A Survey on Simulation-Based Fault Injection Tools for Complex Systems

DTIS'14

Maha Kooli, Giorgio Di Natale

Cross-Layer Early Reliability Evaluation for the Computing cOntinuum

Outline

1. Research Project
2. Dependability
3. Fault Tolerance
4. Fault Injection
5. Conclusion and Perspective

Outline

1. **Research Project**
2. Dependability
3. Fault Tolerance
4. Fault Injection
5. Conclusion and Perspective

Motivation

Availability

Security

Reliability

Fault Tolerance

Space Shuttle Columbia, February 1, 2003

CLERECO

Clereco

Cross-Layer Early Reliability Evaluation for the Computing cOntinuum

OUTLINE

1. Research Project
2. **Dependability**
3. Fault Tolerance
4. Fault Injection
5. Conclusion and Perspective

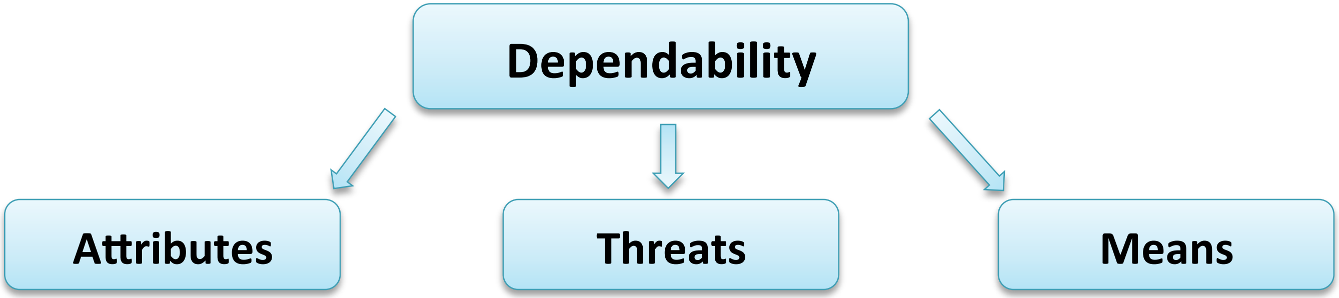
DEPENDABILITY

Dependability

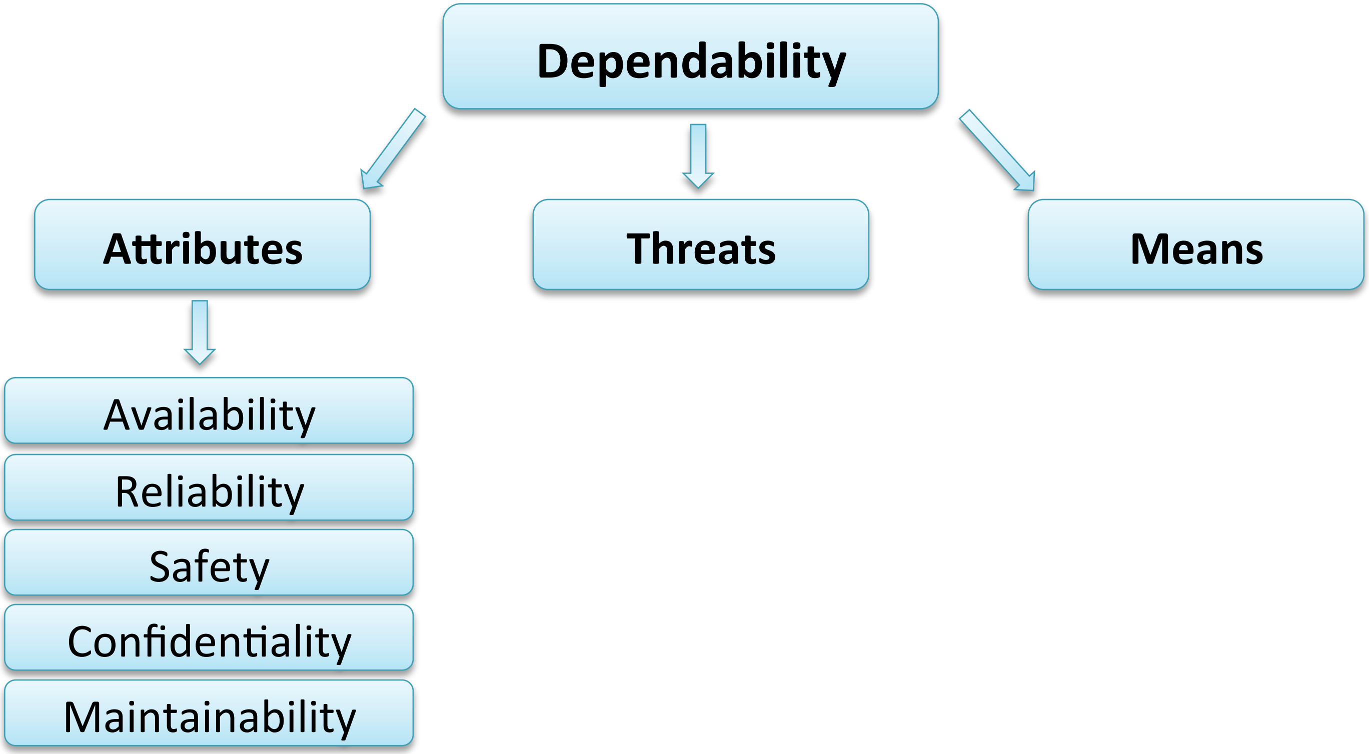
- A global concept that subsumes the usual attributes of reliability, availability, safety, integrity, and maintainability.

