Concepts and Mechanisms of Dependable Systems

Summer Term 2007

References and Readings:

Paulo Veríssimo, Luís Rodrigues:
Distributed Systems for System Architects
Kluwer Academic Publishers, Boston, January 2001

Eugen Schäfer: **"Zuverlässigkeit, Verfügbarkeit und Sicherheit in der Elektronik, Eine Brücke von der Zuverlässigkeittheorie zu den Aufgaben der Zuverlässigkeits-praxis"**, 1. Auflage, Vogel Verlag, 1979, ISBN 3-0823-0586-8,

Karl-Erwin Großpietsch: "**Zuverlässigkeitstheoretische Grundlagen**", GMD-Seminar, St. Augustin

Stefan Poledna: "**Lecture on Fault-Tolerant Systems**", Vorlesungsfolien, Institut für Technische Informatik, TU Wien, SoSe 1996

Dependability

Dependability:

The dependability of a system is its ability to deliver specified services to the end users so that they can justifiably rely on and trust the services provided by the system.

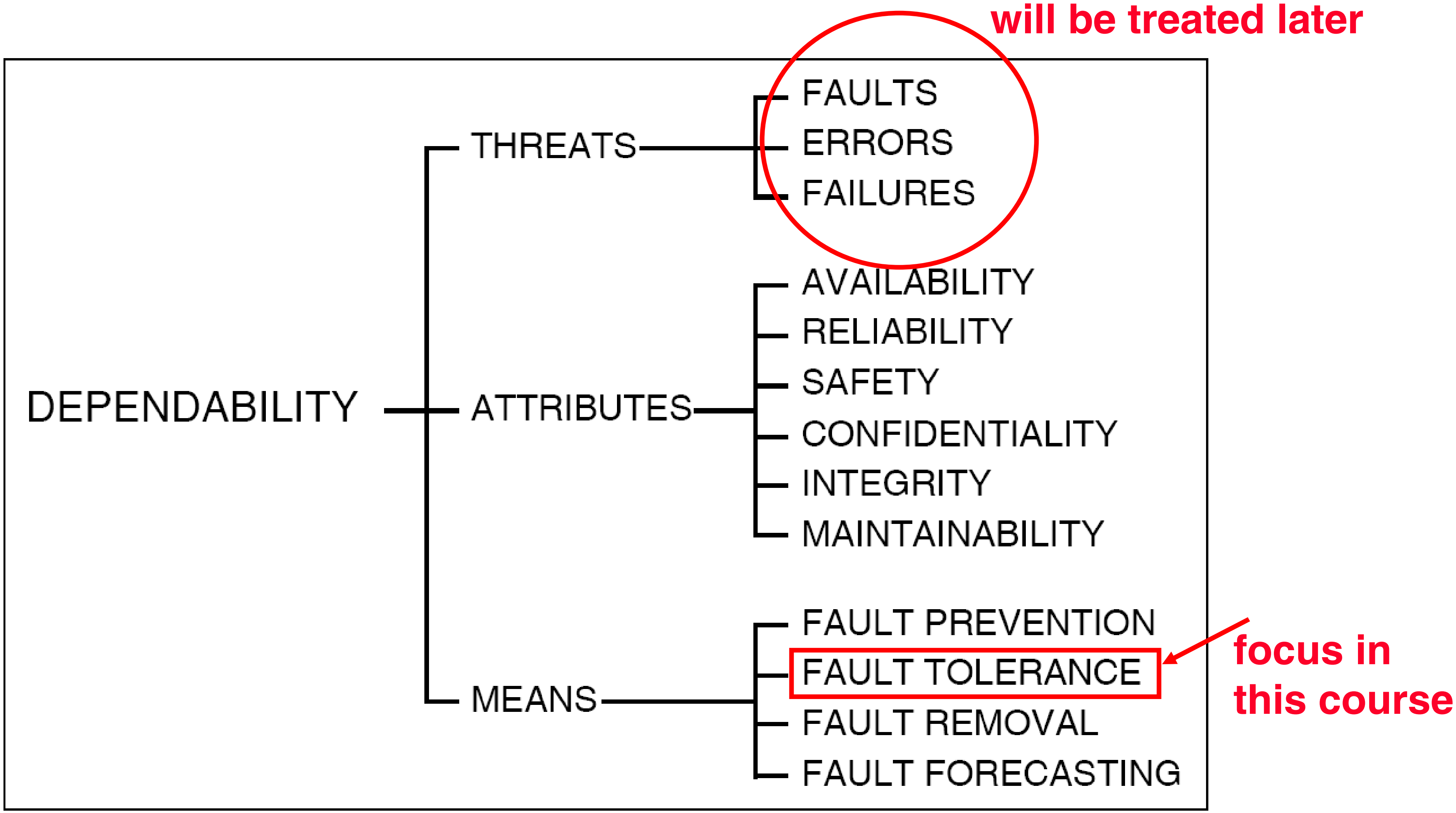
The function or service is the behaviour which can be observed at the interface to other systems which interact with the observed system. Quality referes to the conformance to the specifcations.

Algirdas Avižienis, Jean-Claude Laprie, Brian Randell

Fundamental Concepts of Dependability

UCLA CSD Report no. 010028
LAAS Report no. 01-145
Newcastle University Report no. CS-TR-739

Dependability Tree



Attributes of Dependability

Dependability has several attributes, including reliability, availability, maintainability, security (with aspects like privacy, confidentiality and integrity) and safety.

