Dependability

Faults are unexpected events

A system can, and usually does, fail. Is it however still dependable?

Dependability definition: "that property of a computer system such that reliance can justifiably be placed on the service it delivers"

Alternative definition of dependability that provides a criterion for deciding whether or not, in spite of service failures a system, is still to be regarded as dependable:

Dependability definition:

"The ability of a system to avoid service failures that are more frequent or more severe than is acceptable "

The occurrence of faults in the system cannot be avoided, systems can never be proved totally free of faults

Dependability attribute must be considered in a probabilistic sense

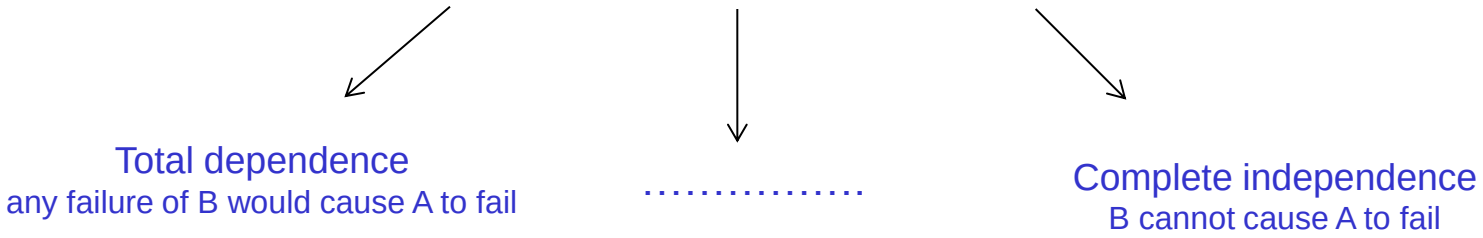
A. Avizienis, J.C. Laprie, B. Randell, C. Landwehr
Basic Concepts and Taxonomy of Dependable and Secure Computing
IEEE Transactions on Dependable and Secure Computing, Vol. 1, N. 1, 2004

Dependability

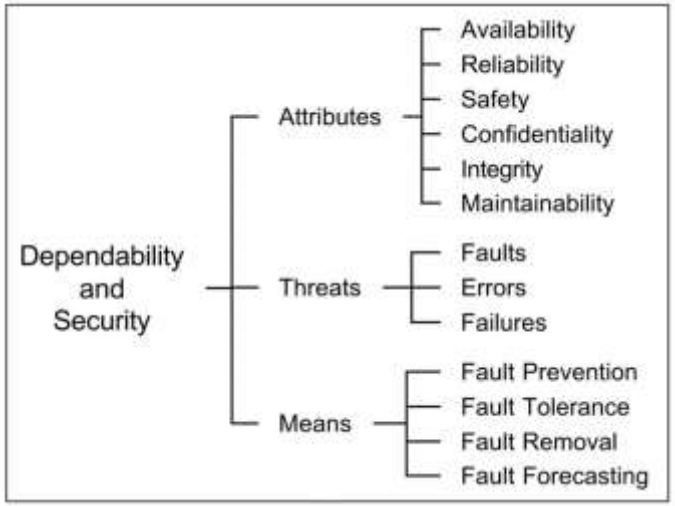
A system is the composition of interacting sub-systems.

Concept of dependence of system A on system B.

(the extent to which the dependability of system A is affected by the dependability of system B)



- if B dependability is insufficient for A dependability, B dependability must be improved, or A dependence on B must be reduced or additional means to tolerate faults must be added
- Concept of accepted dependence:
  the level of dependence of A on B that is acceptable (contract between A and B)
- A may fail to provide means of tolerating B failures



Dependability tree

Means for Dependability

Fault prevention techniques

Fault prevention techniques

- Fault prevention techniques are intended to keep faults out of the system

- Related to general system engineering techniques

  Prevention of development faults is the aim for development methodolgies both for software (formal specification, modularization, …) and hardware (design rules, …)

  Improvement of development processes in order to reduce the number of faults introduced based on recording of faults in the products and the elimination of causes od faults, via process modification.

