

SE3910 – REAL TIME SYSTEMS

~~MISRA Coding Standards~~

Real time SW

Qualities

ROADMAP

- Today
 - Real Time Software Qualities
- Monday
 - Structured Design and Data Flow Diagrams
- Wednesday
 - Performance Analysis

OBJECTIVES

- Explain the difference between internal and external qualities of software
- List the 8 qualities of real-time software
- Explain how one might assess the qualities of real time software
- Explain the concept of software reliability
- Explain the exponential model of software reliability
- Explain the reliability curves typically exhibited by software
- Calculate the reliability of a software system at a given time
- Explain how one might measure the 8 qualities of real time software

WHAT ARE INTERNAL AND EXTERNAL QUALITIES OF SOFTWARE?

- Discussion

Internal Qualities
Source Code

Design

Reliability

Reusability

INTERNAL AND EXTERNAL QUALITIES

- External Qualities

- Qualities of the software which are observable to the user

SW user

- Internal Qualities

- Qualities which are not directly observable by the user but *impact the SW...*

While important to know, the true importance is to be able to measure these.

THE 8 QUALITIES OF SOFTWARE

- Reliability ✓
- Correctness ✓
- Performance ✓
- Usability ✓
- Interoperability ✓
- Maintainability ✓
- Portability ✓
- Verifiability ✓



*Also
SW architecture
terms*

WHAT IS RELIABILITY?