- The ability to avoid service failures that can happen to a system frequently and severely than acceptable.

Dependability



DEPENDABILITY



Dependability

Attributes

Threats

Means

Availability

Reliability

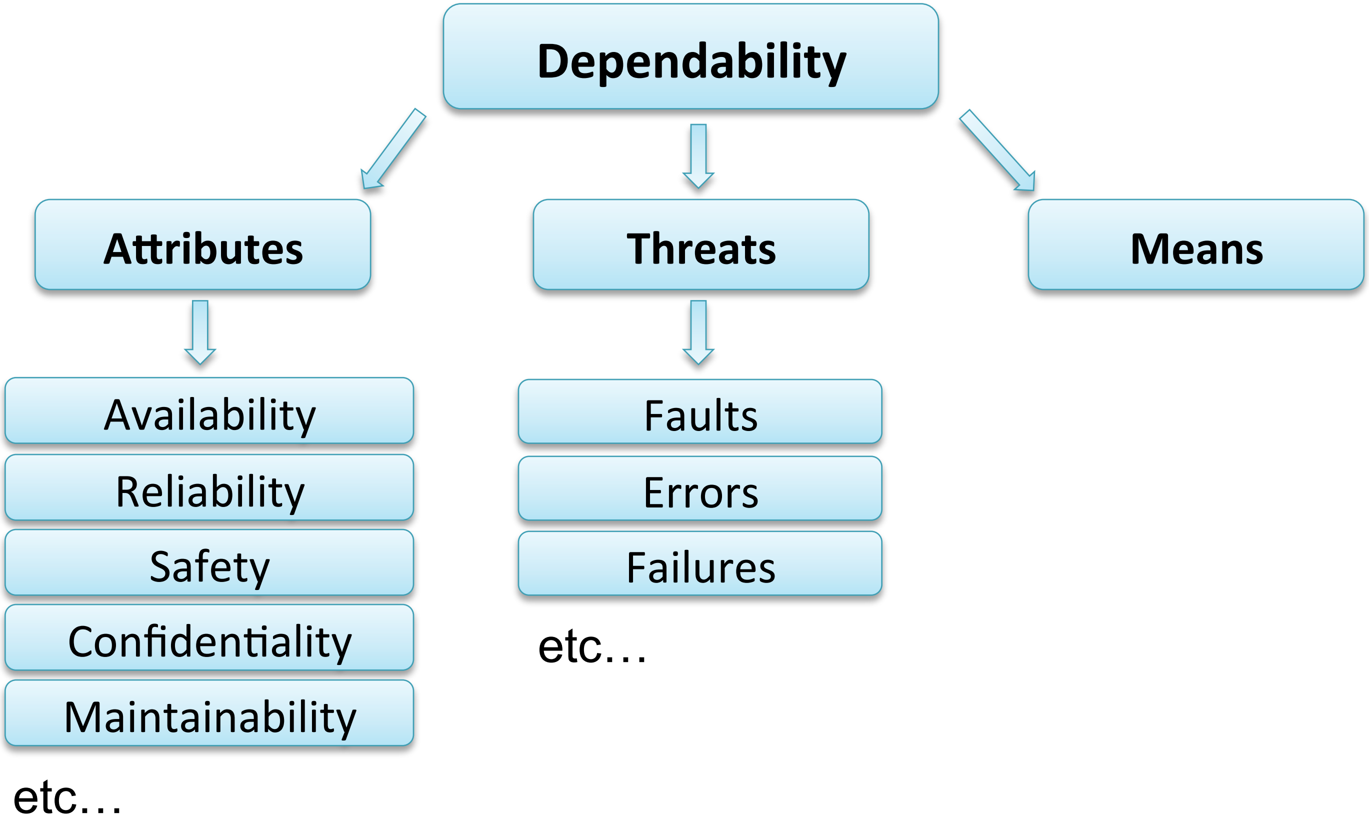
Safety

Confidentiality

Maintainability

etc…

Dependability



Dependability

Attributes

Threats

Means

Availability

Reliability

Safety

Confidentiality