Fault tolerance techniques

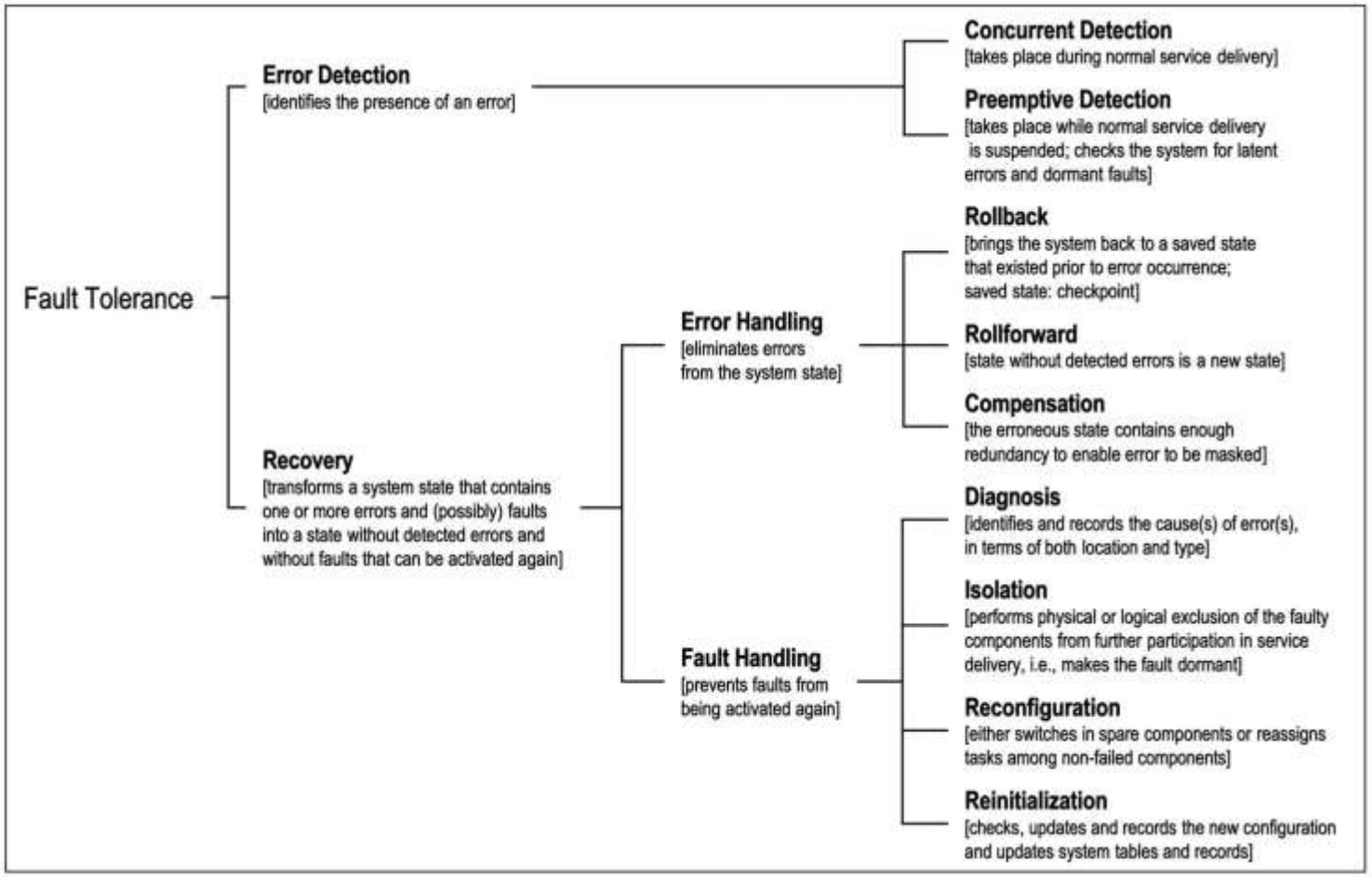
Fault tolerance: ability of the system to deliver a correct service after the occurrence of faults

- Why fault tolerance techniques?
  even with the most careful fault prevention, faults will eventually occur and result in a system failure

- Fault tolerance techniques:
  carried out via error detection and system recovery**,** *redundancy* to counteract the effects of faults

Protective redundancy:
additional components or processes that mask or correct errors or faults inside a system so they do not become observable failures in its service

Organisation of fault tolerance



A. Avizienis, J.C. Laprie, B. Randell, C. Landwehr. Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Transactions on Dependable and Secure Computing, Vol. 1, N. 1, 2004

Basic strategy for fault tolerance

Error detection and system recovery

where system recovery = error handling and fault handling
and fault handling is followed by corrective maintenance aimed at removing faults that were isolated

Error Handling

- Rollback/rollforward (on demand after error detection)

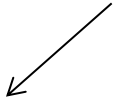
- Compensation (on demand / systematically)

⇨ **fault masking**

Disadvantage: loss of protective redundancy

Practical implementations of compensation: *masking and recovery*
(includes error detection and fault handling)

A general method to achieve fault tolerance is to perform multiple computations through multiple channels, either sequencially or concurrently