Availability:	**The availability of a system for a period (0,t) is the probability that the system is available for use at any random time in (0,t).**
Reliability:	**The reliability of a system for a period (0,t) is the probability that the system is continuously operational (i.e., does not fail) in time interval (0,t) given that it is operational at time 0.**
Maintainability:	**The maintainability of a system is a measure of the ability of the system to undergo maintenance or to return to normal operation after a failure.**
Confidentiality:	**The confidentiality of a system is a measure of the degree to which the system can ensure that an unauthorized user will not be able to understand protected information in the system.**
Integrity	**The integrity of a system is the probability that errors or attacks will not lead to damages to the state of the system, including data, code, etc.**
Safety:	**The safety of a system for a period (0,t) is the probability that the system will not incur any catastrophic failures in time interval (0,t).**

Mechanisms of Fault-Tolerance

(Explizit) Fault-Treatment	Fault-Masking
Dynamic Redundancy	Static Redundancy

Error Detection

↓

Fault-Treatment
- **Damage Assessment and Confinement**

  ↓
- **Reconfiguration**

↓

Recovery

e.g. Error Correcting Codes

n-out-of - m - Majority Voting

All Mechanisms of Fault-Tolerance are based on Redundancy

- **Information Redundancy**
- **Component Redundancy**
- **Time Redundancy**

How to determine reliability of composed systems?

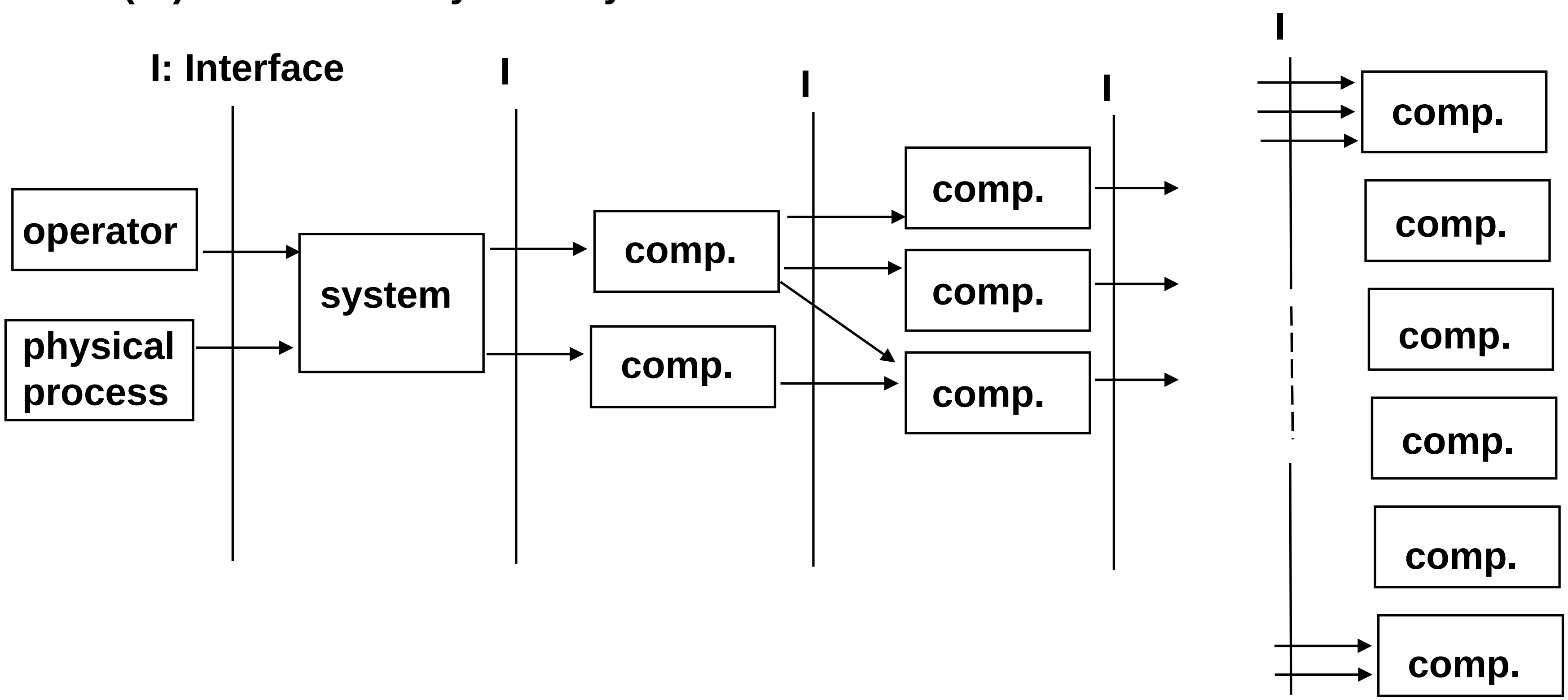
Structure-based modelling:

- **identifiable independent components**
- **every component has an individual reliability**
- **the construction of the model is based on the connection structure**

How to determine reliability of composed systems?

A System is defined by:
- **its structure, i.e.the topology of its components**
- **its behaviour, i.e. by the overall behaviour of all of its components**

systemcomponents are organized in a hierarchical way. This results in a dependency relation (→) between the system layers.



Determining reliability quantitatively by reliability diagrams

Probability of a correctly working component:

For every part of the system we distinguish two states:

• intact (correctly working component)
• failed

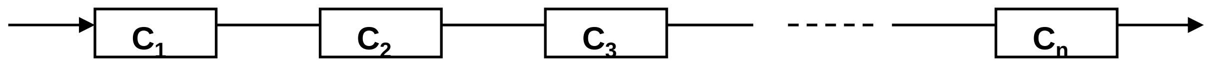
C-Probability (probability of working correctly) of a component is defined by:
Probability that the component exhibits the specified behaviour.

A system is fault-tolerant, if it is showing the overall specified behaviour while some components fail.

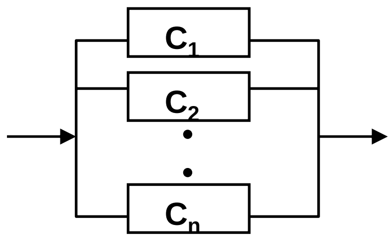
Reliability Diagrams (do not mix up with electrical schematics) :

Abstracting a system in components. Every component has a specified reliability.