~~How often it fails.~~

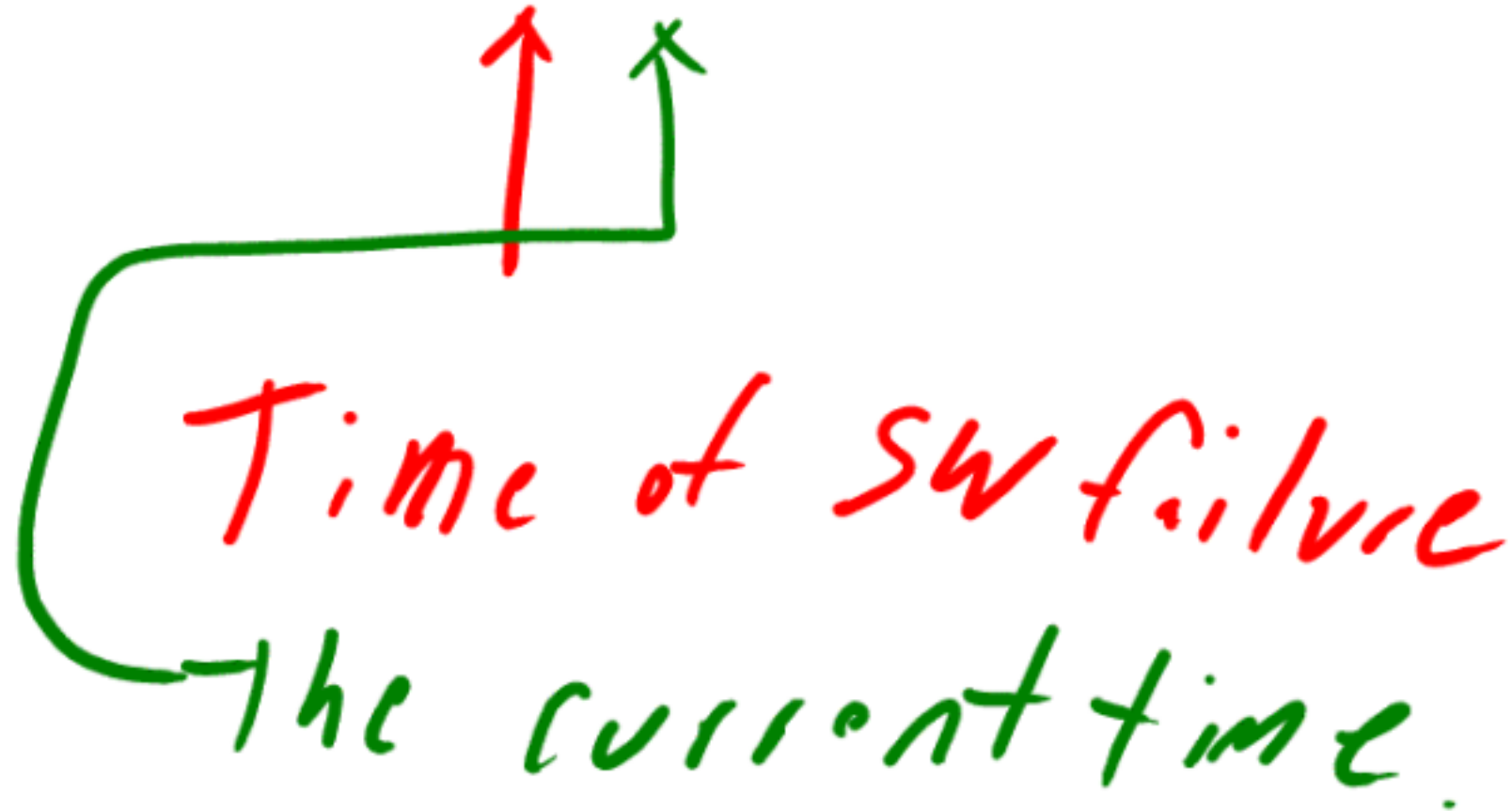
~~⇒ How often it keeps
working?~~

Reliability and
availability

RELIABILITY DEFINITION

- Software Reliability is defined as:
- “the probability of failure-free software operation for a specified period of time in a specified environment.” [ANSI91][Lyu95]

$$r(t) = P(T > t)$$



- An airplane software system has a failure probability of 10^{-6} failures per hour. What is the probability of reliable operation for a 10 hour flight?