Maintainability

etc…

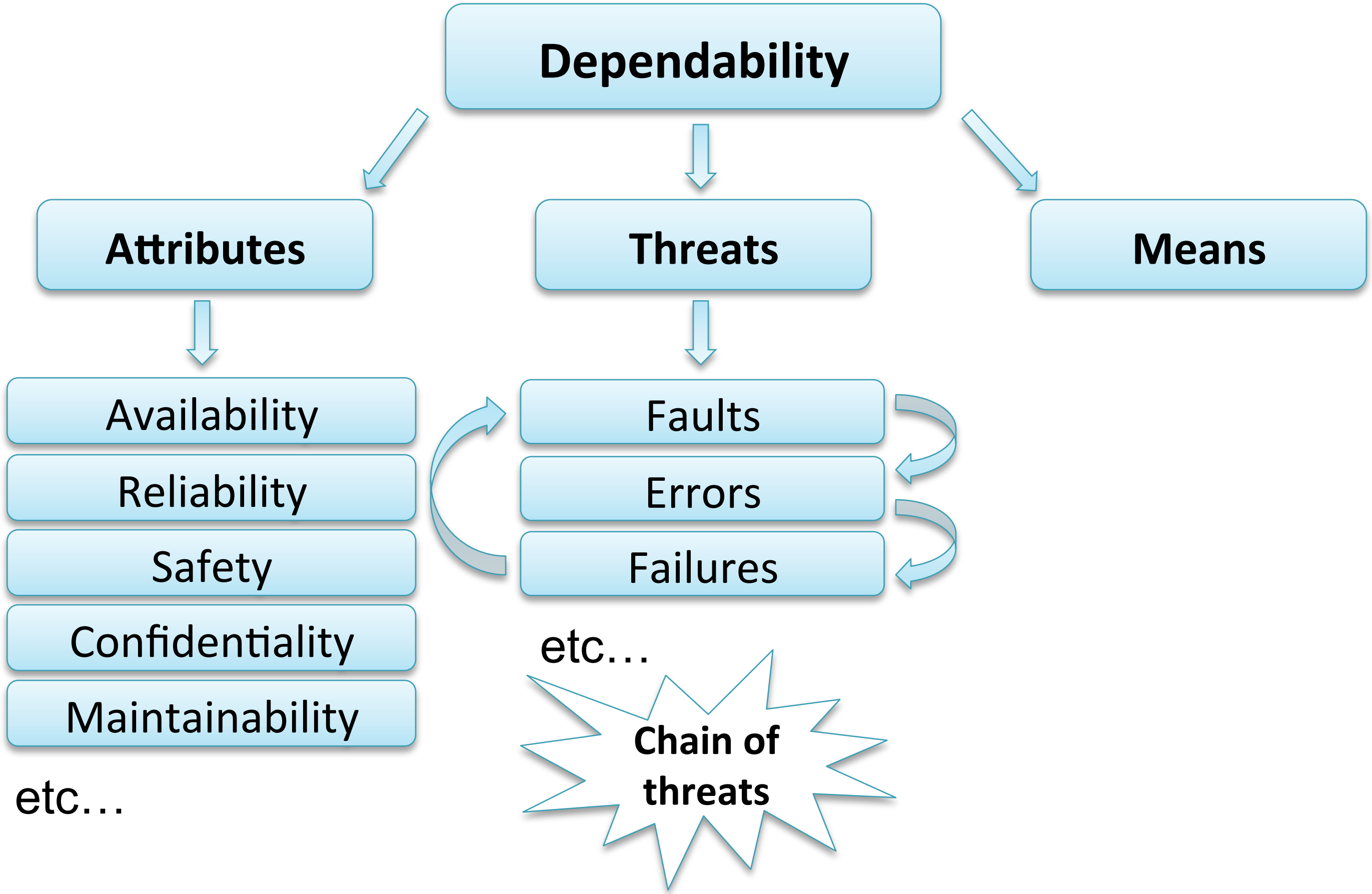
Faults

Errors

Failures

etc…

DEPENDABILITY



Dependability

Attributes

Threats

Means

Availability

Reliability

Safety

Confidentiality

Maintainability

etc…

Faults

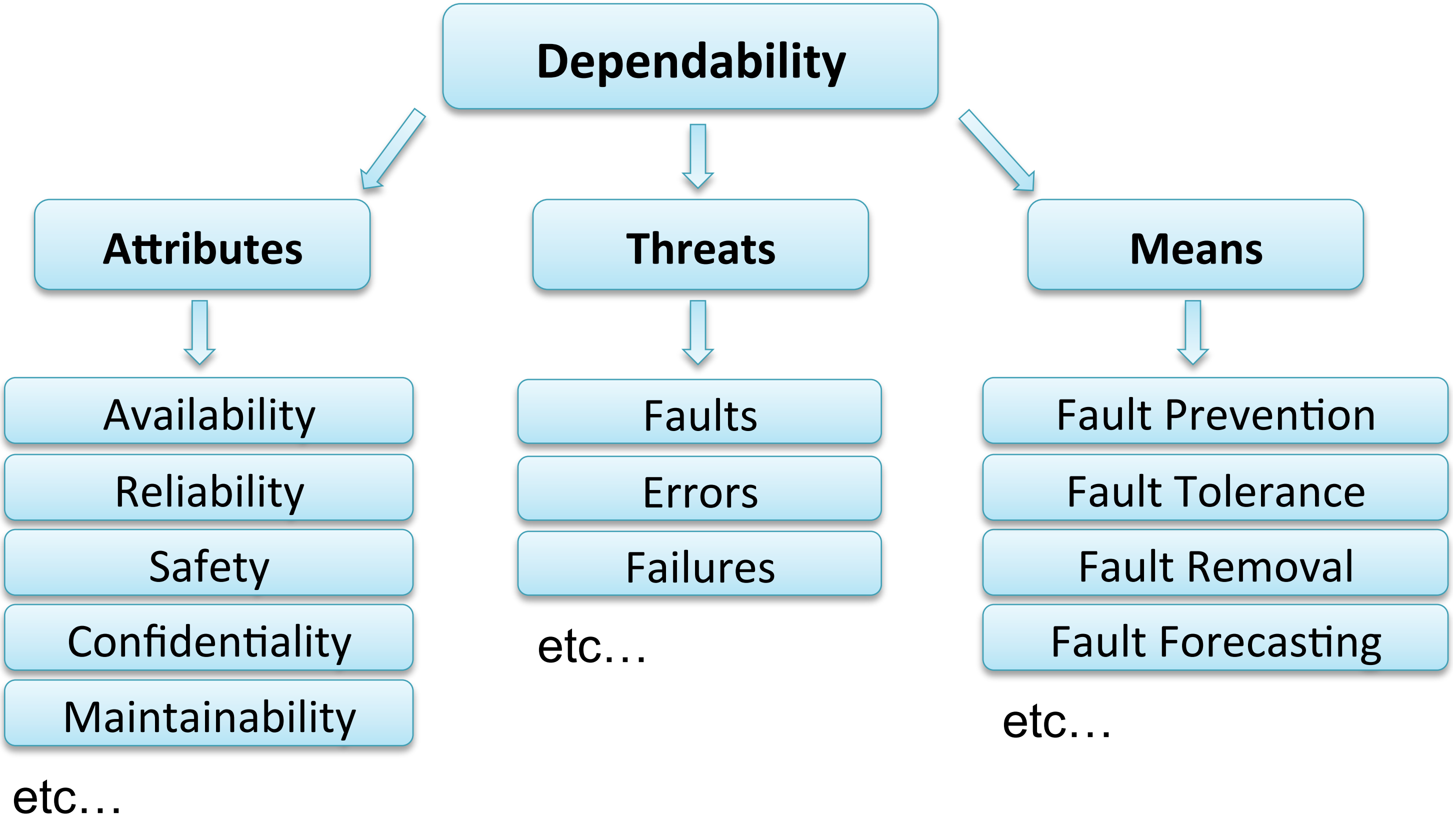
Errors

Failures

etc…

Chain of threats

Dependability



Dependability

Attributes

Availability

Reliability

Safety

Confidentiality

