TRUSTWORTHINESS BENCHMARKING OF (SAFETY) CRITICAL SYSTEMS OR TO BENCHMARK OR NOT TO BENCHMARK

SAFECOMP 2019
Turku, Finland
Sep. 11th, 2019

Marco Vieira
mvieira@dei.uc.pt
Department of Informatics Engineering
University of Coimbra - Portugal

Setting the Expectations…

- What to talk about?
  - Established work?
    - Benchmarking…
  - New ideas?
    - Trustworthiness…
  - Somewhere in the between?
    - Trustworthiness Benchmarking ☺

- Non-established concepts and ideas ahead!
  - Mostly ongoing work

OUTLINE

- Trustworthiness as a Broad Concept
- Benchmarking: Past and Present
- From Security to Trustworthiness Benchmarking
- Trustworthiness Benchmarking Framework
- Challenge: Trustworthiness Benchmarking in Safety Critical Systems
- Conclusions

TRUST & TRUSTWORTHINESS

- Concepts broadly studied in many different areas
  - Sociology, economics, psychology…
- Trust:
  - Reliance on a system or service that it will exhibit the expected behavior (including many perspectives)
  - Trust Level: estimated probability of this reliance
- Trustworthiness
  - Worthiness of a system or service for being trusted
  - Assessed based on evidences
  - Complex and potentially subjective!

TRUST & TRUSTWORTHINESS

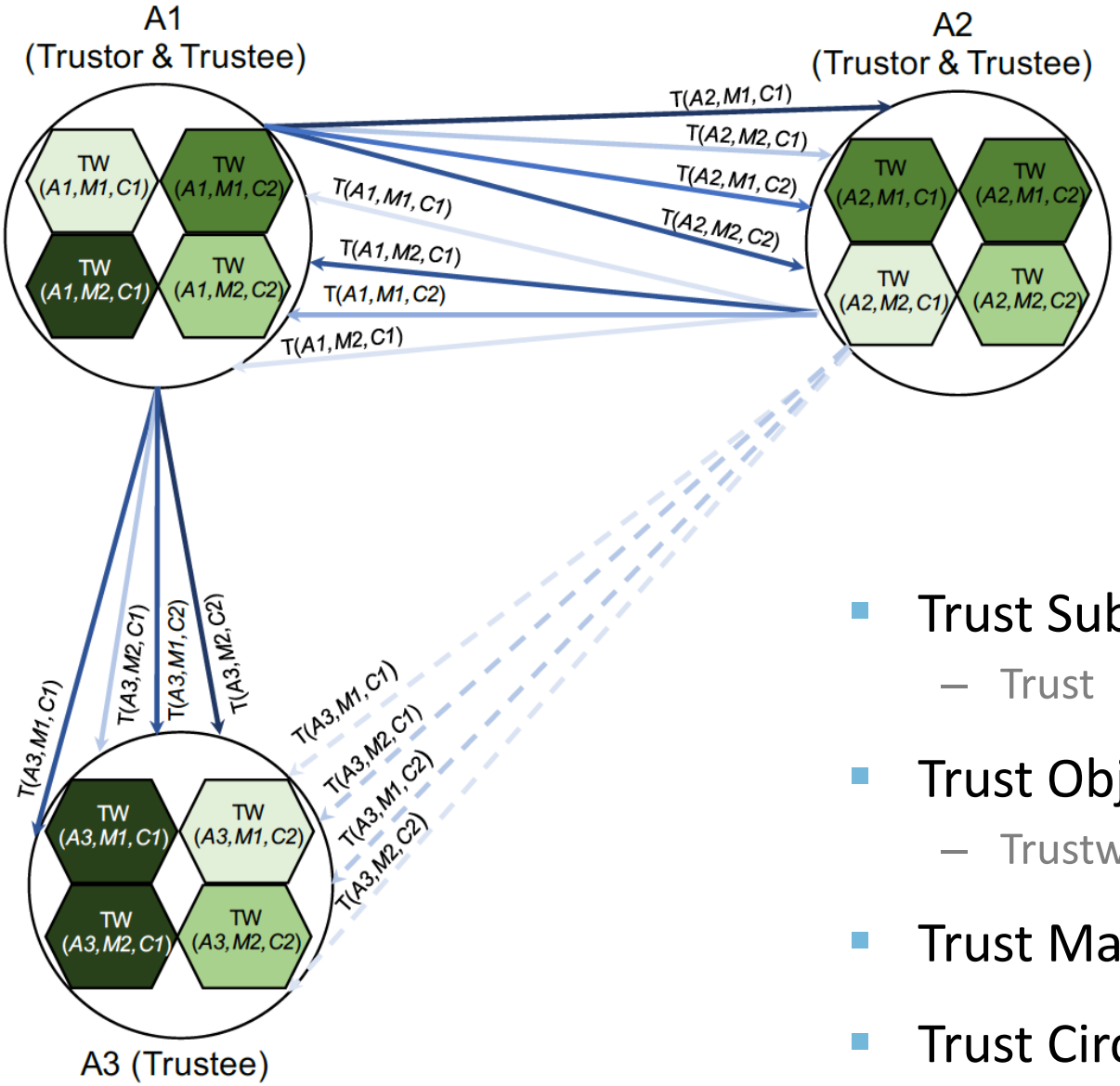
- Human Trust and Trustworthiness

- Changes over time and can be highly subjective

TRUSTWORTHINESS PROPERTIES

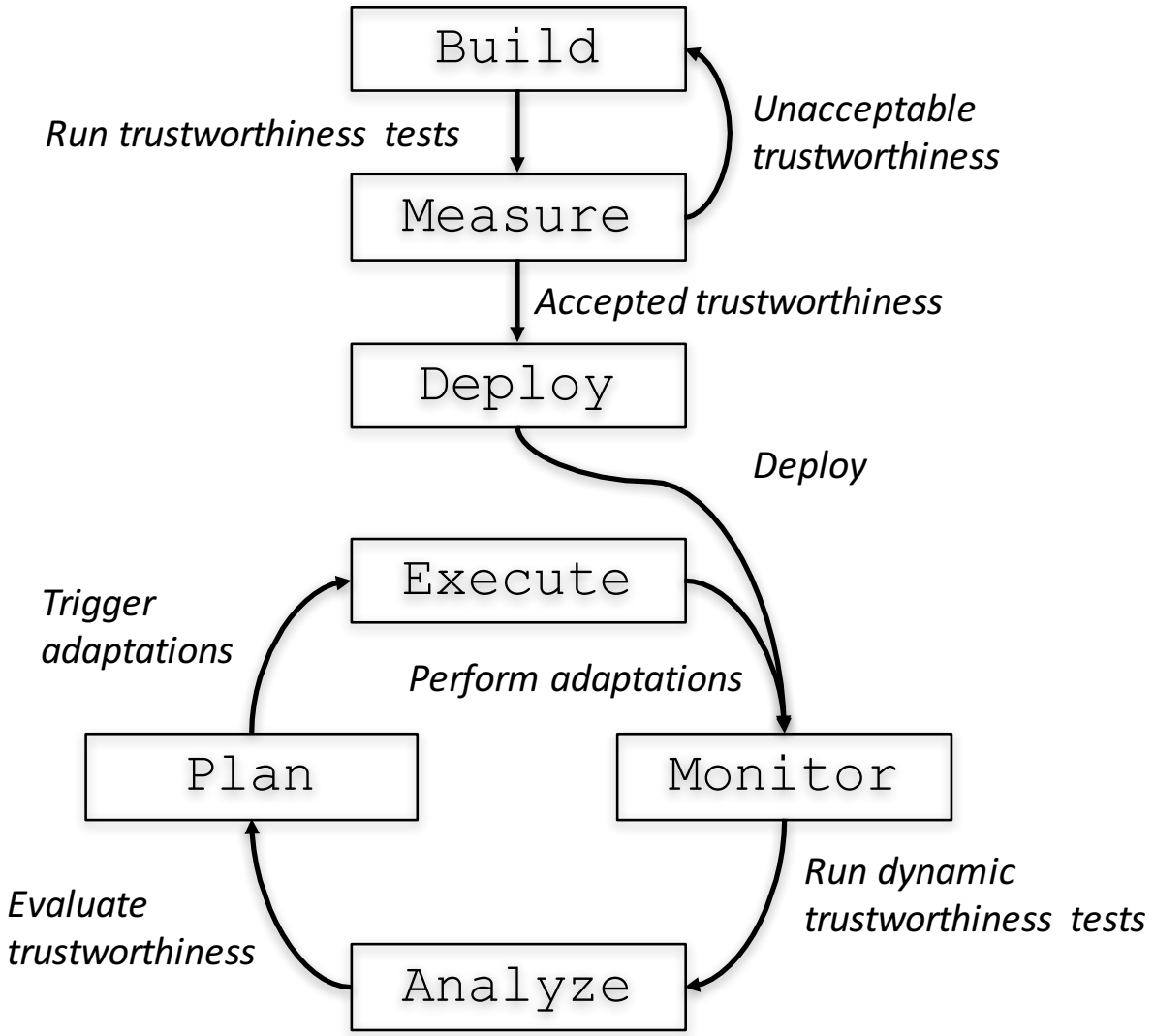
- Trustworthiness is frequently seen as a security aspect
  - It is trustworthy if it is secure!?
- We consider it a more general notion
  - Even broader than dependability…
- Requires identifying and evaluating all relevant measurable characteristics that may influence reliance
  - Functional and non-functional
- Security, privacy, dependability, performance, fairness, transparency, stability…
  - Just define them as needed!

