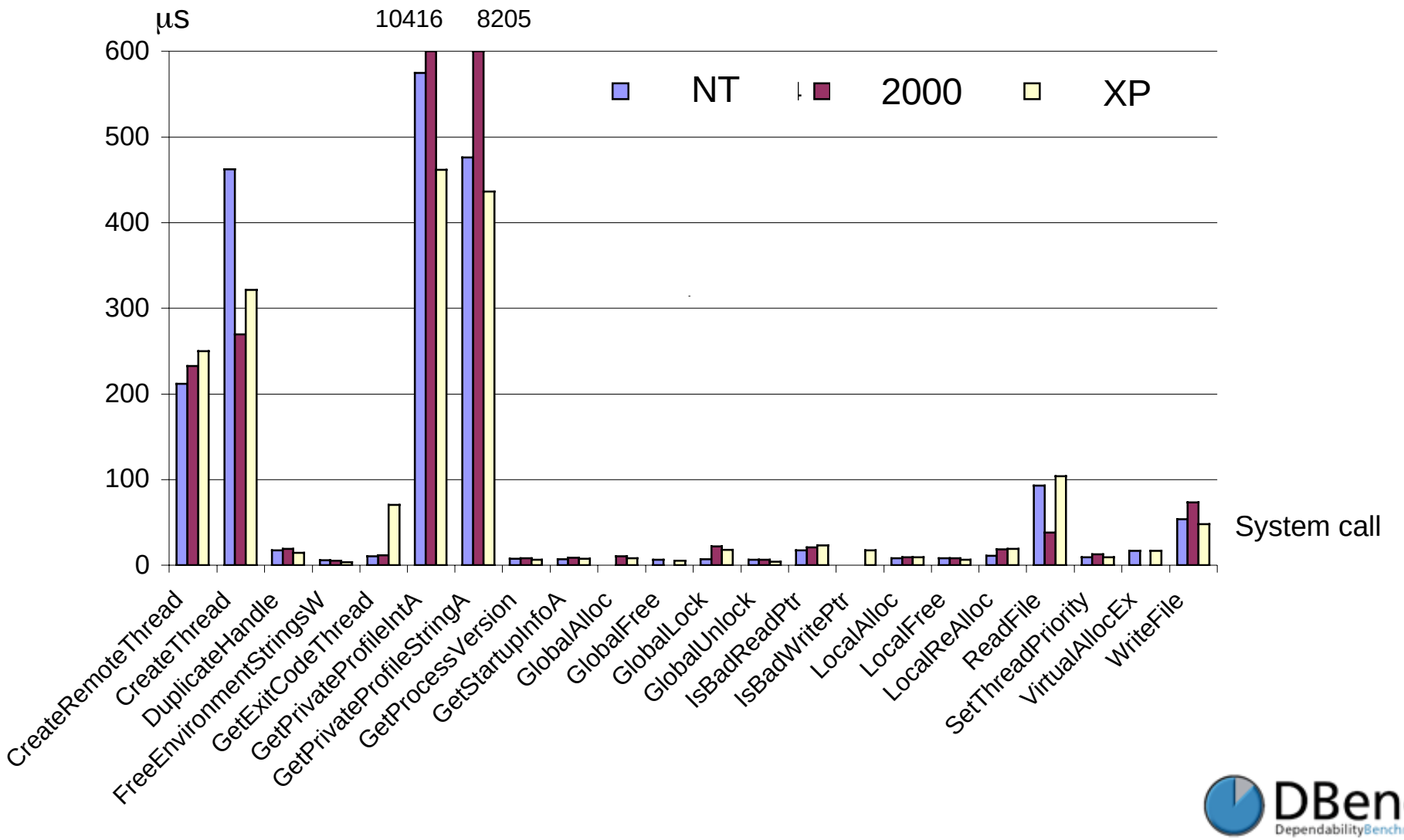
Exception Notification



DBench
DependabilityBenchmarking

Detailed OS Reaction Time

No-Signaling



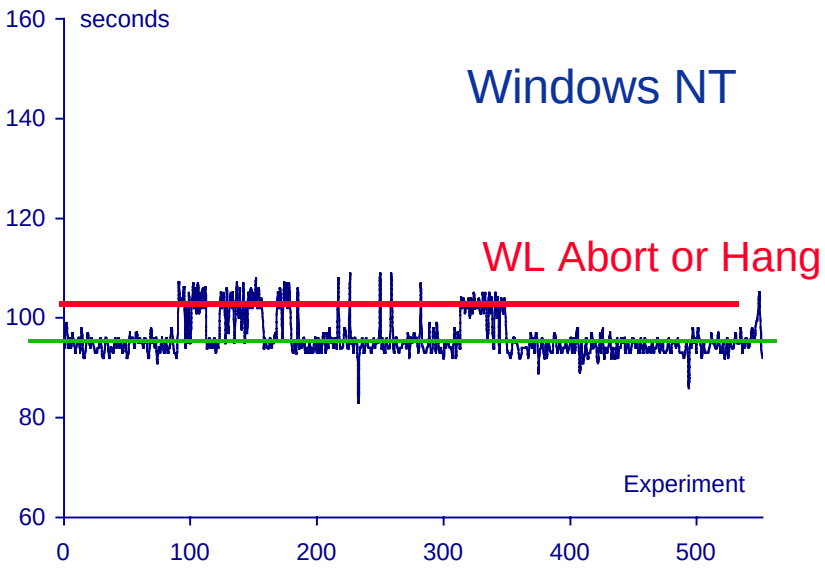
DBench
DependabilityBenchmarking

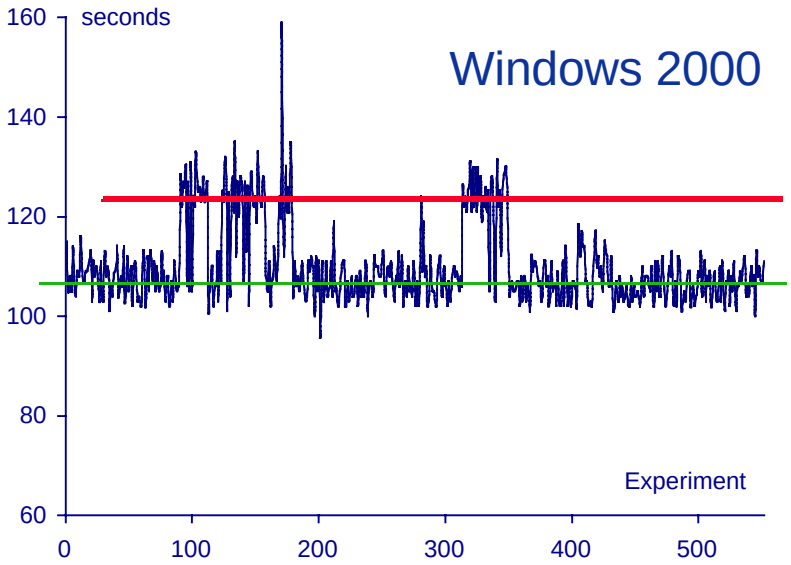
OS Restart Time

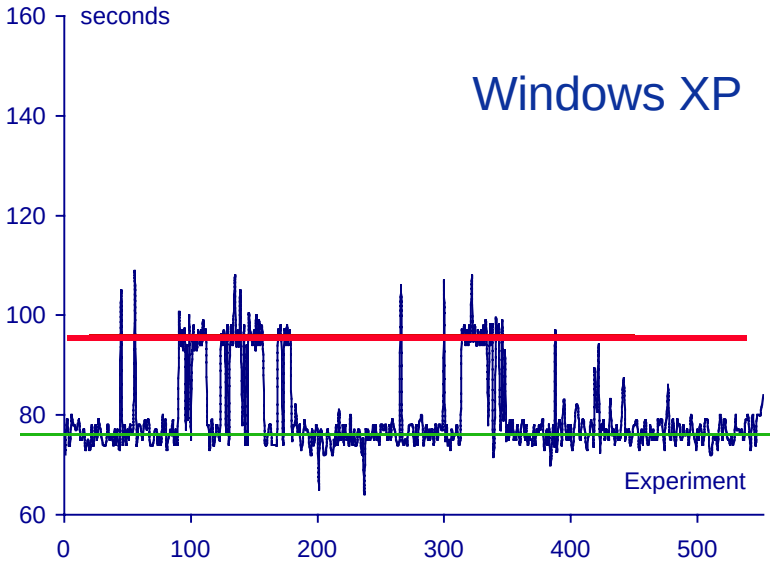
	τres	Tres	Std Deviation
Windows NT	92 s	96 s	4 s
Windows 2000	105 s	109 s	8 s
Windows XP	74 s	80 s	8 s

DBench
DependabilityBenchmarking

Detailed OS Restart Time







DBench
DependabilityBenchmarking

WL Execution Time

	τWC	TWC	Std Deviation
Windows NT	74 s	80 s	12 s
Windows 2000	70 s	74 s	13 s
Windows XP	67 s	69 s	10 s

DBench
DependabilityBenchmarking

Conclusion

- OS robustness benchmark wrt application erroneous behavior
- Dependability benchmark prototype for Windows family
- Novelty
  - Structured set of measures
  - Realistic Workload: TPC-C Client
  - Standard experimental procedures and rules
  - Benchmark properties
  - Benchmark execution duration: 2 days

DBench
DependabilityBenchmarking

- Validation of the benchmark
  - Results in conformance with Microsoft claim
  - Sensitivity study wrt to parameter corruption technique
  - Sensitivity study wrt system calls corrupted
  - Benchmark properties
- Current work
  - Other OS family: Linux
  - Other workload