Maintainability

etc…

Threats

Faults

Errors

Failures

etc…

Means

Fault Prevention

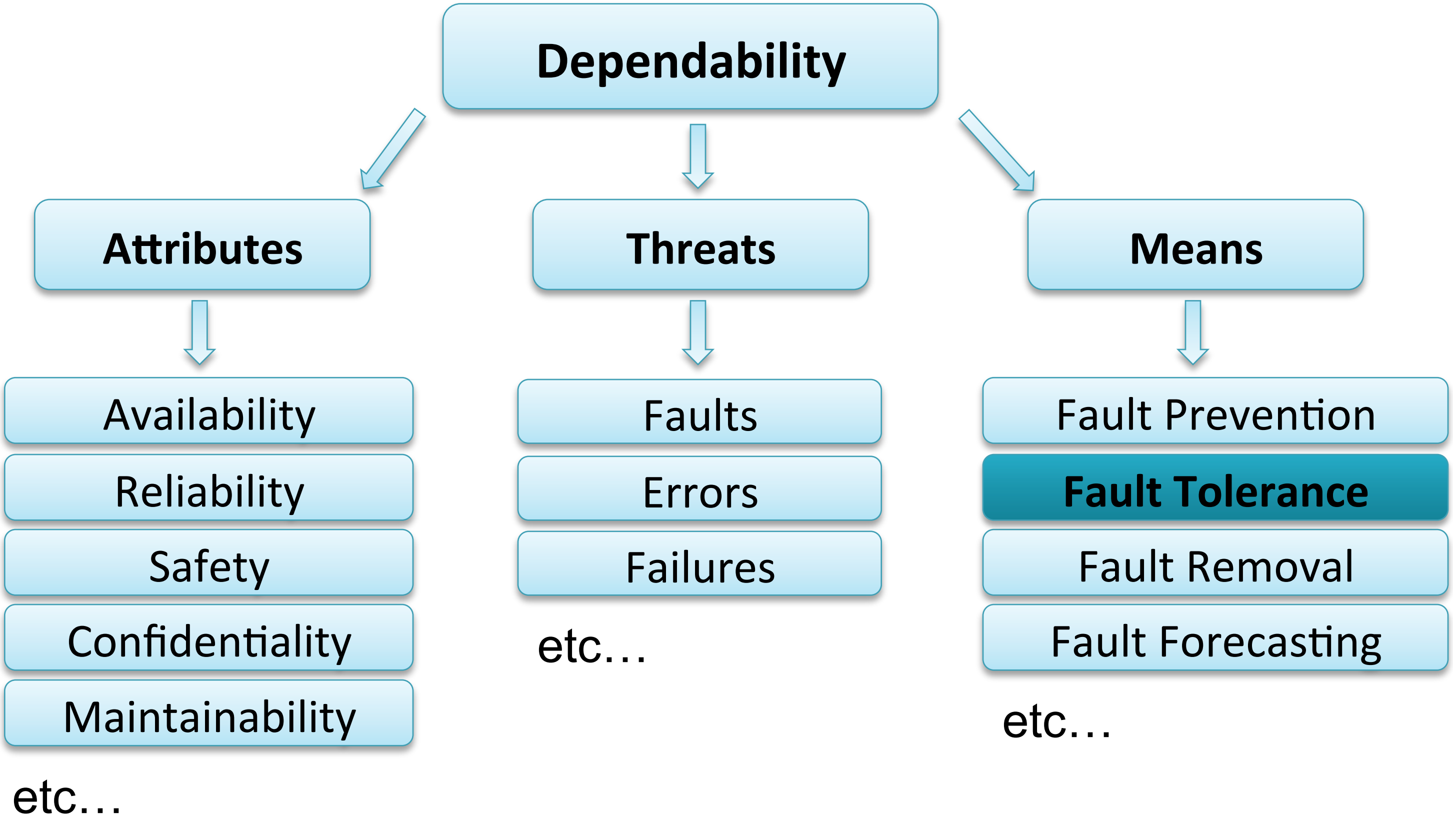
Fault Tolerance

Fault Removal

Fault Forecasting

etc…

Dependability



Dependability

Attributes

- Availability
- Reliability
- Safety
- Confidentiality
- Maintainability

etc…

Threats

- Faults
- Errors
- Failures

etc…

Means

- Fault Prevention
- **Fault Tolerance**
- Fault Removal