TRUST RELATIONSHIP



- Trust Subject: Trustor
  - Trust
- Trust Object: Trustee
  - Trustworthiness
- Trust Matter: Functional
- Trust Circumstances: Non-functional

TRUSTWORTHINESS LIFE-CYCLE



OUTLINE

- Trustworthiness: An Integrative Concept
- Benchmarking: Past and Present
- From Security to Trustworthiness Benchmarking
- Trustworthiness Benchmarking Framework
- Challenge: Trustworthiness Benchmarking in Safety Critical Systems
- Conclusions

BENCHMARKING

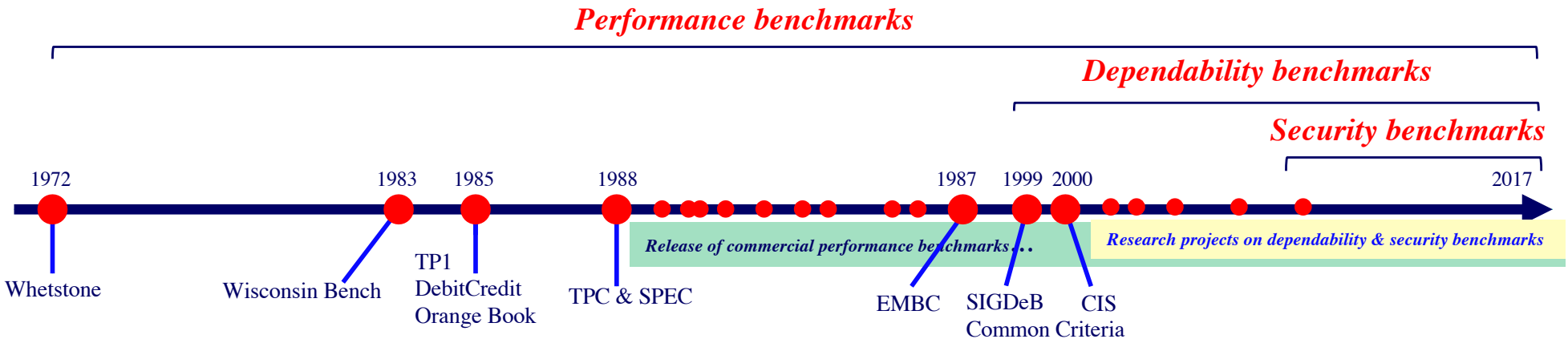
Assessing and comparing computer systems and/or components according to specific quality attributes

- Performance benchmarking
  - Well established both in terms of research and application
  - Supported by organizations like TPC and SPEC
  - Mostly for marketing
- Dependability benchmarking
  - Well established from a research perspective
  - No endorsement from the industry

BENCHMARKING

Assessing and comparing computer systems and/or components according to specific quality attributes

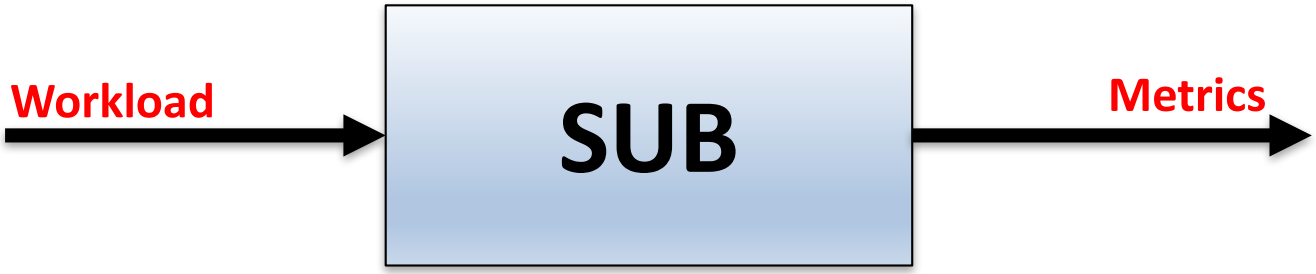
- Security benchmarking
  - Several works can be found
  - No common approach available yet



PERFORMANCE BENCHMARKING

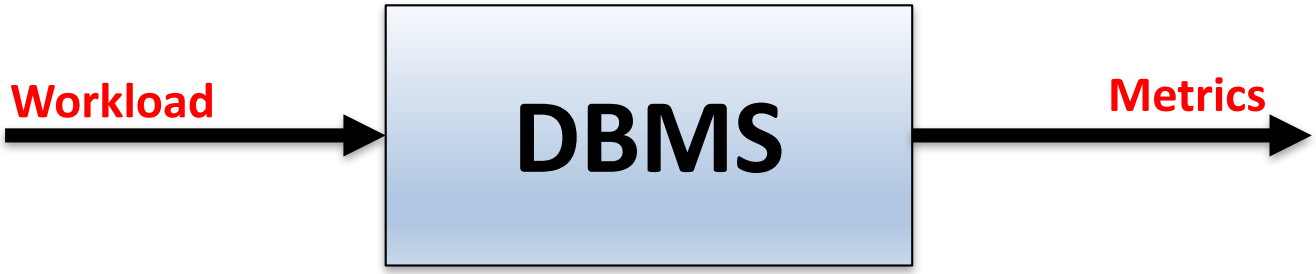
Assessing and comparing
computer systems and/or components
in terms of performance

PERFORMANCE BENCHMARKING



- Workload:
  - Set of representative operations
- Metrics:
  - Throughput
  - Response time
  - Latency
  - …

TPC-C (1992)

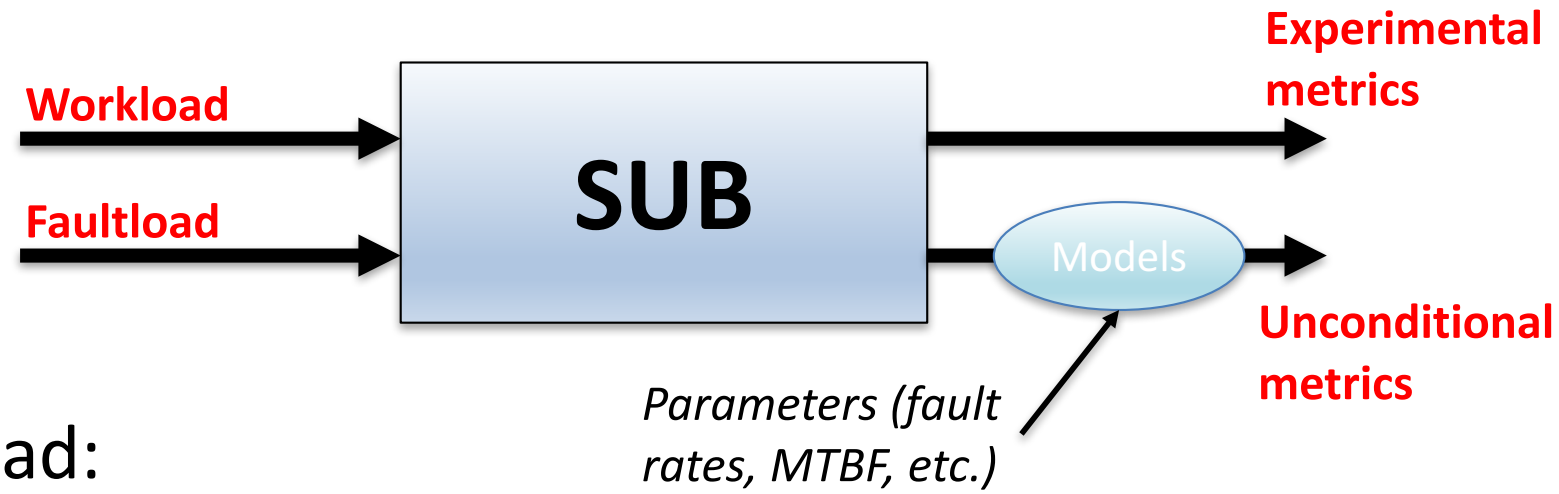


- Workload:
  - Database transactions
- *Although some integrity tests are performed, it assumes that nothing fails*
  - Transaction rate (tpmC)
  - Price per transaction ($/tpmC)