For each hour 10^{-6} failures.
 .000001 failures/hour

EXAMPLE

If I have a 10 hr flight,
 what is the the expected
 reliability?

SE3910 REAL TIME SYSTEMS

Reliability $\Rightarrow 1 - P(f)$
 $\Rightarrow 1 - 10^{-6} \times 1$ 1 hour

$\Rightarrow .999999$

10 hours $\Rightarrow 1 - 10^{-6} \times 10 \Rightarrow .99999$

99% reliability per hour

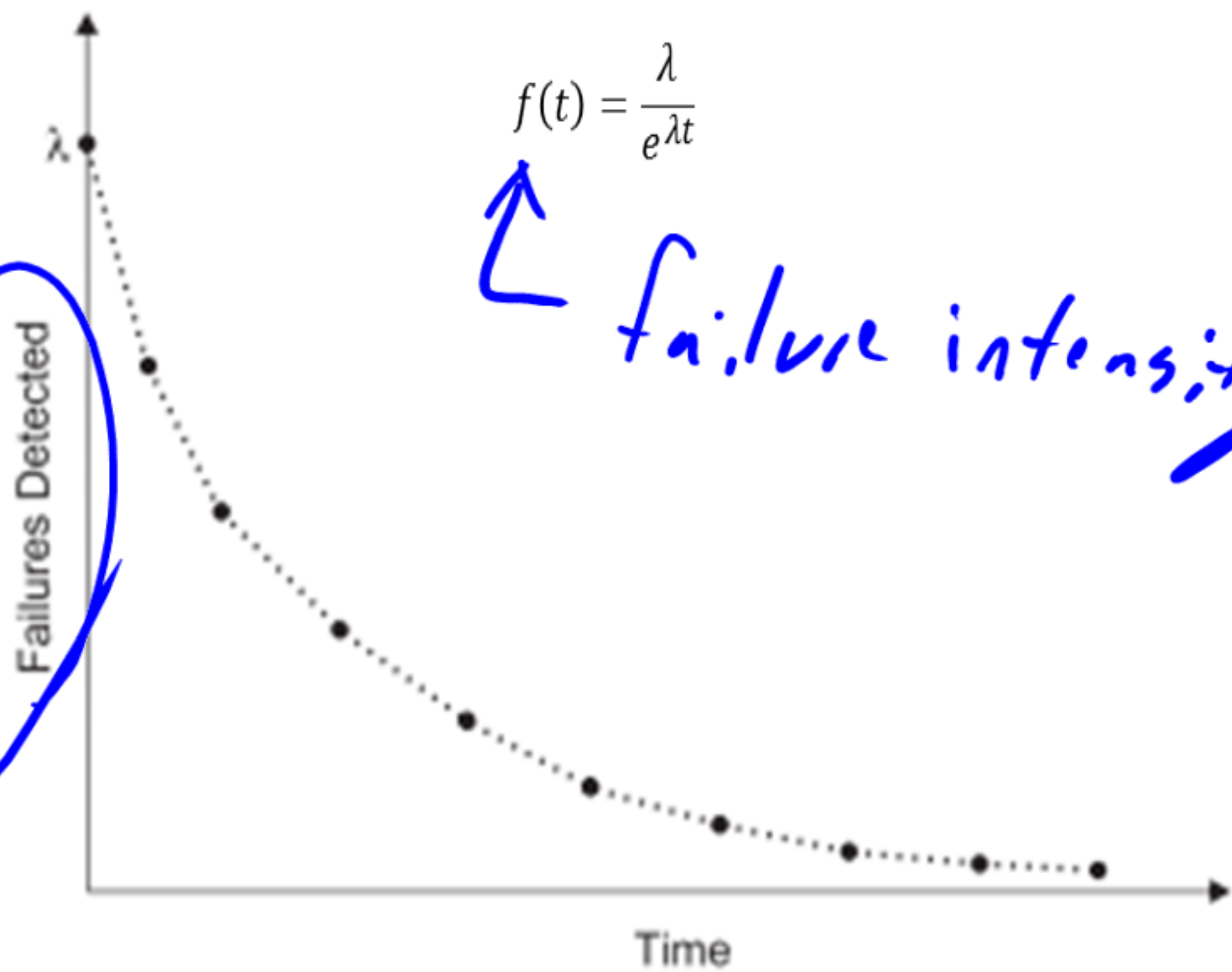
1 hour \Rightarrow .01 probability of failure