- Fault Forecasting

etc…

Outline

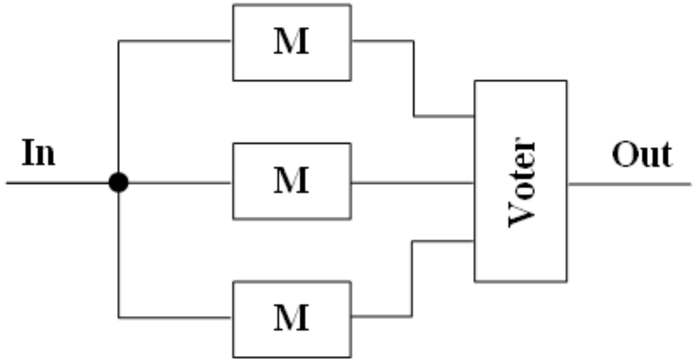
1. Research Project
2. Dependability
3. **Fault Tolerance**
4. Fault Injection
5. Conclusion and Perspective

Fault Tolerance

- A method permitting to:
  - Increase the dependability
  - Avoid services failure in the presence of faults

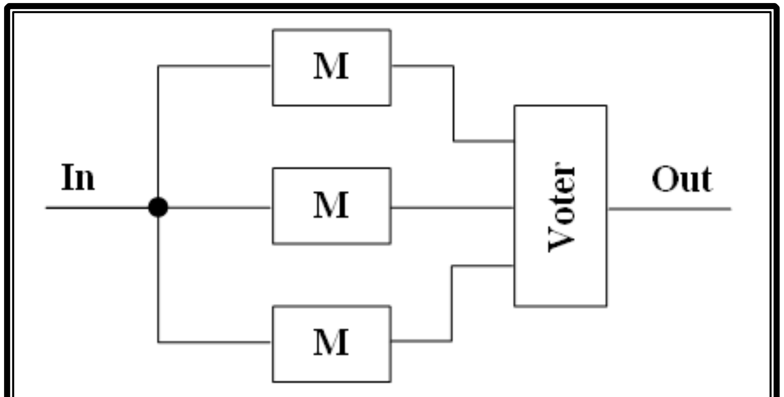
- Several strategies are used to tolerate faults: **Redundancy**