Tolerance of physical faults
channels may be of identical design
(we have the assumption that **hardware components fail independently**)

Tolerance of software faults
channels must implement the same function via separate designs and implementations
(**design diversity**)

Fault handling may directly follow error detection (without error handling)

Preemptive error detection and handling
is commonly performed at system power up and in some forms during operation for example memory scrubbing, audit programs ,…

Self-checking component (hardware or software):
functional capability together with concurrent error detection mechanism
Advantage: clear definition of error confinement areas

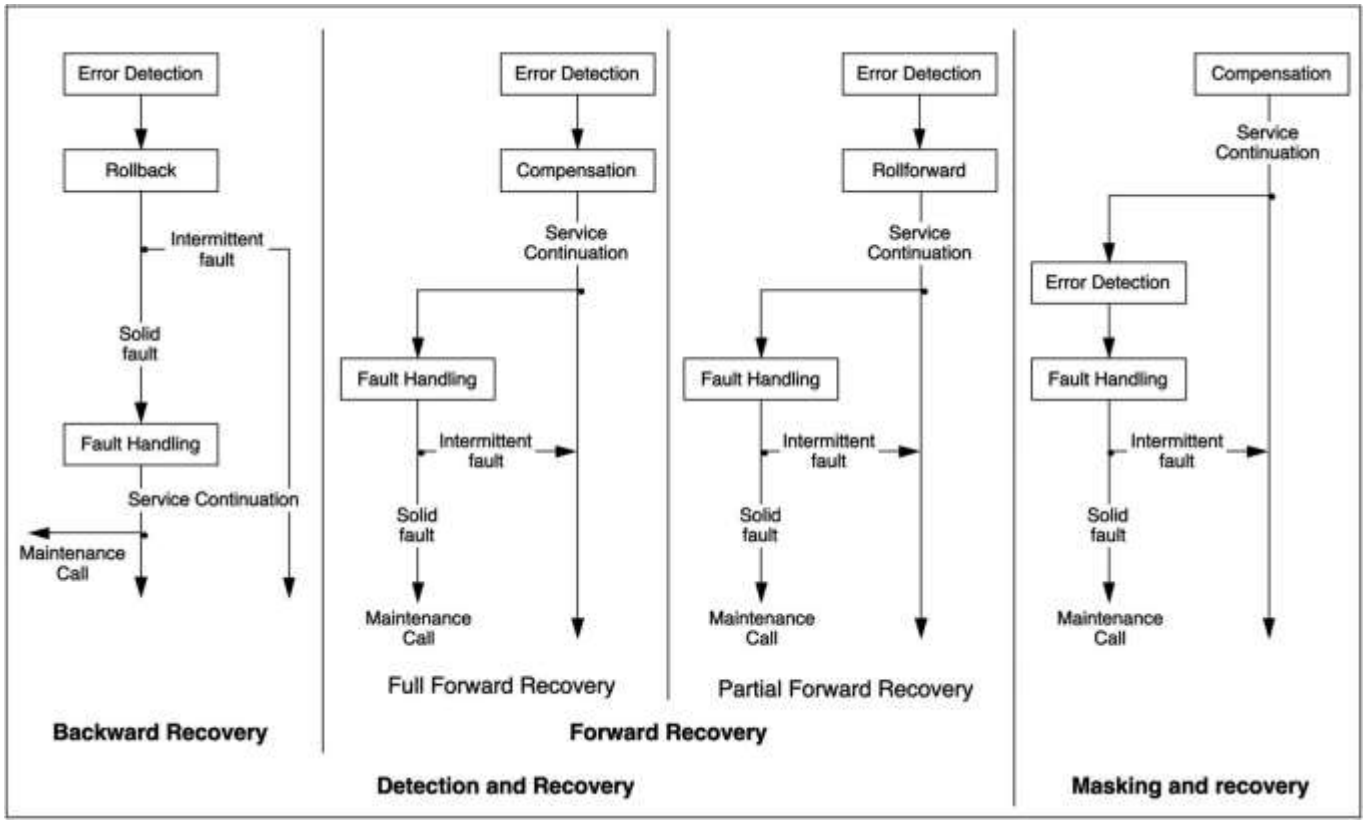
Various starategies for implementing fault tolerance

Choice of the strategy depends upon the underlying fault assumption that is being considered in the development process

