


1918

**TALLINNA TEHNIKAÜLIKOOL**

TALLINN UNIVERSITY OF TECHNOLOGY

Department of computer Engineering

ati.ttu.ee



IAF0530/IAF9530

**Süsteemide usaldusväärsus ja veakindlus**

**Dependability and fault tolerance**

Loengud 2 ja 3

Safety, Hazards, Risks

**Gert Jervan**

gert.jervan@pld.ttu.ee

Tallinn University of Technology

Department of Computer Engineering


Estonia

© Gert Jervan, TTÜ/ATI

IAF0530 - Süsteemide usaldusväärsus ja veakindlus

# Course Overview

- ✓ Topic selection:
  - February 22 (via e-mail)
- ✓ Draft of the report (incl. introductory presentation of the topic):
  - March 15
- ✓ Next lecture: March 1

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY


2


© Gert Jervan, TTU/ATI

IAF0530 - Susteemide usaldusväärsus ja veakindlus

# Lecture Outline

- ✓ Dependability
- ✓ Safety Requirements
- ✓ Hazards
- ✓ Hazard Analysis
- ✓ Risks
- ✓ Risk Analysis
- ✓ Risk Management
- ✓ Safety & SILs
- ✓ Risk Reduction & Design



TALLINNA TEHNILIKAÜLICOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

3

# Dependability: an integrating concept

✓ **Dependability** is a property of a system that justifies placing one's reliance on it.

**Dependability**

- attributes
  - Availability
  - Reliability
  - Safety
  - Confidentiality
  - Integrity
  - Maintainability
- means
  - Fault prevention
  - Fault tolerance
  - Fault removal
  - Fault forecasting
- threats
  - Faults
  - Errors
  - Failures

✓ High reliability and high availability

IAF0530 - Süsteemide usaldusväärsus ja keevindus

TALLINNAN TEHNILIKAKOOL  
TALIN UNIVERSITY OF TECHNOLOGY

4

# Threats: Faults, Errors & Failures

```
graph LR; Fault --> Error; Error --> Failure; FaultText[Cause of error  
(and failure)] --> Fault; ErrorText[Unintended  
internal state  
of subsystem] --- Error; FailureText[Deviation of actual service  
from intended service] --> Failure
```

The diagram illustrates the relationship between Fault, Error, and Failure. A central box labeled "Error" contains the text "Unintended internal state of subsystem". An arrow labeled "Fault" points from the left to the "Error" box, with the text "Cause of error (and failure)" below it. An arrow labeled "Failure" points from the "Error" box to the right, with the text "Deviation of actual service from intended service" below it.

© Gert Jervan, TTÜ/ATI

IAF0530 - Süsteemide usaldusväärsus ja veakindlus

5

# The pathology of failure

The diagram illustrates the 'pathology of failure' in a system composed of two components, Component A and Component B, connected by a Service Interface.

**Component A Internal Flow:**

- An **Internal Command Fault** leads to an **Error** via **Activation**.
- An **External Fault** also leads to an **Error** via **Activation**.
- The **Error** propagates to another **Error** via **Propagation**.
- This second **Error** propagates to the **Service Interface** via **Propagation**.

**Service Interface:**

- The **Service Interface** is marked with a vertical line and labeled **Error** on the left and **Input** on the right.
- An **External Fault** can also lead to an **Input** via **Activation**.
- The **Input** leads to an **Input error** via **Activation**.

**Component B Internal Flow:**

- The **Input error** propagates to an **Error** via **Propagation**.
- This **Error** propagates to the **Service Interface** via **Propagation**.
- The **Service Interface** is again marked with a vertical line and labeled **Error** on the left and **Input** on the right.
- The **Input** leads to an **Input error** via **Activation**.

**Service Status Transitions:**

- Service status of component A:** Starts at **Correct Service**. A **Failure** event leads to **Incorrect Service**.
- Service status of component B:** Starts at **Correct Service**. A **Failure** event leads to **Incorrect Service**.

The diagram highlights that failures can originate from internal faults, external faults, or errors propagated through the system's components and interfaces.

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Three-universe model

- ✓ **Physical universe:** where the faults occur
  - Physical entities: semiconductor devices, mechanical elements, displays, printers, power supplies
  - A fault is a physical defect or alteration of some component in the physical universe
- ✓ **Informational universe:** where the error occurs
  - Units of information: bits, data words
  - An error has occurred when some unit of information becomes incorrect
- ✓ **External (user's universe):** where failures occur
  - User sees the effects of faults and errors
  - The failure is any deviation from the desired or expected behavior

PTIS TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

7

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Causes of faults

- ✓ Problems at any stages of the design process can result in faults within the system.

```

graph LR
    SM[Specification Mistakes] --> SF[Software Faults]
    IM[Implementation Mistakes] --> SF
    ED[External Disturbances] --> HF[Hardware Faults]
    CD[Component Defects] --> HF
    SF --> E[Errors]
    HF --> E
    E --> SysF[System Failures]
  
```

PTIS TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

8

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Causes of faults, cont.