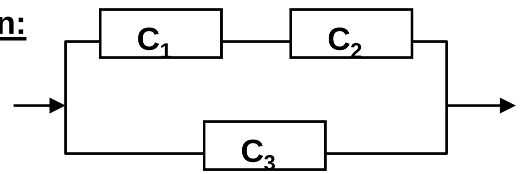
- **serial composition:**



- **parallel composition:**

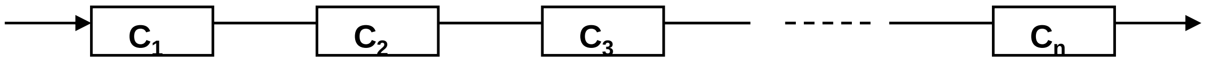


- **serial/parallel composition:**



Probability for a correctly working system:

Serial composition



P_{series} = P (C_1 intact) and P(C_2 intact) and ....P(C_n intact)

Assumption: The properties (C_i intact) (i=1,..,n) are independent.

⇨ P_{series} = P (C_1 intact) • P(C_2 intact) • .... •P(C_n intact)

with p_i : probability of unfailed component (C-probability):

⇨ $P_{series} = p_1 \cdot p_2 \cdot \ldots \cdot p_n$

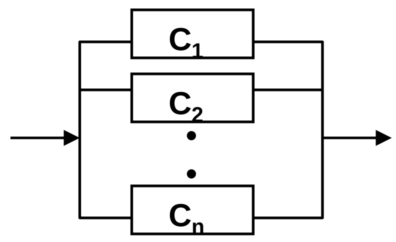
Exampleł:

n identical Components:

P_{series} for p_i^n, n = 5, p_i = 0,99: $P_{series} = 0{,}99^5 = 0{,}95$
P_{series} for p_i^n, n = 5, p_i = 0,70 : $P_{series} = 0{,}70^5 = 0{,}16$

Probability for a correctly working system:

parallel composition



Probability of failure (F-probability) = 1 - C-probability
(correct and failed are complementary events).

$P_{parallel}$ = P (C_1 failed) and P(C_2 failed) and ....P(C_n failed)

Assumption: The properties (C_i failed) (i=1,..,n) are independent..

⇒ **$P_{parallel}$ = P (C_1 failed) • P(C_2 failed) • .... •P(C_n failed)**

p_i : F-probability of component i:

⇒ **$P_{parallel} = 1 - (p_1 \cdot p_2 \cdot \ldots \cdot p_n)$**

Example F-probability:

n identical Components:

$P_{parallel}$ for p_i^n, n = 5, p_i = 1 - 0,99 : $P_{parallel}$ = 1 - $0{,}01^5$ = 1- 0,0000000001 = 0,9999999999
$P_{parallel}$ for p_i^n, n = 5, p_i = 1- 0,70 : $P_{parallel}$ = 1 - $0{,}30^5$ = 1 - 0,00243 = 0,99757

k-out-of-n - Systeme

Systems of n components in which at least k components are working correctly.

Probability that exactly k defined components are correct (components 1,..,k), while the other n-k components failed (componenten k+1,...,n) is given by:

$$P_{k\text{-aus-n}} = p_1 \bullet p_2 \bullet .... \bullet p_k \bullet (1 - p_{k+1}) \bullet (1 - p_{i+2}) \bullet .... \bullet (1 - p_n)$$

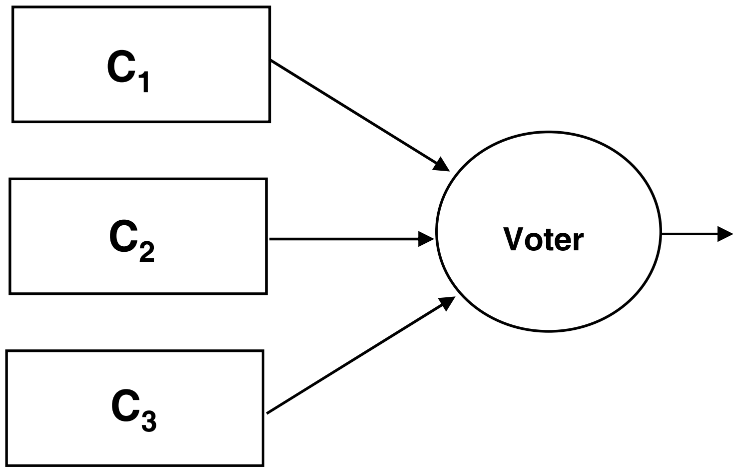
There are $\binom{n}{i}$ possibilities, to select i components out of n components:

$$P_{k\text{-out-of-n}} = \sum_{i=k}^{n} \binom{n}{i} p^i \bullet (1 - p)^{n-i}$$

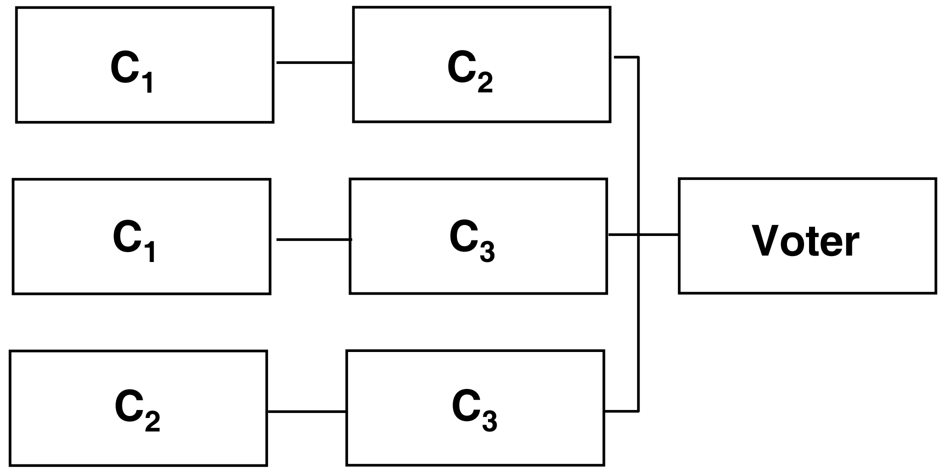
Example: 2-out-of-3 System: $\binom{3}{2} p^2 \bullet (1 - p)^{3-2} + \binom{3}{3} p^3 \bullet (1 - p)^{3-3} = 3 \bullet p^2 \bullet (1 - p) + p^3 \bullet 1$