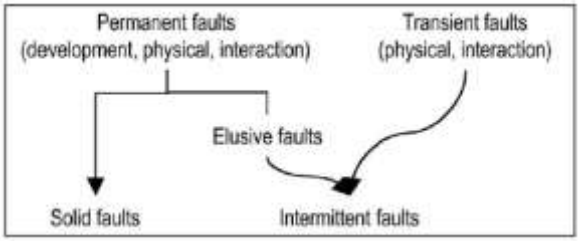
The classes of faults that can actually be tolerated depend on the fault assumption and on the independence of the redundancies with respect to the fault creation and activation

Various starategies for implementing fault tolerance

A. Avizienis, J.C. Laprie, B. Randell, C. Landwehr. Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Transactions on Dependable and Secure Computing, Vol. 1, N. 1, 2004

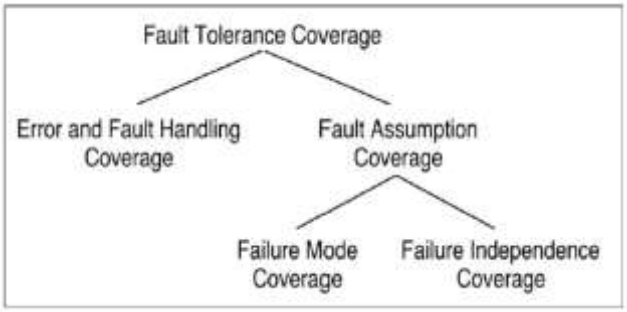


Solid faults: permanent faults whose activation is reproducible
Elusive faults: permanent faults whose activation is not systematically reproducible (e.g, conditions that occur in relation to the system load, pattern sensitive faults in semiconductor memories, …)
Intermittent faults: transient physical faults + elusive development faults



Fault tolerance technique coverage

measure of effectiveness of a fault tolerance technique (probability that it is effective given that a class of errors or faults have occurred, e.g, stuck-at, …)



Lack of fault tolerance technique coverage due to:

1) development faults affecting the fault tolerance mechanism with respect to the fault assumptions stated during the development
(lack of error and fault handling coverage)

2) fault assumptions in the development that differ from the faults really occurring in operation **(lack of fault assumption coverage)** because

- failed components not behaving as assumed
  (lack of failure mode coverage)
  -> conservative fault assumptions (Byzantine faults)
  higher failure mode coverage, and an increase in the redundancy and complexity of fault tolerance mechanisms

- occurrence of common-mode failures (failures caused by similar errors) when independent ones are assumed **(lack of failure independence coverage)**

Observations

Fault assumptions play a fundamental role

Fault tolerance applies to all classes of faults

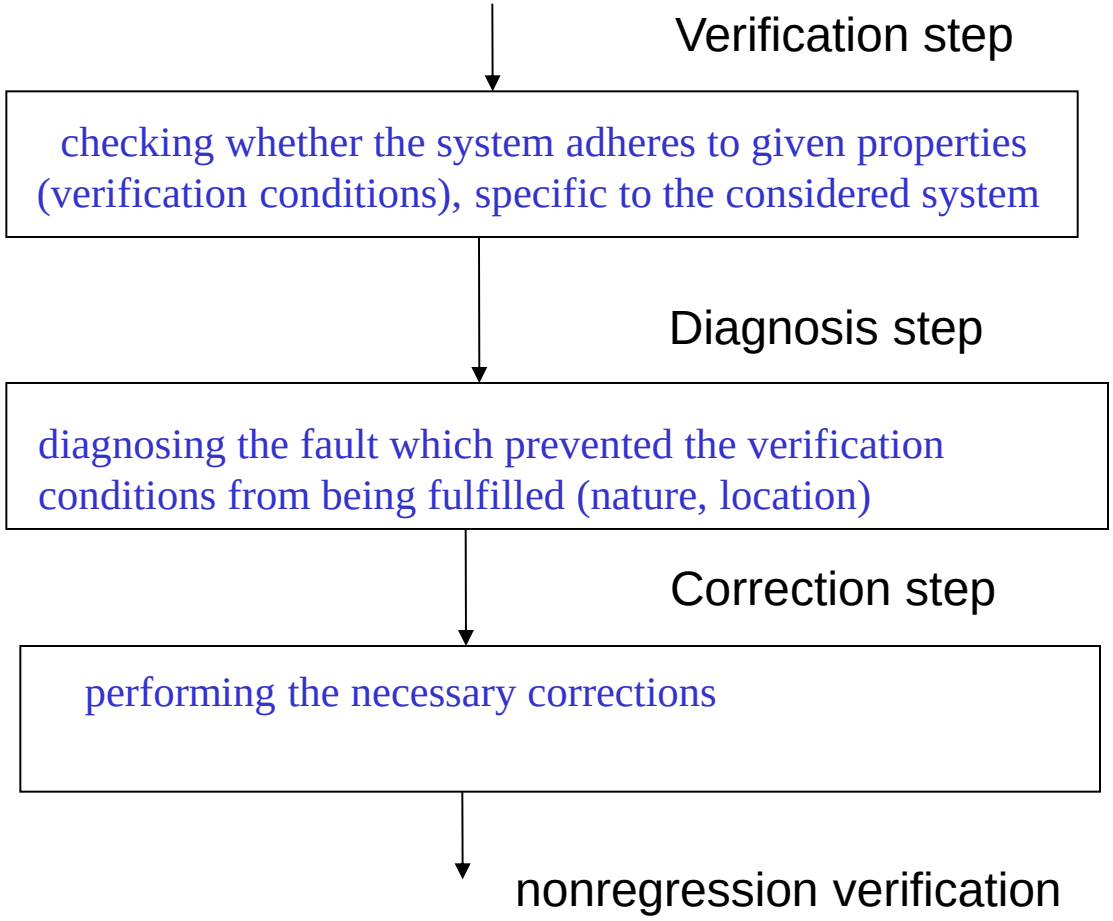
Mechanisms that implements fault tolerance should be protected against the faults that might affect them