DEPENDABILITY BENCHMARKING

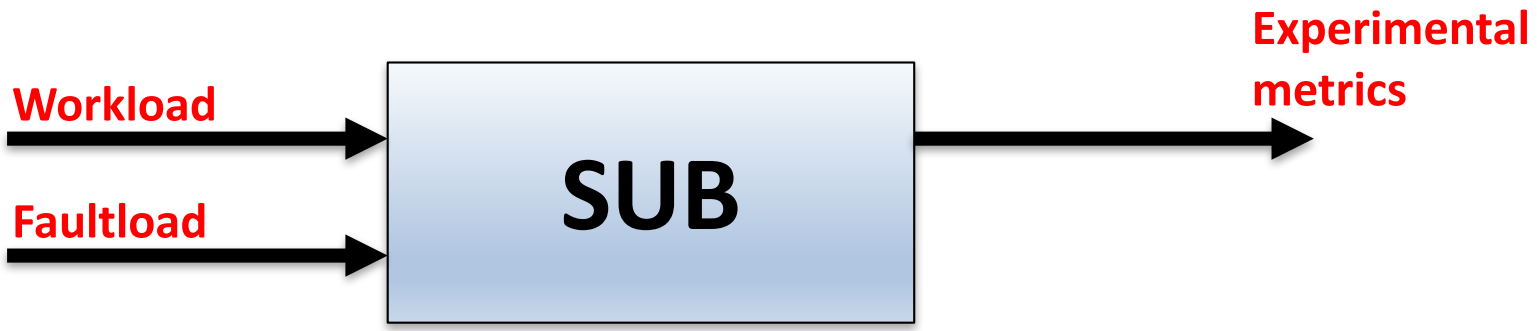
Assessing and comparing
computer systems and/or components
considering dependability attributes

DEPENDABILITY BENCHMARKING



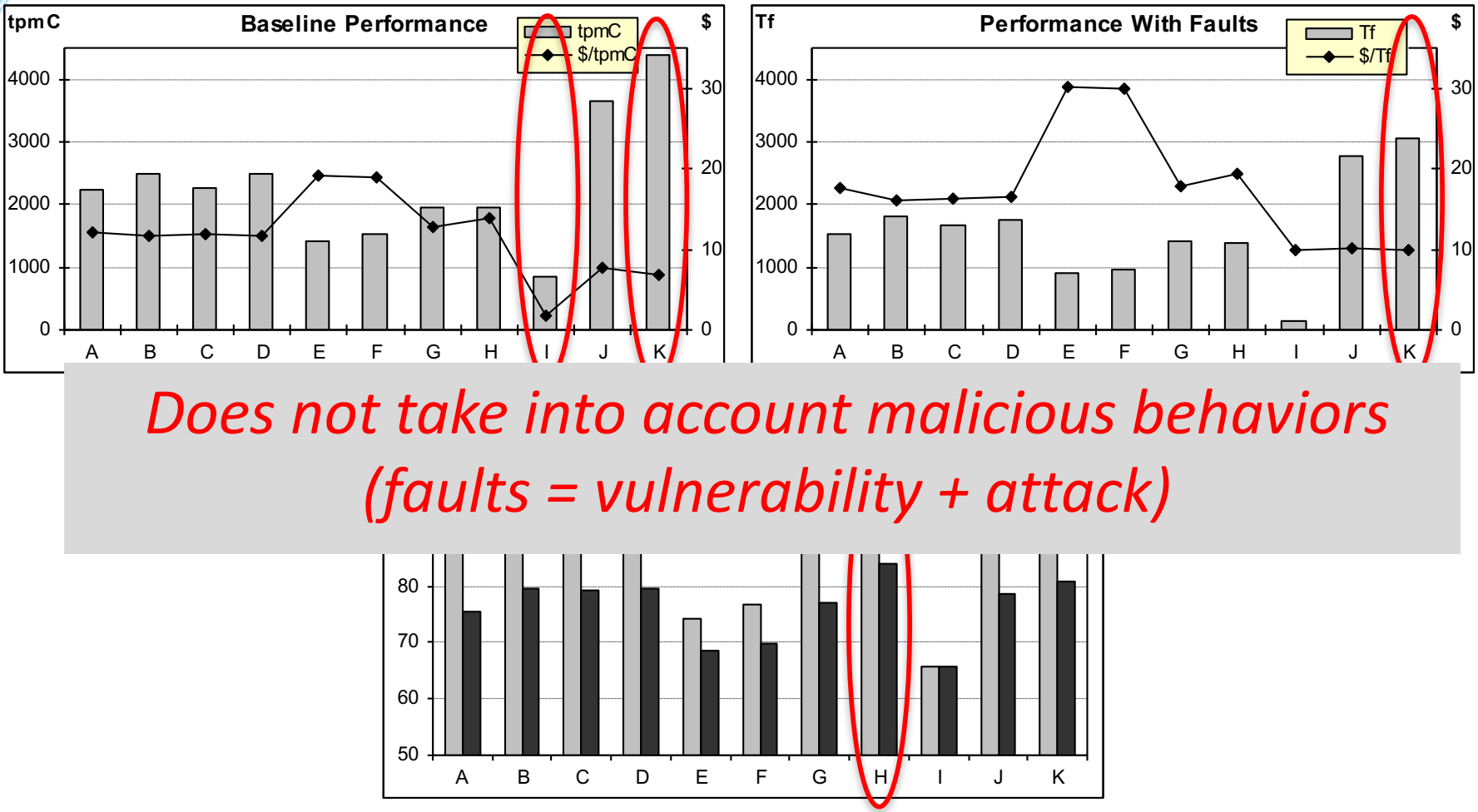
- Faultload:
  - Set of representative faults, injected into the system
- Metrics:
  - Performance and/or dependability
    - Both baseline and in the presence of faults
  - Unconditional and/or direct

DBENCH-OLTP (2005)



- Workload:
  - TPC-C transactions
- Faultload:
  - Operator faults + Software faults + HW component failures
- Metrics:
  - Performance: tpmC, $/tpmC, Tf, $/Tf
  - Dependability: Ne, AvtS, AvtC

DBENCH-OLTP (2005)



Does not take into account malicious behaviors (faults = vulnerability + attack)

OUTLINE

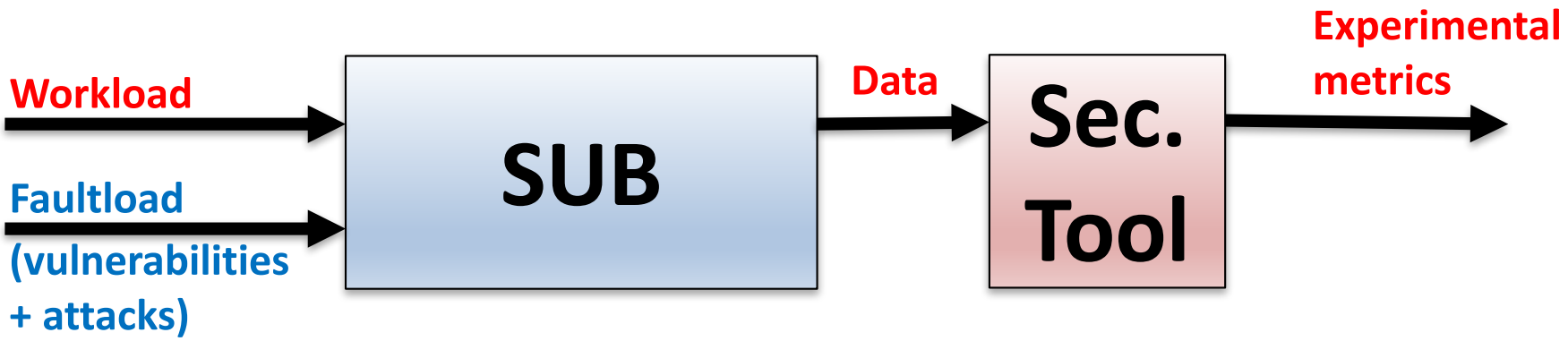
- Trustworthiness: An Integrative Concept
- Benchmarking: Past and Present
- From Security to Trustworthiness Benchmarking
- Trustworthiness Benchmarking Framework
- Challenge: Trustworthiness Benchmarking in Safety Critical Systems
- Conclusions

SECURITY BENCHMARKING

Assessing and comparing
computer systems and/or components
considering security aspects