Example TMR (Triple Modular Redundancy: 2-out-of-3 system)

(electr.) block schematics



reliability diagram



$P_{TMR} = (p^3 + 3\,p^2 \bullet (1 - p)) \bullet p_{voter}$

$p = 0{,}9,\ p_{voter} = 0{,}99$: $P_{TMR} = (0{,}9^3 + 3 \bullet 0{,}9^2 \bullet (1 - 0{,}9)) \bullet 0{,}99$

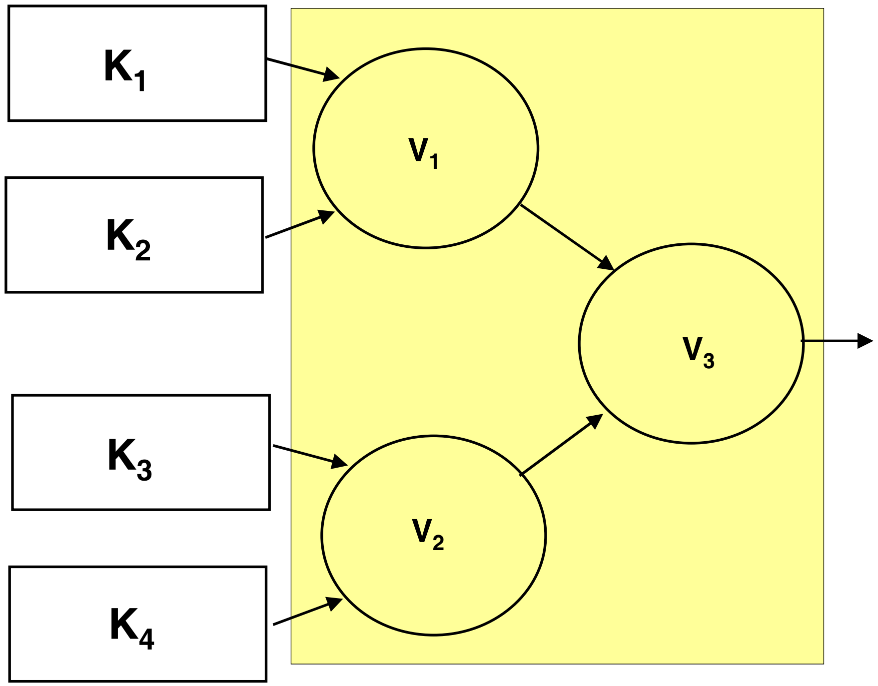
$= (0{,}729 + 3 \bullet 0{,}81 \bullet (1 - 0{,}9)) \bullet 0{,}99$

$= (0{,}729 + 2{,}43 \bullet 0{,}1) \bullet 0{,}99 = 0{,}972 \bullet 0{,}99$

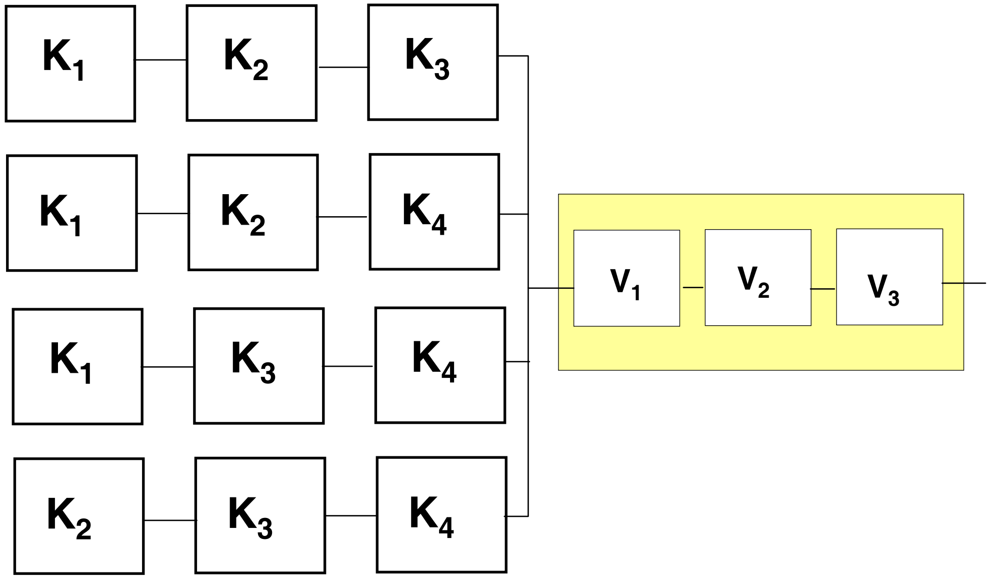
$= 0{,}96228$

Example Pair&Spare (3-out-of-4-System)

(electr.) block schematics



reliability diagram



Example Pair&Spare (3-out-of-4-System)

$P_{P\&S} = (p^4 + 4\, p^3 \bullet (1 - p)\ ) \bullet p_{voter}$

$p = 0{,}9,\ p_{voter} = 0{,}99$: $P_{P\&S} = (0{,}9^4 + 4\bullet 0{,}9^3 \bullet (1 - 0{,}9)) \bullet 0{,}99$