Redundancy

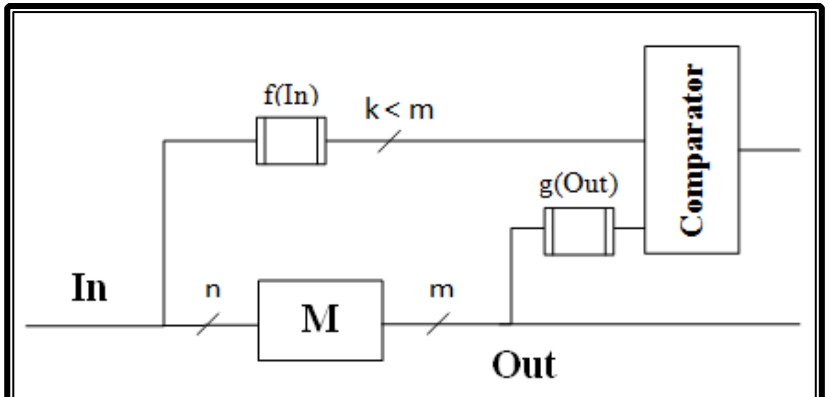


Space Redundancy

Redundancy

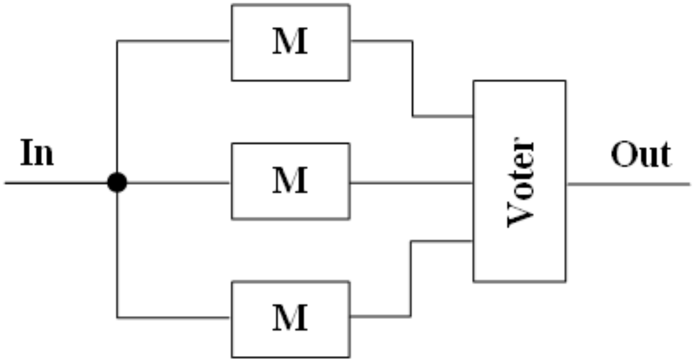


Space Redundancy

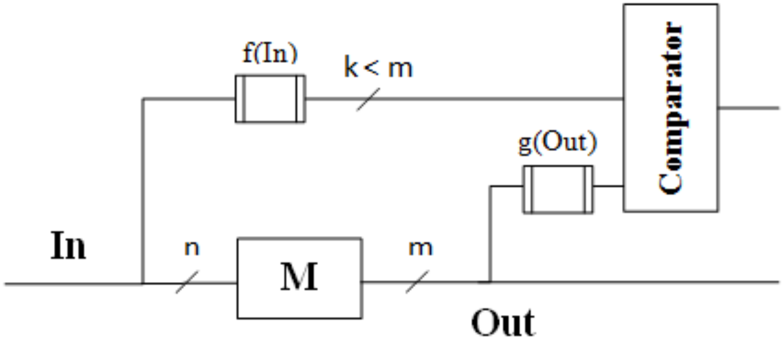


Information Redundancy

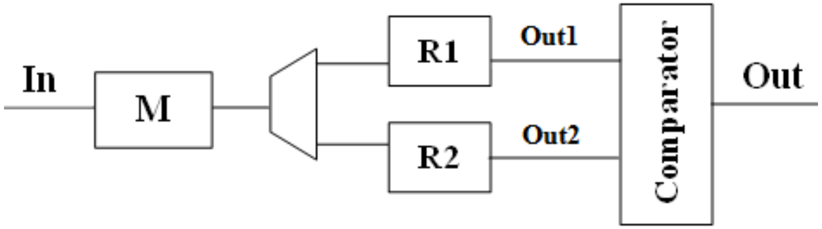
Redundancy



Space Redundancy



Information Redundancy



Time Redundancy

Fault Tolerance Techniques

Techniques	Targeted Layer	+/-
Hardware Fault Tolerance	Hardware Layer	- Effective - Costly in term of equipment
Software Implemented Hardware Fault Tolerance (SIHFT)	Operating System Layer	- Requires high skills to modify the OS
	Middleware Layer	- Efficient when the application is not modifiable, and the source code is not available
	Application Software Layer	- Cheap - Better solution when the source code is available