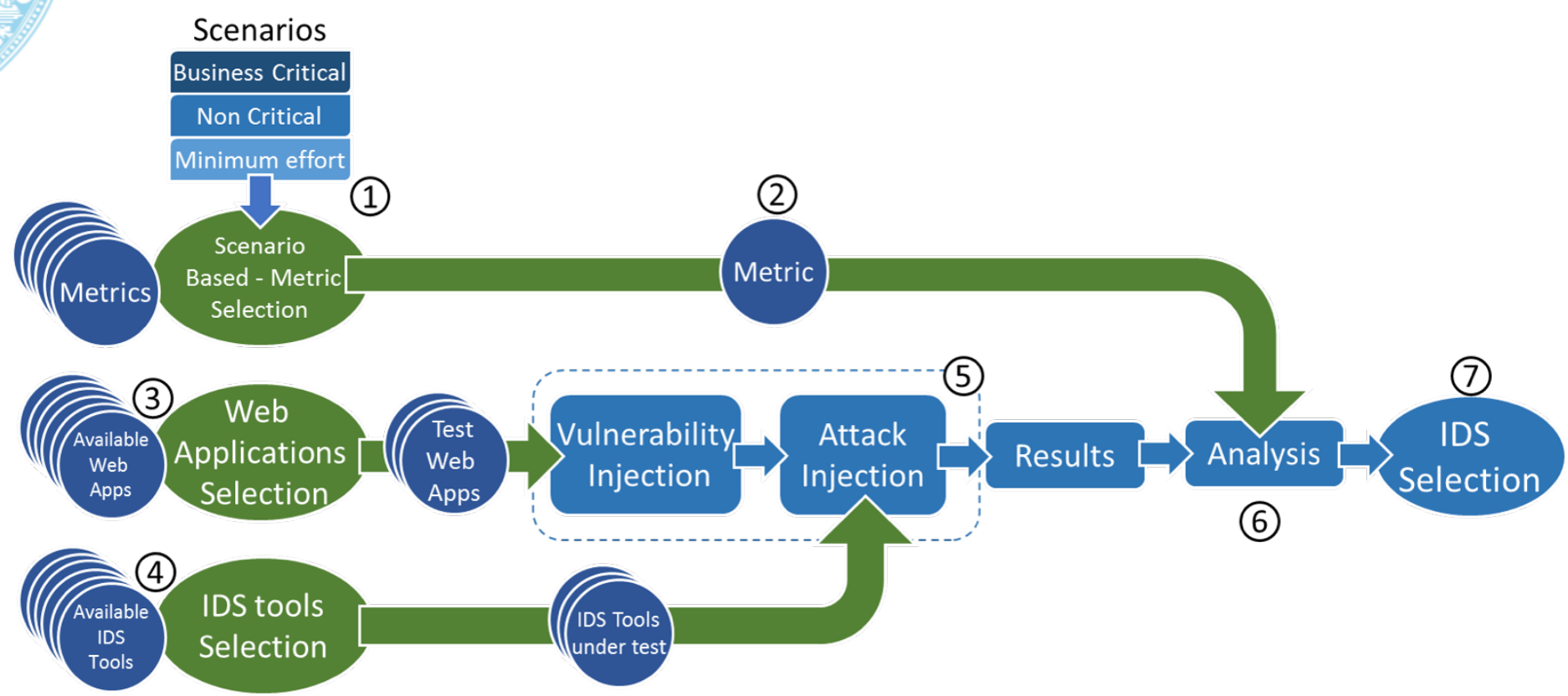
- Benchmarking **Security Tools**
  - Tools used to improve the security of systems
  - Penetration testers, static analyzers, IDS, etc.
- Benchmarking the Security of **Systems / Components**
  - Systems that should implement security requirements
  - OS, middleware, server software, etc.

BENCHMARKING SECURITY TOOLS



- Faultload:
  - Vulnerabilities are injected
  - Attacks target the injected vulnerabilities
- Data can be collected for benchmarking security tools
  - Penetration testers, static analyzers, IDS, etc.

EXAMPLE: BENCHMARKING IDS

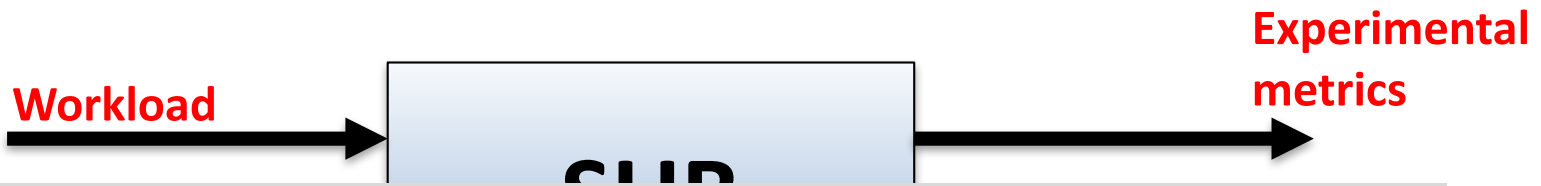


Main Results

All

lvl	Tool	Review			Reported				Prec.	Recall	Mark.	Infor.
		P	N	Pop	TP	TN	FN	FP				
App	ACD	1051	224	1275	376	174	675	50	0.883	0.358	0.088	0.135
App	Scalp			1275	206	224	845	0	1.000	0.196	0.210	0.196
App	ModSecurity	826	225	1051	236	225	590	0	1.000	0.286	0.276	0.286
Net	Snort 2.8	458	817	1275	0	817	458	0	-	0.000	-	0.000
DB	GreenSQL			1275	244	813	214	4	0.984	0.533	0.775	0.528
DB	DB IDS			1275	451	384	7	433	0.510	0.985	0.492	0.455
Net	Snort 2.9	173	878	1051	0	878	173	0	-	0.000	-	0.000

BENCHMARKING SECURITY OF SYSTEMS



Attacking what? Do we know the vulnerabilities? What are representative attacks?

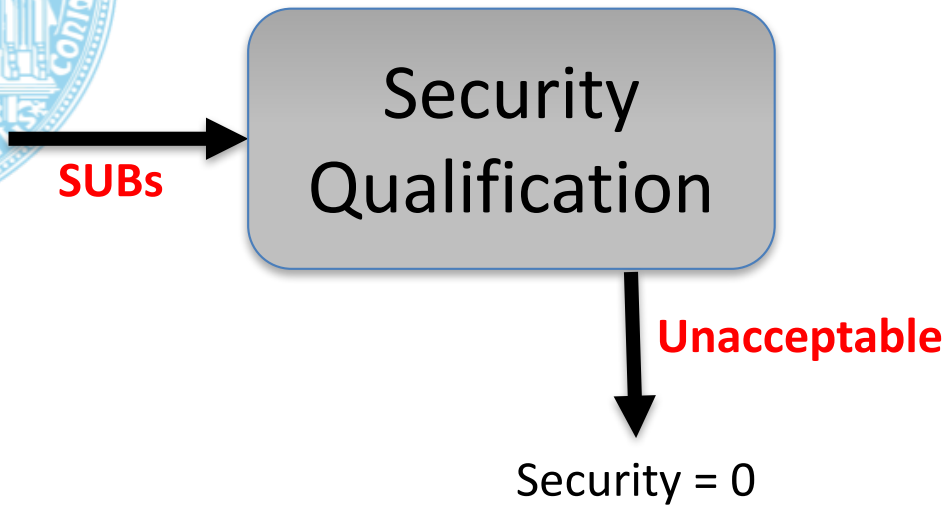
Does not work if one wants to benchmark how secure different systems are!

e.g. does the number of vulnerabilities of a system represent anything?

onal

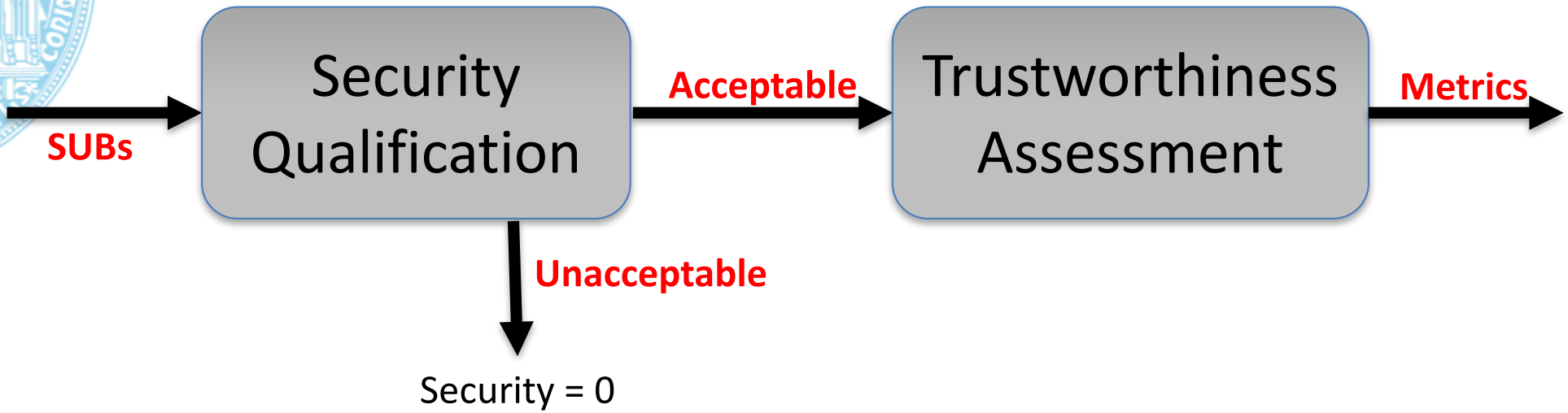
- Performance + dependability
- Security (e.g., number vulnerabilities, attack detection)