$= (0{,}656 + 4\bullet 0{,}73 \bullet (1 - 0{,}9)) \bullet 0{,}99$

$= (0{,}656 + 2{,}92 \bullet 0{,}1) \bullet 0{,}99 = 0{,}948 \bullet 0{,}99$

$= 0{,}9385$

$p = 0{,}9,\ p_{v1,2} = 0{,}99,\ p_{v3} = 0{,}999$:

$P_{P\&S} = (0{,}9^4 + 4\bullet 0{,}9^3 \bullet (1 - 0{,}9)) \bullet 0{,}99^2 \bullet 0{,}999$

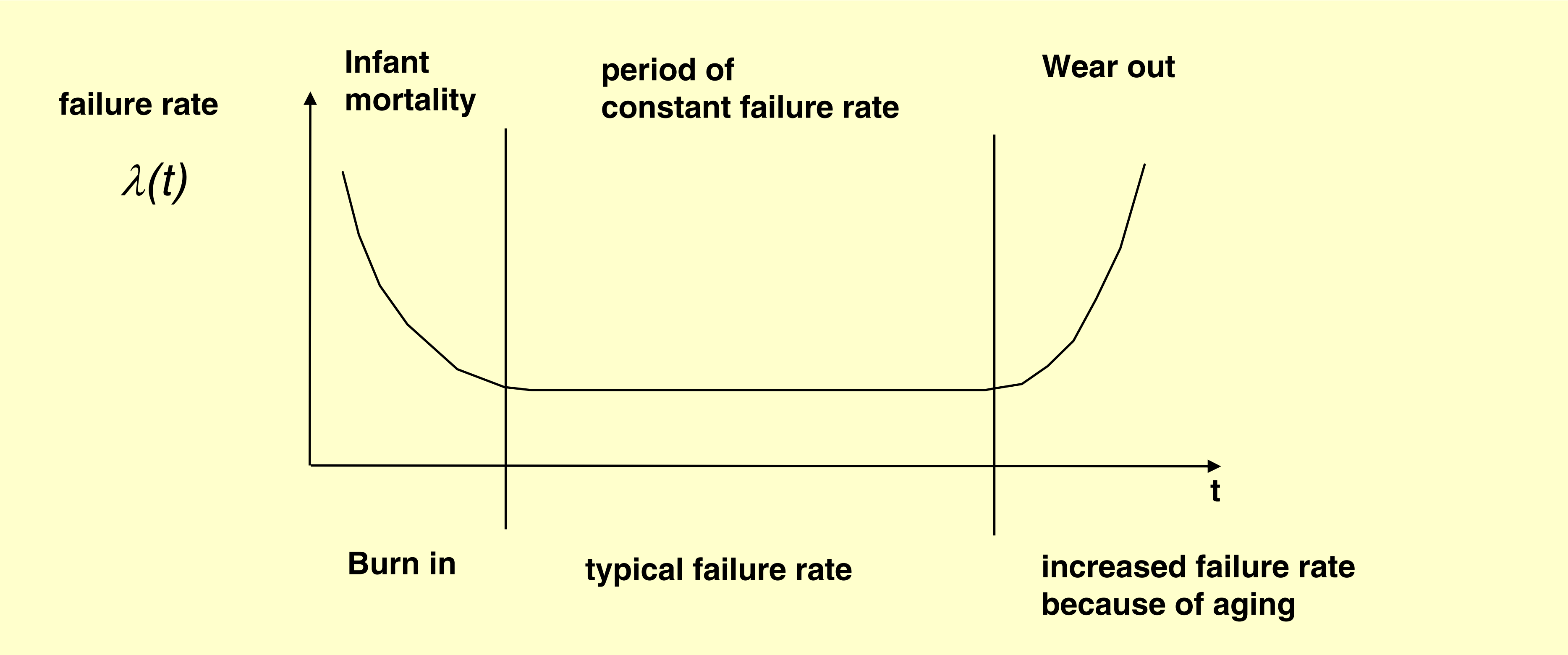
$= (0{,}656 + 4\bullet 0{,}73 \bullet (1 - 0{,}9)) \bullet 0{,}979$

$= (0{,}656 + 2{,}92 \bullet 0{,}1) \bullet 0{,}99 = 0{,}948 \bullet 0{,}9879$

$= 0{,}928$

How to derive the probability of component failure ?

The "bath tub" curve



Typical failure rates:
VLSI-Chip: 10^{-8} failures/h = 1 failure during 115000 years

Note:

The failure rate is defined relative to the number of correct components. In a certain time interval, if always the same number of components fail, the failure rate increases relatitively to the number of correct components that becomes smaller by every failed component.

Dependability measures

Lifetime T
Time interval from the mission start to a non-repairable failure

Probability of failure F(t)
probability to fail in the interval [0,T], $T < t_i$.

Reliability R(t)
Probability that a component did not fail until time t_i.
F(t) is the complement to R(t).