1000 hours \Rightarrow (.01 * 1000)

~~1 - 10 = -9~~
Not good

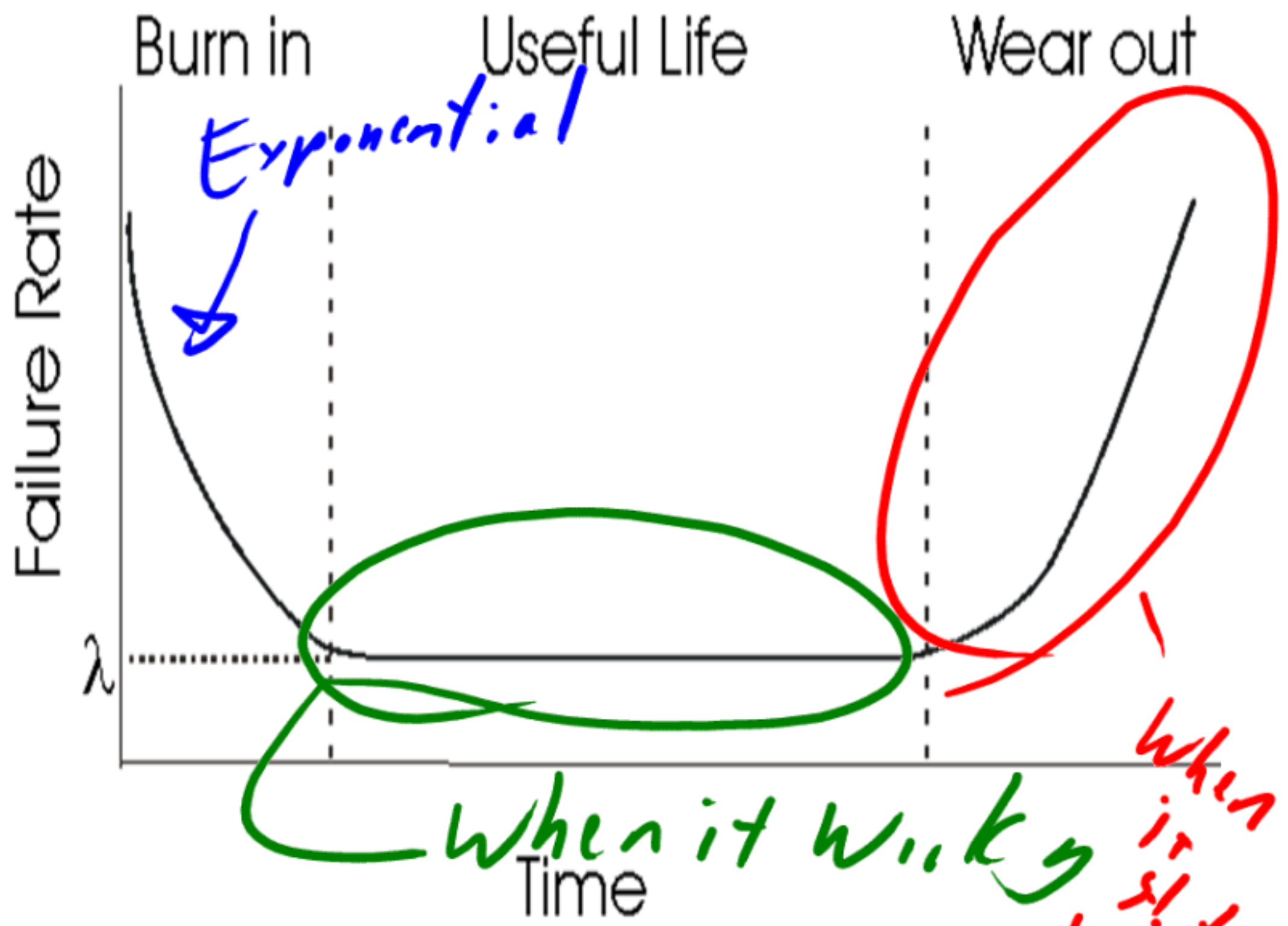
THE EXPONENTIAL FAILURE

FUNCTION

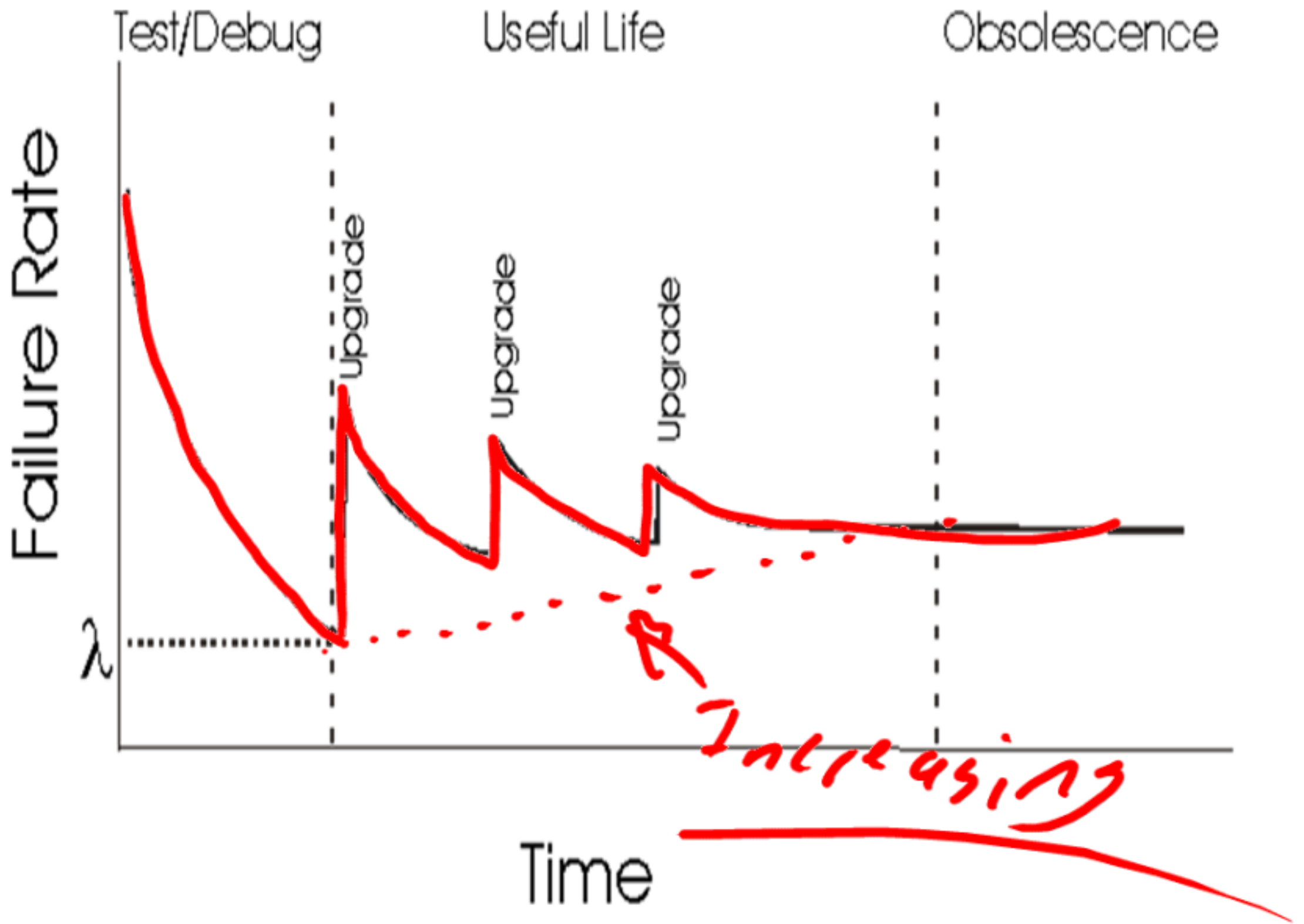


$f(t) = \frac{\lambda}{e^{\lambda t}}$
failure intensity

THE BATHTUB CURVE OF RELIABILITY



TYPICAL SOFTWARE RELIABILITY



WHAT IS MTTTF AND MTBF?

- MTTTF
 - The amount of time that it takes for the first failure to occur
- MTBF
 - The mean time between software failures

- Related to reliability but not the same
- How many deviations from the specified performance are exhibited by the software system?

- ANY deviation.

CORRECTNESS

↳ Reliability usually
~~means~~ means 'substantial
failure'

PERFORMANCE

- An explicit measure for measuring algorithmic performance based on complexity theory

SW Algorithms

- For a real time system, it is extremely important that the system be built estimating performance from the beginning
- Performance must be verified after the system is built

- The ease of use of the system
- Typically requires prototyping of the system for users to evaluate

USABILITY

Building
Mockups



INTEROPERABILITY

- The ability to exist and cooperate with other relevant software
 - Very important in component based systems
- Extremely important for real time systems
- Improved by using open systems

AUTOSAR AUTomotive Open System ARchitecture