A DIFFERENT APPROACH…



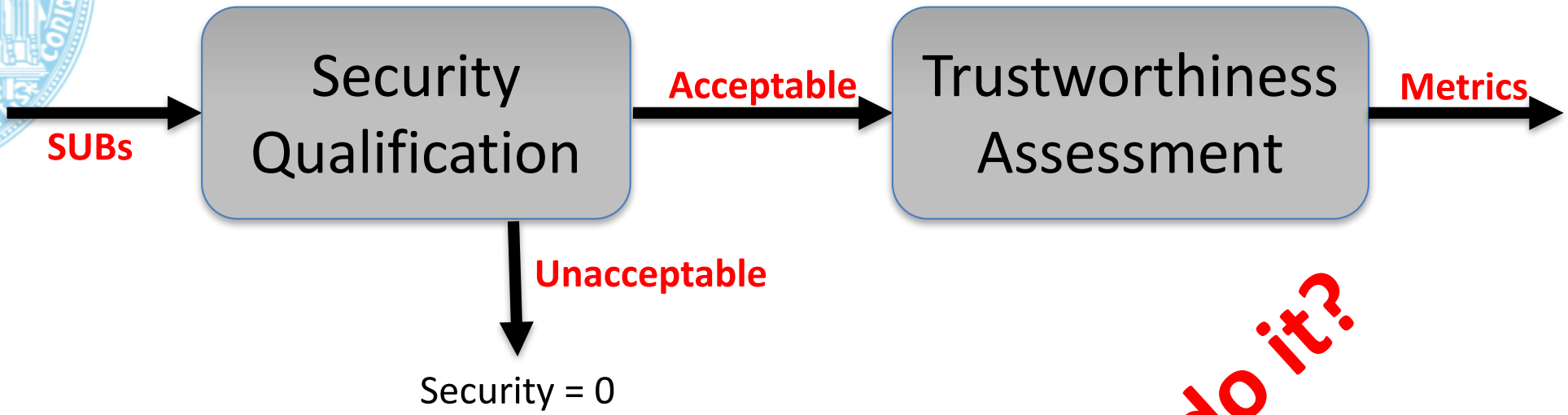
- Security Qualification:
  - Apply state-of-the-art techniques and tools to detect vulnerabilities
  - SUBs with vulnerabilities are:
    - Disqualified!
    - Or vulnerabilities are fixed…

A DIFFERENT APPROACH…



- Trustworthiness Assessment:
  - Gather evidences on how much one can trust
  - e.g., best coding practices, development process, bad smells

A DIFFERENT APPROACH…



How to do it?

- Metrics:
  - Portray trust from a user perspective
  - Dynamic: may change over time
  - Depend on the type of evidences gathered
  - Different metrics for different attack vectors

OUTLINE

- Trustworthiness: An Integrative Concept
- Benchmarking: Past and Present
- From Security to Trustworthiness Benchmarking
- Trustworthiness Benchmarking Framework
- Challenge: Trustworthiness Benchmarking in Safety Critical Systems
- Conclusions

What We Have…

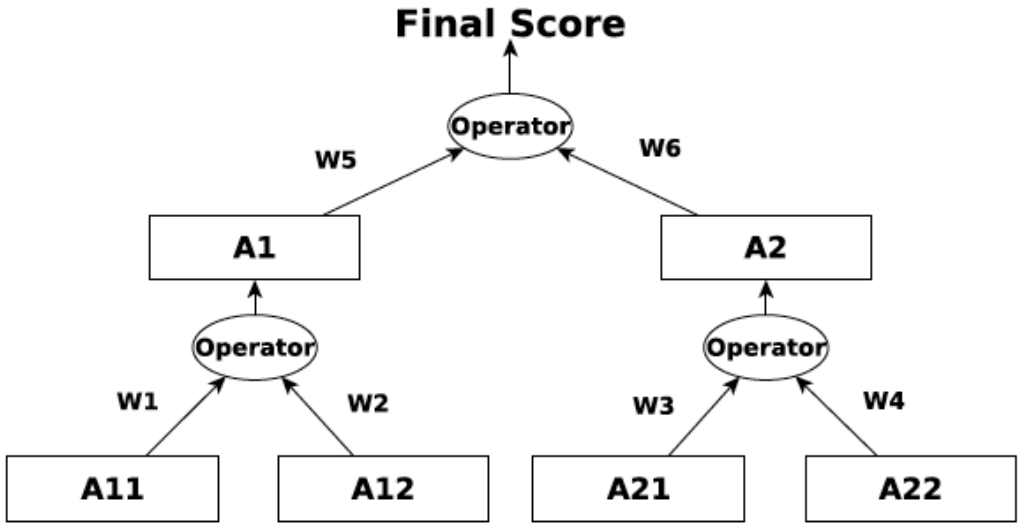
- Established benchmarks are mostly for marketing!
- Strict benchmarking conditions
  - Fixed workload & faultload + Small set of metrics
- Workload & faultload:
  - May not be representative of the user scenario
- Metrics:
  - Fixed! May not satisfy the user needs
  - Decision based on several metrics is difficult!

What We Need...

- Benchmarking conditions adaptable to the user needs
- To be able to consider multiple properties
- Include multiple usage scenarios:
  - Metrics depend on the scenario
  - Adaptable workload, faultload, attackload…
- Use quality models instead of independent metrics
  - Quality models should also adapt to the scenario

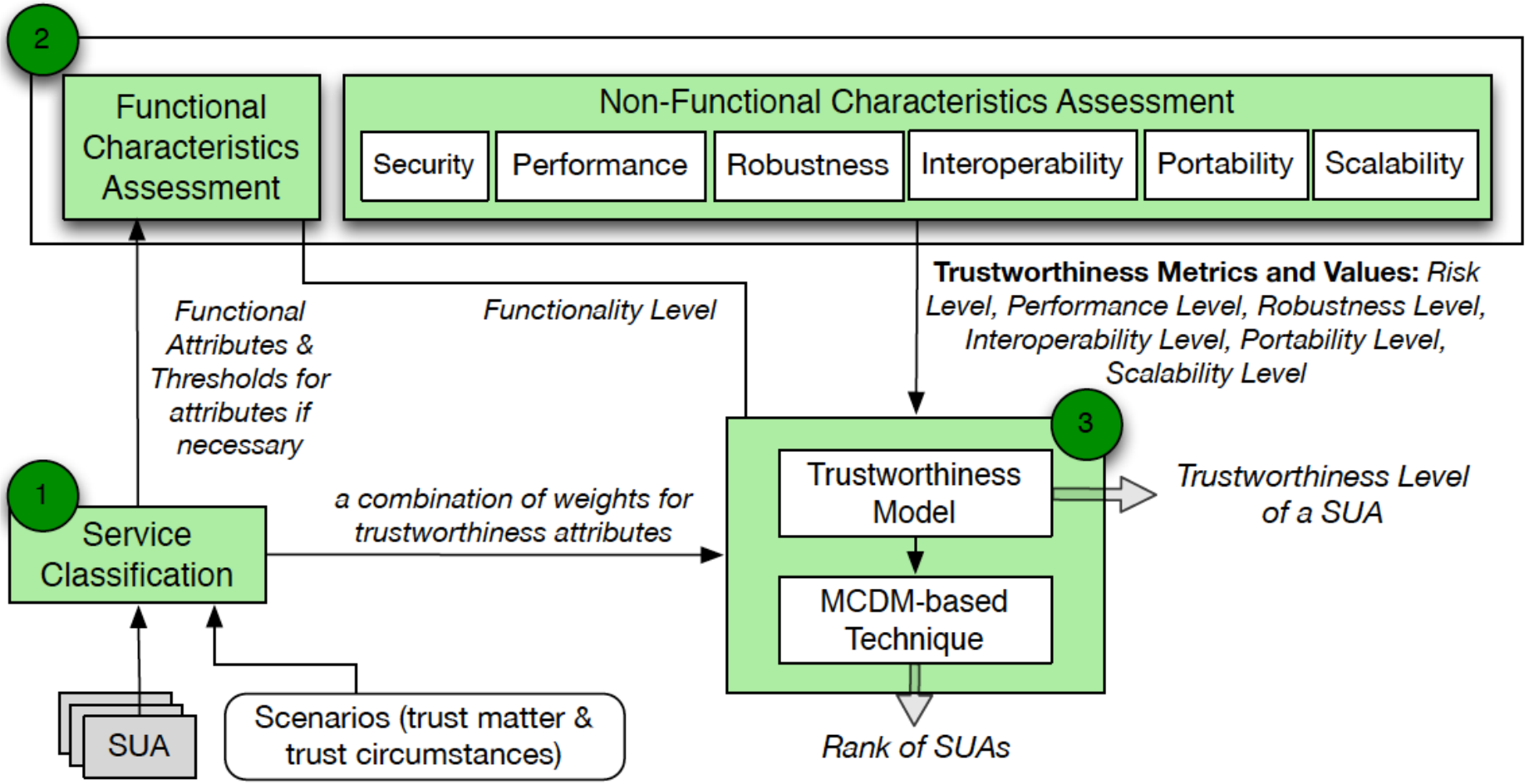
QUALITY MODELS

- A property may require several different scores
  - We need a model to map properties to scores