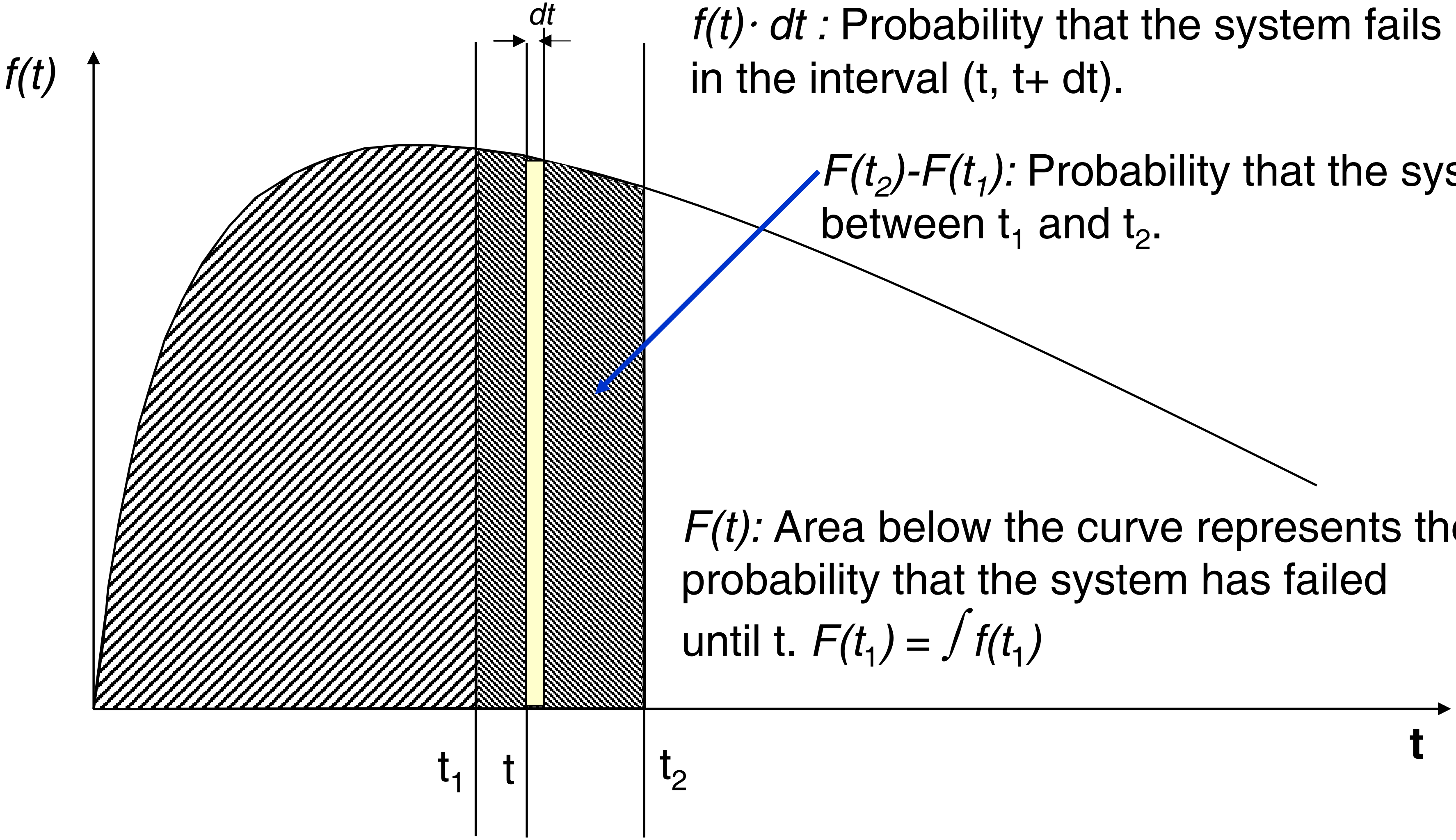
$$R(t) = 1 - F(t)$$

for non repairable systems R(t) is a monotonely decreasing function. $R(0) \leq 1$, $R(\infty) = 0$

Probability density f(t)
f(t) • dt is the probability that a failure occurs in interval (t, t+dt)
f(t) is the probability that failures can be expected within this interval.

$$f(t) = \frac{dF(t)}{dt} = - \frac{dR(t)}{dt}$$

Life time modelling



f(t)· dt : Probability that the system fails in the interval (t, t+ dt).

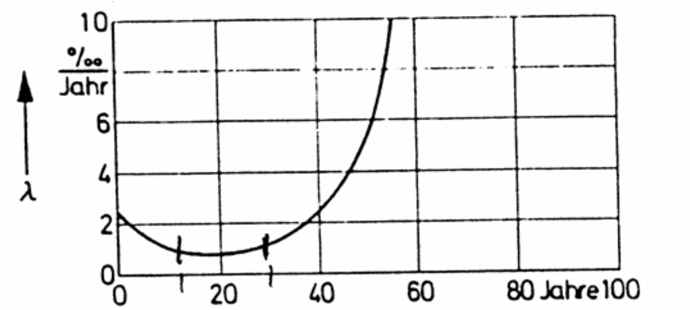
$F(t_2)$-$F(t_1)$: Probability that the system fails between t_1 and t_2.

F(t): Area below the curve represents the probability that the system has failed until t. $F(t_1) = \int f(t_1)$

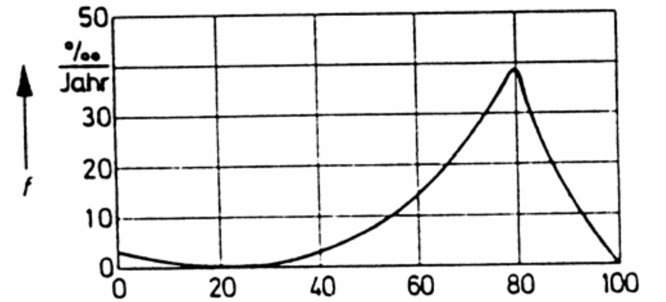
f(t):	PDF: Probability Density Function
F(t):	CDF: Cumulative Density Function. For $t_{\rightarrow\infty}$: $F(t) = 1$

Probability distribution for human life

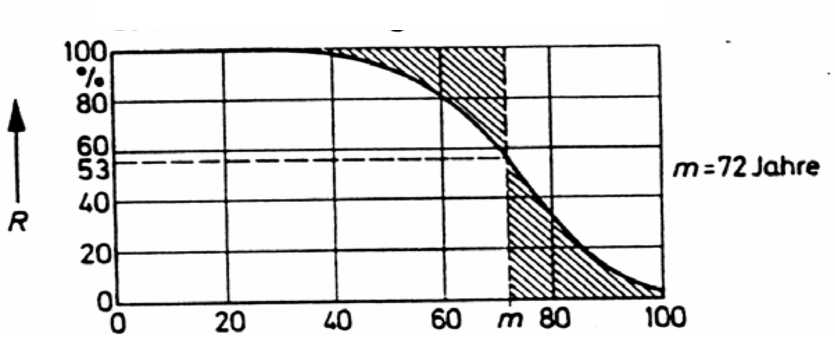
failure rate λ (t)