Outline

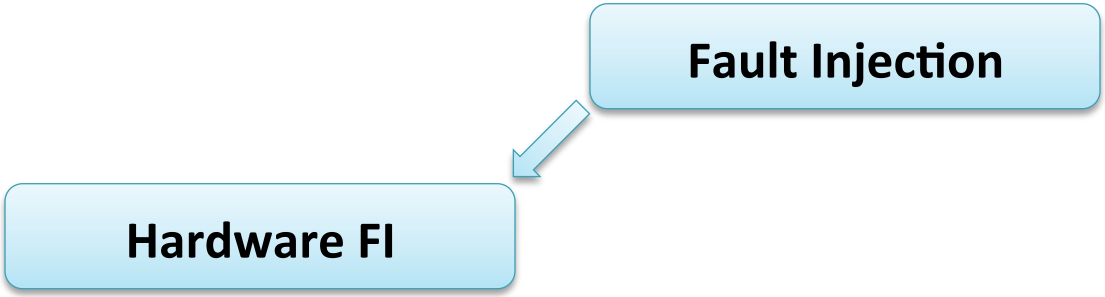
1. Research Project
2. Dependability
3. Fault Tolerance
4. **Fault Injection**
5. Conclusion and Perspective

Fault Injection

Fault Injection

- A validation technique of the dependability for fault tolerance systems
- Evaluate the behavior of the system in the presence of faults

Fault Injection

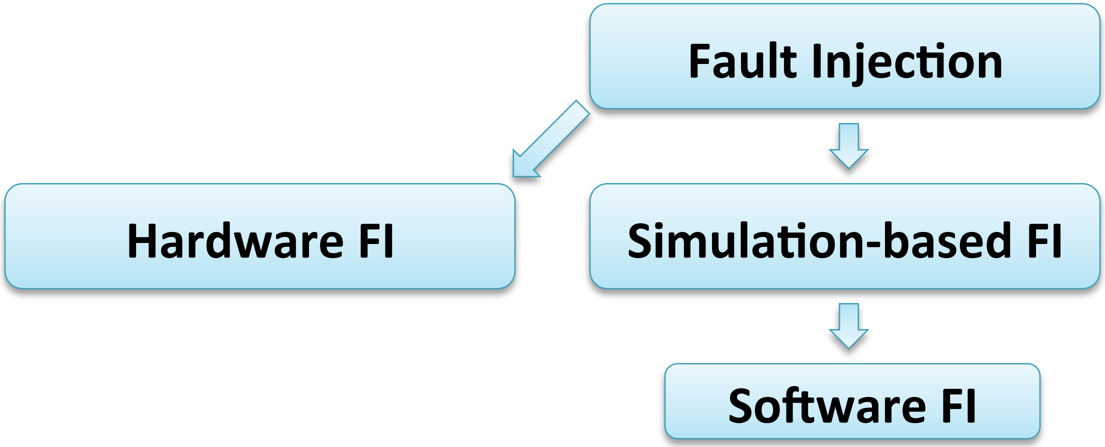


+ High time-resolution.

− Expensive in term of equipment.

− Risk to damage the system.

Fault Injection


Fault Injection

Hardware FI

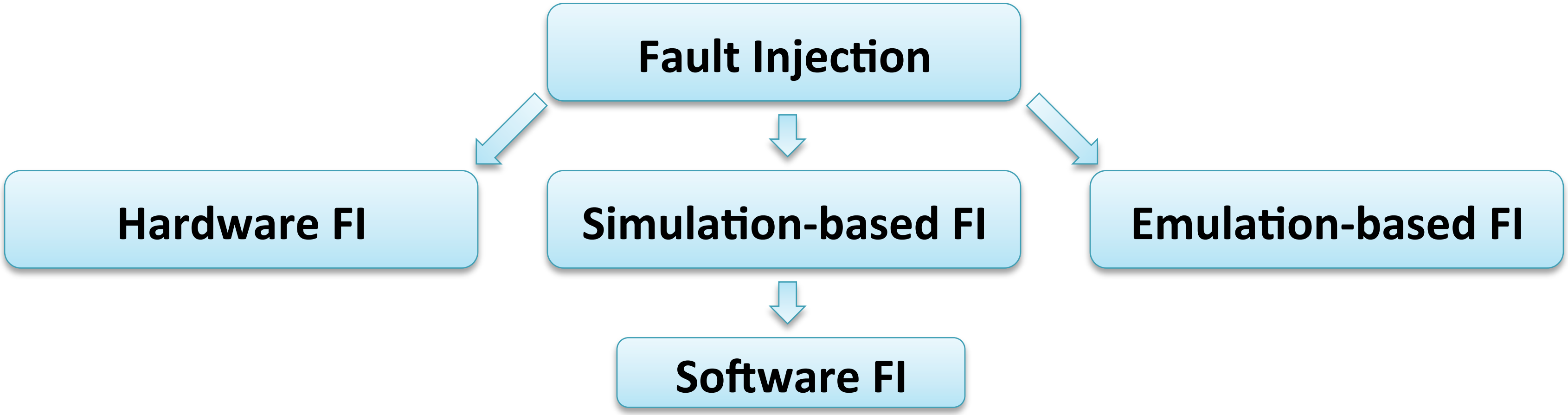
Simulation-based FI

Software FI

- + High time-resolution.
- − Expensive in term of equipment.
- − Risk to damage the system.