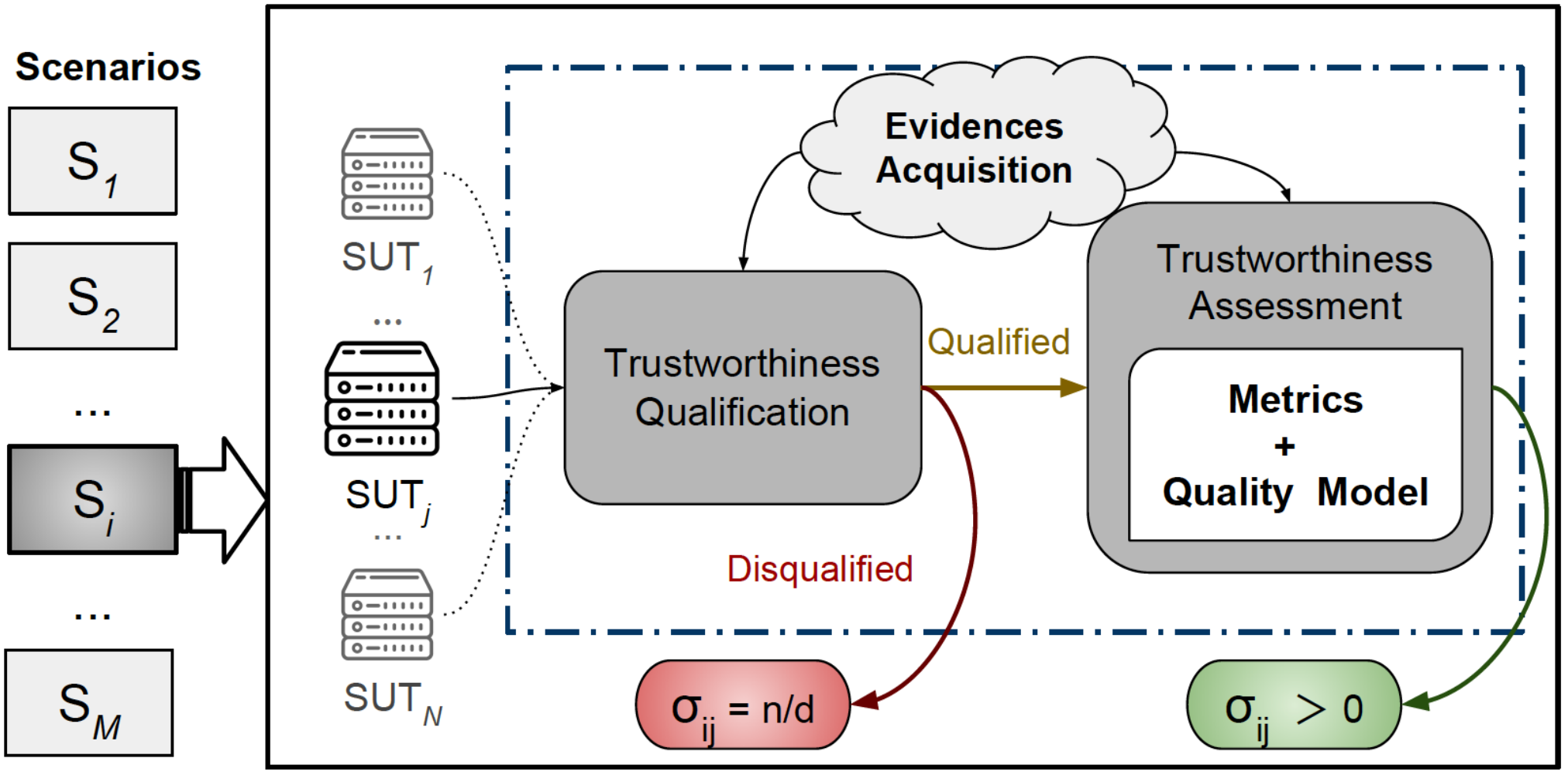


- Calculating a score requires collecting evidences
  - Trust evolves over time…
- Safety *vs* Availability *vs* Security *vs* Performance vs …