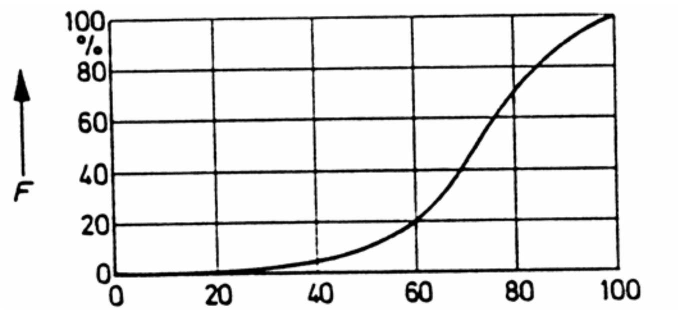
probability density f(t)



Reliability R(t)



failure probablity F(t)

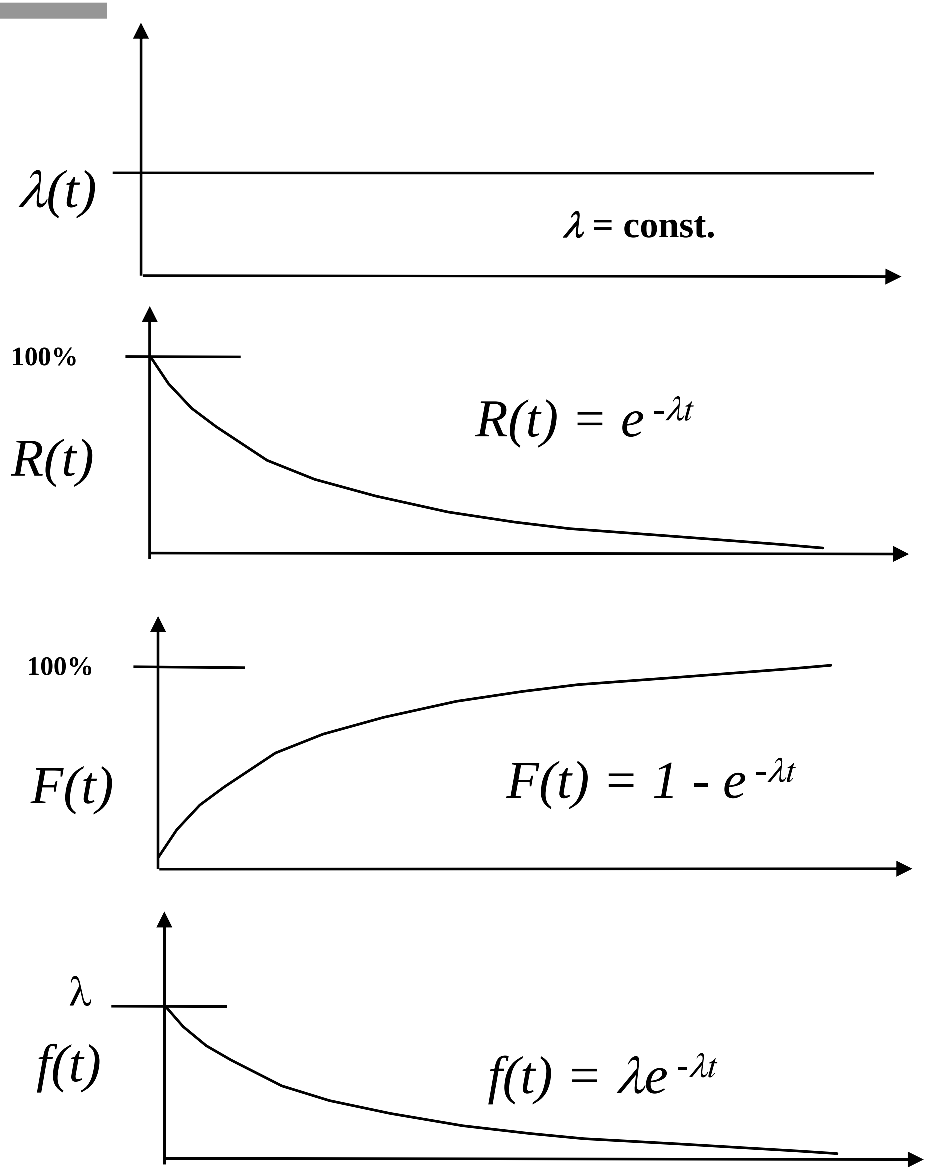


Dependability measures

failure rate λ (t)
number of failures per hour

Remember: **The failure rate is defined relative to the number of correct components. In a certain time interval, if always the same number of components fail, the failure rate increases relatitively to the number of correct components that becomes smaller by every failed component.**

If the failure rate remains constant wrt. the set of correct components, this results in an exponential distribution for the reliability *R(t)*.