ASHRAE **BACnet**TM

MAINTAINABILITY

- The anticipation of change within a software system
 - Evolvability 
 - How easily the system can be changed to accommodate new features or modifications
 - Reparability
 - Allows for the fixing of defects with reasonable effort 

- How easily can the software be adapted to run in different environments
 - Achieved through deliberate design strategies
- Often best met through the usage of an industry standard API

PORTABILITY

✗ Hard to do with real time systems, but it can be done. ✗

VERIFIABILITY

- The degree to which various qualities can be verified
- Main focus for real time systems
 - Deadline satisfaction ✓

WHAT IS HAPPENING WITH EMBEDDED SYSTEMS?

- There is a general shift

Correctness ← Early Systems



Reliability and
maintainability

TABLE 6.2. Software Qualities and Possible Means for Measuring Them

Software Quality	Possible Measurement Approach
Reliability	Probabilistic measures, MTFF, MTBF, heuristic measures
Correctness	Probabilistic measures, MTFF, MTBF
Performance	Algorithmic complexity analysis, simulation, direct measurement
Usability	User feedback from surveys and problem reports
Interoperability	Compliance with relevant open standards
Maintainability	Anecdotal observation of resources spent
Portability	Anecdotal observation of resources spent
Verifiability	Insertion of special monitoring code