- + No risk to damage the system
- + Low-cost
- + Simple to set-up
- − Accuracy of fault model and system model

Fault Injection



- + High time-resolution.
- − Expensive in term of equipment.
- − Risk to damage the system.

- + No risk to damage the system
- + Low-cost
- + Simple to set-up
- − Accuracy of fault model and system model

- + It reduces the execution time compared to the simulation-based FI
- − Costly and not flexible

Fault Injection Environments

Tool	Developer	Category	Description
Xception	University of Coimbra, Portugal	Software FI	• Fault are injected in software • It uses advanced debugging features to observe the behavior of the system in detail in presence of faults.

Fault Injection Environments

Tool	Developer	Category	Description
Xception	University of Coimbra, Portugal	Software FI	• Fault are injected in software • It uses advanced debugging features to observe the behavior of the system in detail in presence of faults.
Jaca	University of Campinas, Brazil	Software FI	• Faults are injected in object-oriented systems. • Adapted to any Java application.

Fault Injection Environments

Tool	Developer	Category	Description
Xception	University of Coimbra, Portugal	Software FI	• Fault are injected in software • It uses advanced debugging features to observe the behavior of the system in detail in presence of faults.
Jaca	University of Campinas, Brazil	Software FI	• Faults are injected in object-oriented systems. • Adapted to any Java application.
RIFLE	University of Coimbra, Portugal	Hardware FI	• Faults are injected in pin-level of the modules. • It performs analysis to observe the impact of faults on the processor

Fault Injection Environments

Tool	Developer	Category	Description
LLVM-based	Many	Software FI	•LLVM is a compiler infrastructure that supports several programming languages and instruction set •LLVM-based FI inject faults in the intermediate level of the application source code

Fault Injection Environments

Tool	Developer	Category	Description
LLVM-based	Many	Software FI	•LLVM is a compiler infrastructure that supports several programming languages and instruction set •LLVM-based FI inject faults in the intermediate level of the application source code
LIFTING	LIRMM Montpellier , France	Simulation-based FI	•A simulator able to perform both logic and fault simulation for stuck-at faults and single event-upset on digital circuits •It provides many features to analyze the fault simulation results

Fault Injection Environments

Tool	Developer	Category	Description
LLVM-based	Many	Software FI	•LLVM is a compiler infrastructure that supports several programming languages and instruction set •LLVM-based FI inject faults in the intermediate level of the application source code
LIFTING	LIRMM Montpellier , France	Simulation-based FI	•A simulator able to perform both logic and fault simulation for stuck-at faults and single event-upset on digital circuits •It provides many features to analyze the fault simulation results
FAUmachine	University of Erlangen-Nuremberg in Germany	Simulation-based FI	•A virtual machine that permits to inject faults and observe the whole operating system or application software •It injects faults in different component of the system such as memory, disk, and network

Fault Injection Environments

Tool	Developer	Category	Description
LLVM-based	Many	Software FI	•LLVM is a compiler infrastructure that supports several programming languages and instruction set •LLVM-based FI inject faults in the intermediate level of the application source code
LIFTING	LIRMM Montpellier , France	Simulation-based FI	•A simulator able to perform both logic and fault simulation for stuck-at faults and single event-upset on digital circuits •It provides many features to analyze the fault simulation results
FAUmachine	University of Erlangen-Nuremberg in Germany	Simulation-based FI	•A virtual machine that permits to inject faults and observe the whole operating system or application software •It injects faults in different component of the system such as memory, disk, and network

LLVM-based Fault Injector

- LLVM is a compiler infrastructure, defines an abstraction layer to make the information at software level and the information at hardware level, compatible and easily exchangeable

KULFI & **LLFI**

- Transient and permanent faults injected in the LLVM Intermediate Representation code level

⇨ Target the software layer independently of the architecture of the hardware system

⇨ Observe the effect of hardware faults in the behavior of the application

FAUmachine

- Virtual machine similar to QEMU or Virtual Box, supports Just-In-Time (JIT) compiling
- It supports Linux as OS, and i386 and x86_64 as hardware

- **Memory:** Transient bit flips and Permanent stuck-at and coupling faults
- **Disk/CDROM:** Transient and permanent block and whole disk faults
- **Network:** Transient, Intermittent and Permanent send and receive faults

⇨ Thanks to virtualization, we can observe the fault impacts in the behavior of the whole operating system, without having a real hardware system.

OUTLINE

1. Research Project
2. Dependability
3. Fault Tolerance
4. Fault Injection
5. **Conclusion and Perspective**

Conclusion and Perspective

- Define the software fault model:
  - The misbehaviors of the software due to the propagation of a hardware error
  - It should be able to simulate in a software environment that allows injecting faults, without knowing the hardware structure.

- Simulate the fault models at a high level of the system

- Compare the obtained results with the result of the simulation of hardware fault in a real hardware system

Thanks for your attention

Questions?