Fault removal

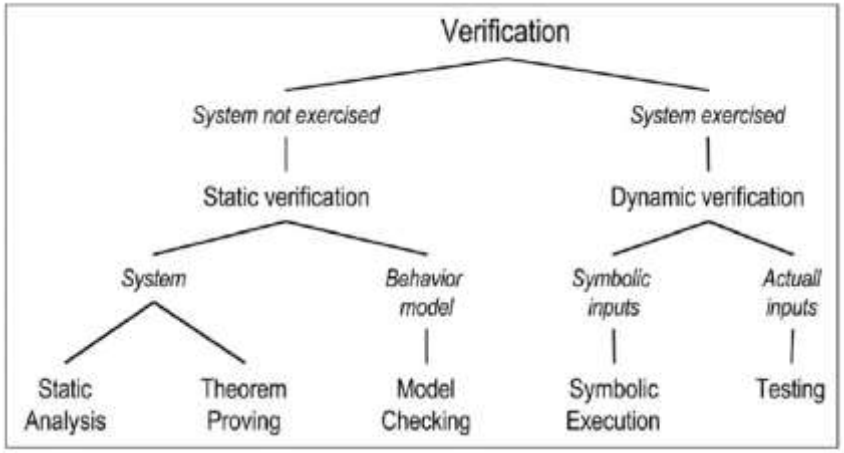
Fault removal techniques

1. During the development phase of the system

Consists of three steps. Moreover, the verification process must be repeated to check that the fault removal had no undesired consequences (nonregression verification)



Verification techniques



Static analysis: inspections, data flow analysis, …

Behaviour model: a model of the system behaviour (generally a state transition model: Petri nets, state automata, …)

Verification techniques:

- applicable to the various forms of the system during the development: prototype, components, …
- applicable to fault tolerance mechanisms
  in this case faults and errors are parts of test patterns (fault injection)

Fault removal techniques

2. During the use phase of the system

corrective maintenence
remove faults that have produced errors and have been reported
Two steps:
1) isolation of the fault by a patch
2) removal of the fault

preventive maintenence
remove faults before they cause errors during normal operation:
1) physical faults occurred
2) development faults that have led to errors in similar systems

Systems can be maintainable on line (without interrupting the service delivery) or offline (during service outage)

Fault forecasting

Fault forecasting techniques

by performing an evaluation of the system behaviour with respect to fault occurrence and activation

- qualitative evaluation
identify, classify and rank the failure modes or the event combinations (component failures or environmental conditions) that would lead to system failure. Example: Failure Mode and Effect Analysis (FMEA)

- **quantitative (or probabilistic) evaluation:**
determining the extent to which dependability attributes (availability, reliability, …) are satisfied (dependability measures)

Activity based on **modelling** or **testing**

Some techniques can be used for both evaluations :
reliability block diagrams or fault trees

Reliability growth models: used to perform reliability predictions from data about past system failures

Important: Evaluation of coverage provided by error and fault handling mechanism (performed through modelling or testing)

Safety critical systems

Often in safety critical systems field, the term **hazard** is used for an error

A **hazard** is a safety error:
threats for human life or the environment (e.g.,radiation in a nuclear power plant)

An active hazard is an incident, that leads to an event called accident

Terminology in industry (nuclear power plant, automotive, railway, aviation, …):
hazard identification, hazard analysis, hazard probability, ….

Hazard analysis is critical thinking about all possible hazard states and all possible consequences

Different perspective:
interested in all the possible error states