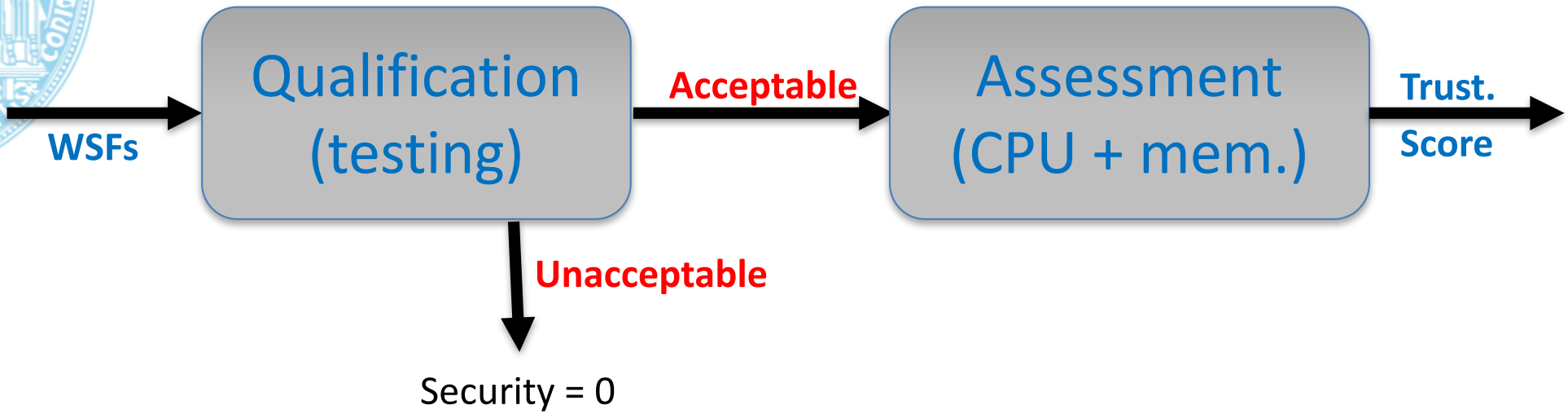
ASSESSMENT MODEL



BENCHMARKING PROCESS…

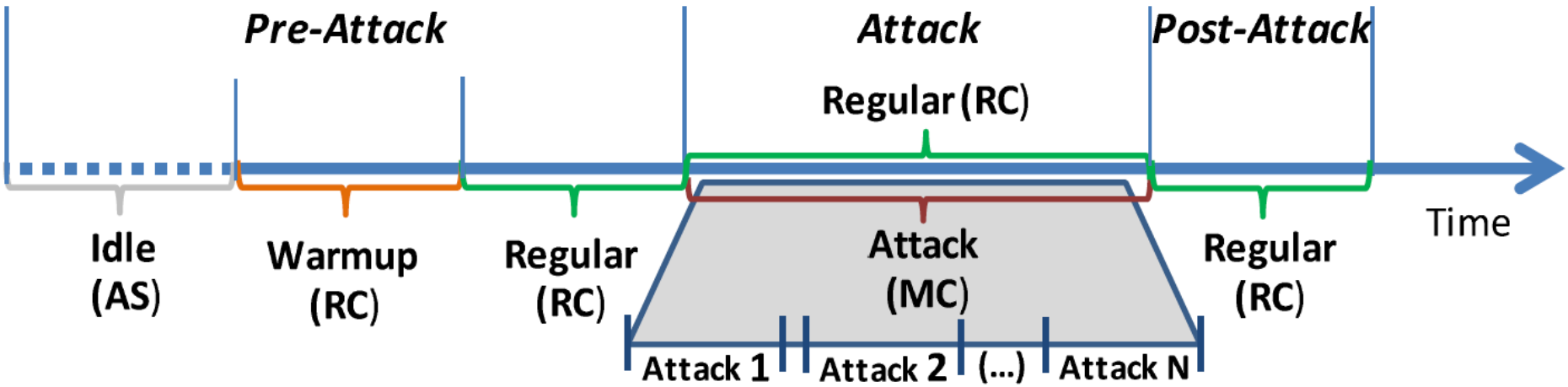


EXAMPLE: WEB SERVICE FRAMEWORKS

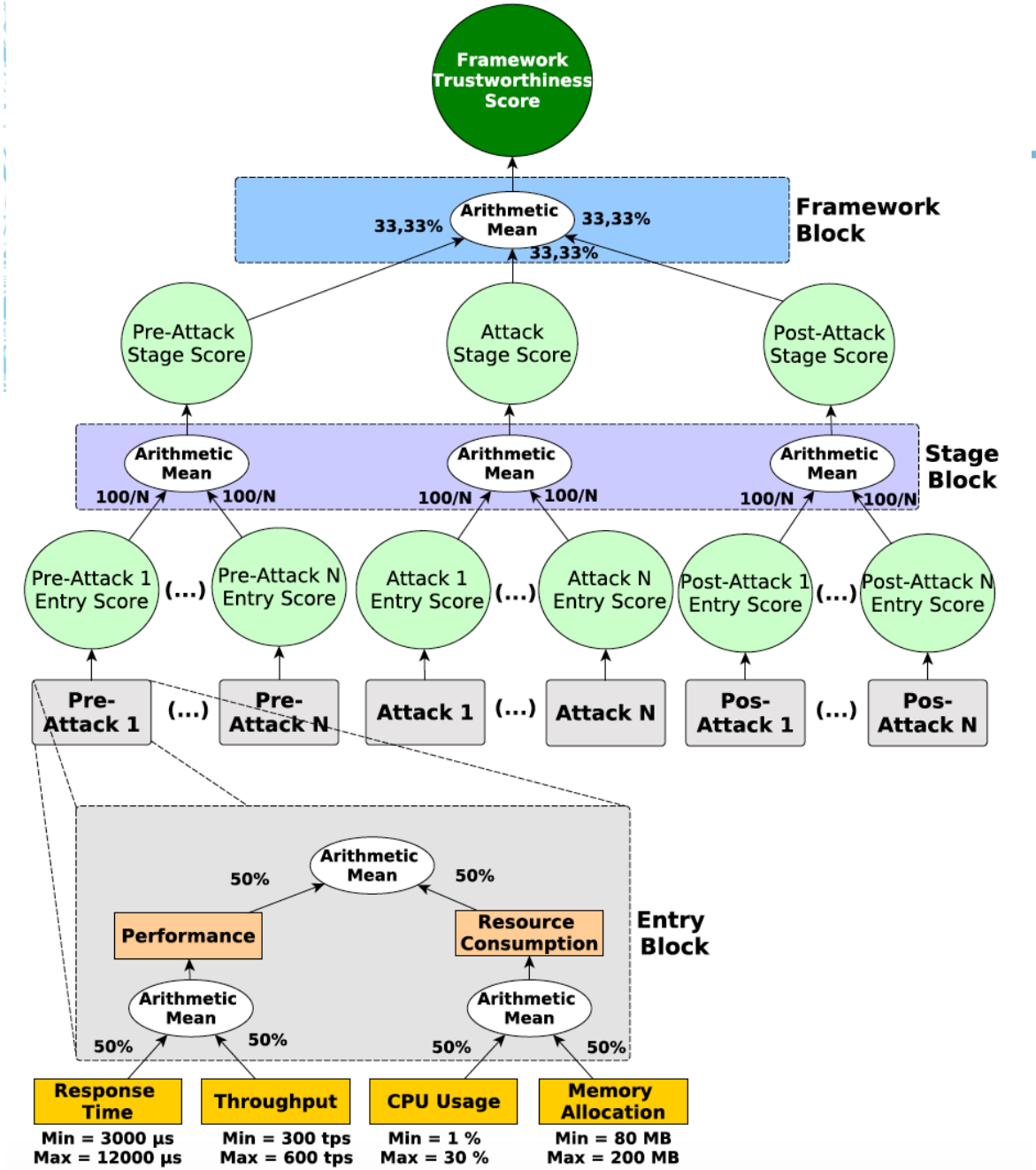


- Qualification
  - DoS Attacks
  - *Coercive Parsing, Malformed XML, Malicious Attachment, etc.*
- Trustworthiness Assessment:
  - Quality model to compute a score

EXPERIMENTS



QUALITY MODEL



SYSTEMS UNDER BENCHMARKING

Framework	Version	Security Qualification
Apache Axis 1	1.4.1	✗
Apache Axis 2	1.6.1	✓
	1.6.2	✗
Apache CXF	2.5.1	✓
	3.0.3	✓
Oracle Metro	2.1.1	✗
	2.3.1	✓
XINS	3.1	✗
Spring JAX-WS	1.9	✗
Spring WS	2.2.0	✗

TRUSTWORTHINESS RESULTS

Scenario	Axis 2	CXF v2	Metro	CXF v3
Neutral	72.3 (1)	70.7 (2)	58.1 (3)	57.9 (4)
Scenario1	73.4 (2)	77.1 (1)	66.5 (4)	70.0 (3)
Scenario2	67.4 (3)	73.1 (1)	66.6 (4)	68.7 (2)
Scenario3	61.8 (4)	70.3 (1)	63.6 (3)	67.0 (2)

Other Ongoing Works

- Trustworthiness Assessment of Source Code
  - Evidences: security coding practices
- Trustworthiness Benchmarking of Source Code
  - Evidences: software metrics
- Benchmarking Configurations of Mobile Devices
  - Evidences: security configurations
- Benchmarking Virtualization Infrastructures
  - Evidences: resource consumption and performance
- Failure Prediction for Trustworthiness Assessment
  - Evidences: predicted failures using Machine Learning

ATMOSPHERE

Adaptive, Trustworthy, Manageable, Orchestrated, Secure, Privacy-assuring, Hybrid Ecosystem for REsilient Cloud Computing

- Ongoing EU/Brazil H2020 project
- Provide a solution to enable the implementation of trustworthy and resilient cloud services
- Strong focus on trustworthiness assessment and benchmarking
  - Both at design- and run-time

CONSORTIUM

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Trust-IT Services Ltd
Communicating ICT to markets