Summary of Measures

Parameter	Symbol	Unit
life time	*T*	h
failure probability	*F*	%
reliability	*R*	%
probability density	*f*	%/h
failure rate	λ	1/h

Dependability measures

Assuming λ (t) = const. we have:

$$\frac{1}{\lambda} = MTBF = MTTFF = MTTF$$

MTBF : Mean Time Between Failures

MTTFF: Mean Time To First Failure

MTTF : Mean Time To Failure

Dependability measures

Availability

time in which the system works correct related to the (down-) time when it is repaired.

$$A = \frac{\text{U (Up time)}}{\text{M (Mission time)}}$$

M = U + TR (Repair time)

$$A = \frac{\text{MTBF}}{\text{MTBF + MTTR}}$$

Dependability measures

Availability Classes

class: $\lfloor \log_{10} (1/(1-A)) \rfloor$

1 year = 525600 minutes = 8760 h

system type	non-availability minutes/year	availability %	class
non-adminitrated systems	50 000	~ 90	1
administrated systems	5 000	99	2
well admin. syst.	500	99,9	3
fault-tolerant syst.	50	99,99	4
high availability syst.	5	99,999	5
very high avail. syst.	0,5	99,9999	6
ultra-high avail. syst.	0,05	99,99999	7

Impairments:

Faults, errors, failures

The Cause-Effect-Chain: Classifying Impairments

inherently unavoidable

Fault

failure of a physical component or a faulty statement in a program.

Methods of ***fault avoidance***

this must be treated!
a faulty state must be recovered to a correct state.

a fault may probably cause a erroneous change of the system state.

Error

faulty state, e.g. memory or register contents

Methods of ***fault-tolerance***

an error may cause a change of the system behaviour

this cannot be tolerated because it becomes visible at the system's interface and may be propagated to other systems.

Failure

deviation from the specified behaviour.

cannot be handled by the system.

action from outside needed. May lead to a disaster in safety-critical apps.

The Cause-Effect-Chain: Classifying Impairments

… → fault —*activation*→ error —*propagation*→ failure —*causation*→ fault → … *

transitions:

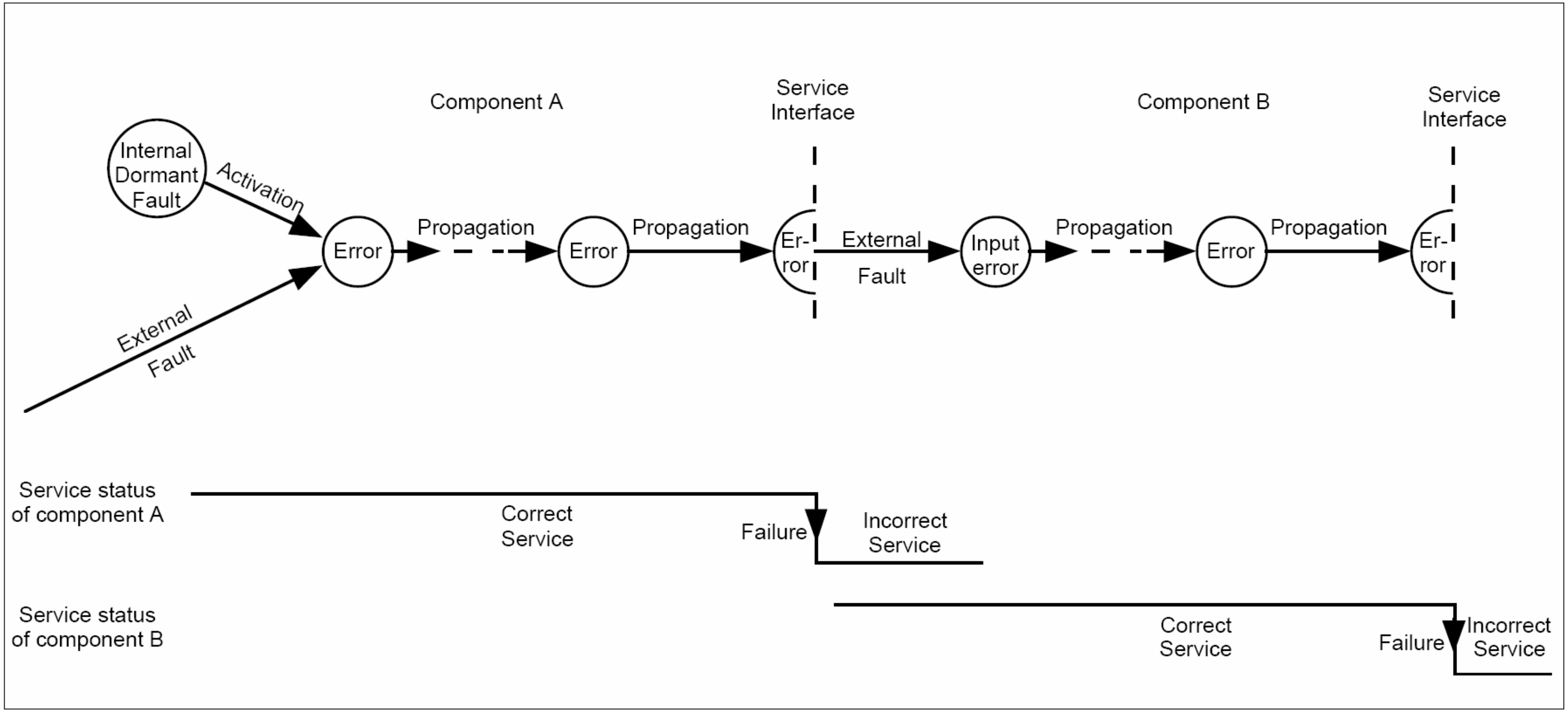
fault → error: **A fault which has not been activated by a computation is called *dormant*. A fault is *activated* if it causes an error.**

error→ failure: **An error is *latent* if it has not yet lead to a failure or has been detected by some error detection mechanism.**
An error is *effective* if it caused a failure.

failure→ fault: **A fault is caused if the error becomes effective and the specified service is affected. This failure can be propagated and appears as a fault on a higher system layer or in a connected component.**

* Algirdas Avižienis, Jean-Claude Laprie, Brian Randell: Fundamental Concepts of Dependability

The Cause-Effect-Chain: Classifying Impairments



*

Error Propagation

* Algirdas Avižienis, Jean-Claude Laprie, Brian Randell: Fundamental Concepts of Dependability