Universidade Federal de Campina Grande

UNIVERSIDADE DE COIMBRA

POLITECNICO DI MILANO

UNIVERSIDADE FEDERAL DE MINAS GERAIS

Universidade de Brasília
Departamento de Ciência da Computação

TECHNISCHE UNIVERSITÄT DRESDEN

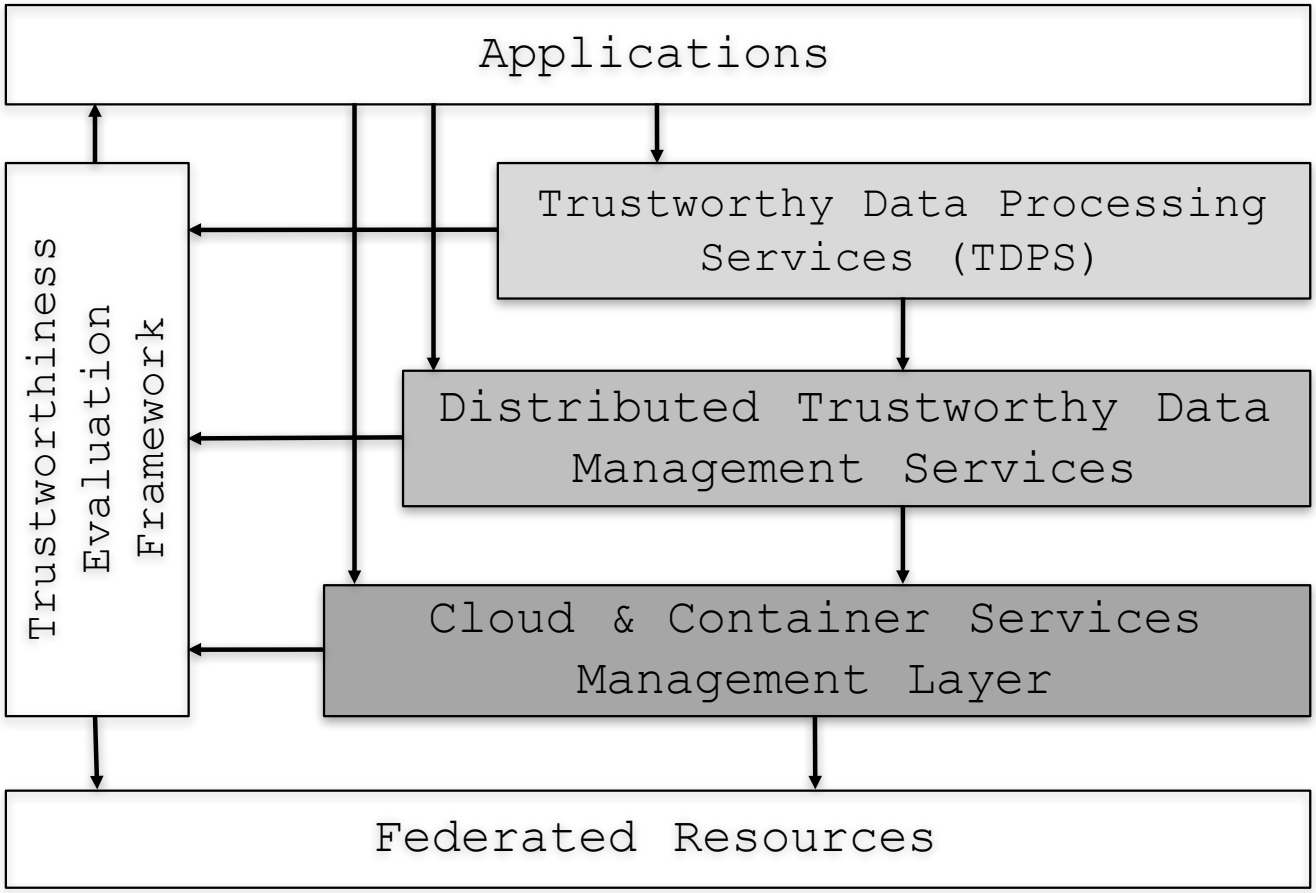
UNIVERSITY OF PIRAEUS RESEARCH CENTER

UNICAMP
UNIVERSIDADE ESTADUAL DE CAMPINAS

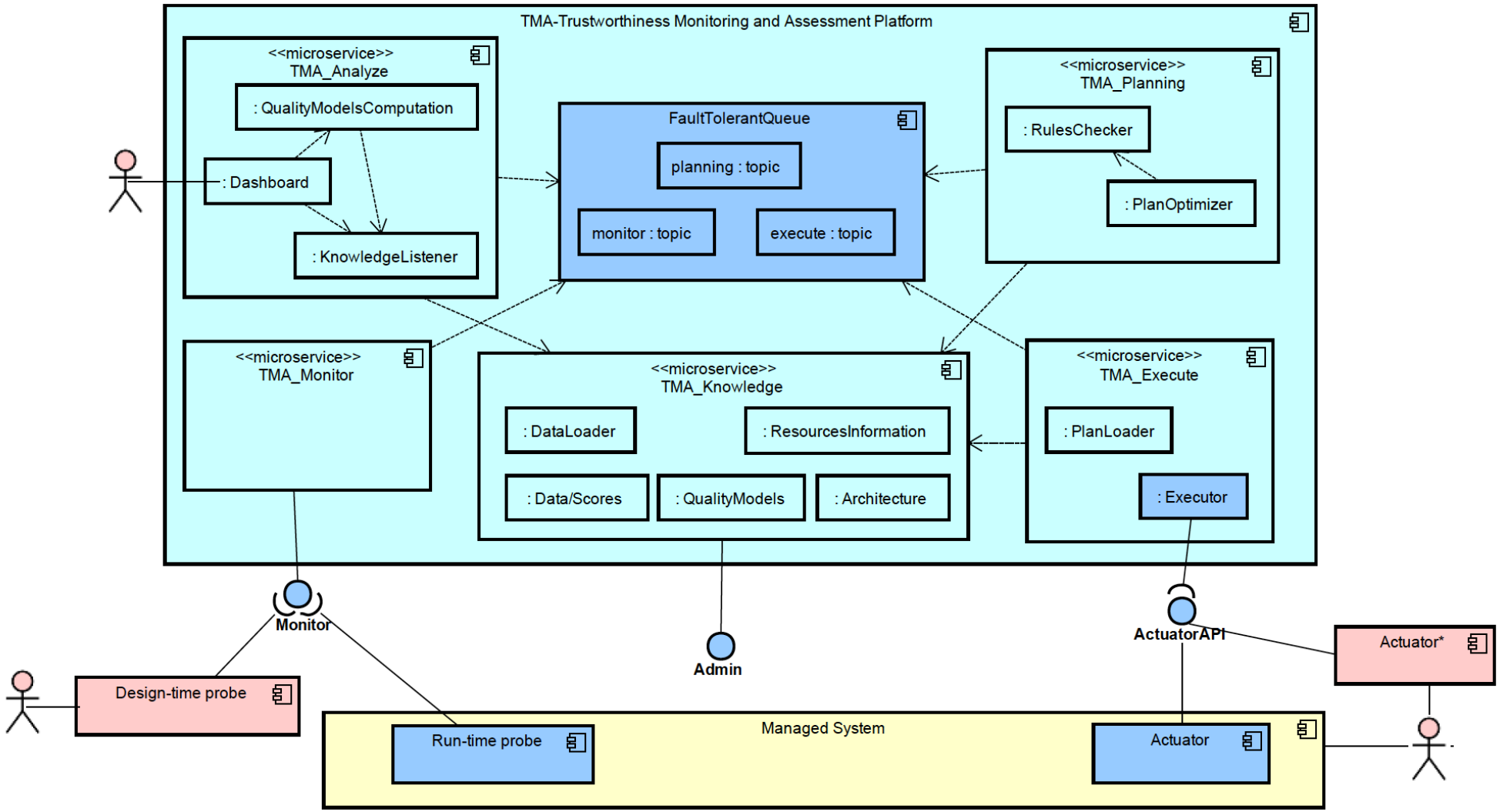
DELL EMC

quibim

TRUSTWORTHINESS FRAMEWORK…



TMA PLATFORM



QMs

- Cloud Federated Resources
  - Characterize trustworthiness of federated resources, namely services, resources and network links
  - Focusing on availability, isolation, performance and latency
- Cloud & Container Services
  - Virtual services and virtual resources
  - Availability, performance, scalability and isolation
- Data Management
  - Performance, security, privacy and fault tolerance
- Data Processing
  - Fairness, stability, transparency and privacy of ML models

CHALLENGES

- How to define attributes and sub-attributes for each trustworthiness property?
  - As well as the appropriate scores to characterize them
- How to define a set of scenarios that suggest a set of different weights for the different properties?
- How to define a measurement mechanism for each trustworthiness property?
  - Design-time and run-time
- How to build trustworthiness models based on the relevant attributes?

OUTLINE

- Trustworthiness: An Integrative Concept
- Benchmarking: Past and Present
- From Security to Trustworthiness Benchmarking
- Trustworthiness Benchmarking Framework
- Challenge: Trustworthiness Benchmarking in Safety Critical Systems
- Conclusions

TRUSTWORTHINESS IN CRITICAL SYSTEMS

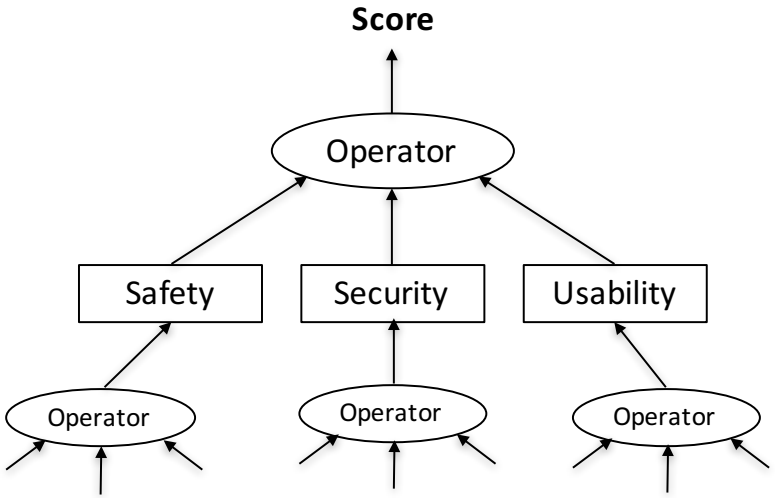
- Safety is a trustworthiness property (safety risk)
- Safety cases supported by evidence
  - Failure modes and effects analysis (FMEA)
  - Failure mode, effects, and criticality analysis (FMECA)
  - …
- Can these evidences be used to benchmark/compare?
- Is benchmarking useful in this context?
  - Assess different alternatives for a given component…
  - Compare different components
  - Identify weaknesses in the system
  - …

SAFETY *VS* SECURITY

- Safety implications on security?
- Security implications on safety?
- How to benchmark?
  - Safety evidences
  - Security evidences
  - Safety vs Security
  - Quality Models
  - Procedure
  - …

SAFETY *VS* ...

- Safety *vs* Performance
- Safety *vs* Availability
- Safety *vs* Usability
- Safety *vs* Performance *vs* Security
- ...



- How to benchmark, in general?

SAFETY IN AI

- Machine Learning is/will be widely used in Safety Critical systems
- How to assess safety in those cases?
  - Unpredictability
  - Evolution
  - Properties like fairness and transparency
  - Safety cases may be imitated…
- Design- and run-time assessment
- Collecting trustworthiness evidences is probably the best we can do: no absolute measure

EXAMPLE…

Select a component for a Safety Critical system

1. Identify trustworthiness properties of interest
2. Define trustworthiness scores for each property
3. Define/adapt/reuse the QMs to compute the scores
4. Identify the data sources for feeding the QMs
5. Implement probes to collect data
6. Deploy probes and QMs
7. Stimulate the SUB (run experiments)
8. Compute the scores and compare

CONCLUSIONS

- The benchmarking concept is well established!
  - Performance and dependability benchmarks are well known
  - Security benchmarking approaches are weak
- Trustworthiness brings a different perspective
  - Multiple properties, metrics, and scenarios
  - Qualification and assessment based on (potentially subjective) evidences
- Trustworthiness benchmarking in safety critical systems is an open topic
  - Particularly interesting for addressing the relation and impact among multiple properties!

More Questions than Answers

QUESTIONS?

Marco Vieira
Department of Informatics Engineering
University of Coimbra
mvieira@dei.uc.pt
http://eden.dei.uc.pt/~mvieira