- ✓ **Specification mistakes**
  - Incorrect algorithms, architectures, hardware or software design specifications
    - Example: the designer of a digital circuit incorrectly specified the timing characteristics of some of the circuit's components
- ✓ **Implementation mistakes**
  - Implementation: process of turning the hardware and software designs into physical hardware and actual code
  - Poor design, poor component selection, poor construction, software coding mistakes
    - Examples: software coding error, a printed circuit board is constructed such that adjacent lines of a circuit are shorted together

PTIS TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

9

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Causes of faults, cont.

- ✓ **Component defects**
  - Manufacturing imperfections, random device defects, component wear-out
  - Most commonly considered causes of faults
    - Examples: bonds breaking within the circuit, corrosion of the metal
- ✓ **External disturbance**
  - Radiation, electromagnetic interference, operator mistakes, environmental extremes, battle damage
    - Example: lightning

PTIS TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

10

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Elementary fault classes

```

graph LR
    FAULTS --> PCO[PHASE OF CREATION OR OCCURRENCE]
    FAULTS --> SB[SYSTEM BOUNDARIES]
    FAULTS --> D[DOMAIN]
    FAULTS --> PC[PHENOMENOLOGICAL CAUSE]
    FAULTS --> INT[INTENT]
    FAULTS --> P[PERSISTENCE]

    PCO --> DF[DEVELOPMENTAL FAULTS]
    PCO --> OF[OPERATIONAL FAULTS]

    SB --> IF[INTERNAL FAULTS]
    SB --> EF[EXTERNAL FAULTS]

    D --> HF[HARDWARE FAULTS]
    D --> SF[SOFTWARE FAULTS]

    PC --> NF[NATURAL FAULTS]
    PC --> HMF[HUMAN-MADE FAULTS]

    INT --> ADF[ACCIDENTAL OR NON-MALICIOUS DELIBERATE FAULTS]
    INT --> DMF[DELIBERATELY MALICIOUS FAULTS]

    P --> PF[PERMANENT FAULTS]
    P --> TF[TRANSIENT FAULTS]
  
```

PTIS TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

11

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Classification of faults

```

graph TD
    FAULTS --> DEV[DEVELOPMENTAL]
    FAULTS --> OP[OPERATIONAL]

    DEV --> DEV_SOFTWARE[SOFTWARE]
    DEV --> DEV_HARDWARE[HARDWARE]
    OP --> OP_HARDWARE[HARDWARE]
    OP --> OP_SOFTWARE[SOFTWARE]
    OP --> OP_EXTERNAL[EXTERNAL]

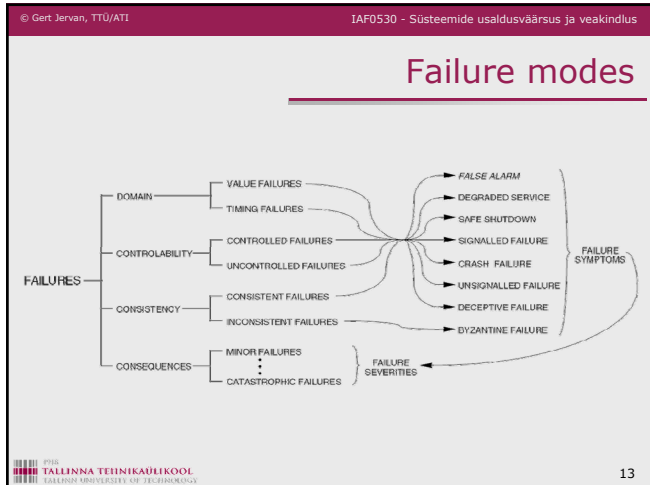
    DEV_SOFTWARE --> DEV_SOFTWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE --> DEV_HARDWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE --> DEV_HARDWARE_HARDWARE[HARDWARE]
    OP_HARDWARE --> OP_HARDWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE --> OP_HARDWARE_HARDWARE[HARDWARE]
    OP_SOFTWARE --> OP_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL --> OP_EXTERNAL_SOFTWARE[SOFTWARE]
    OP_EXTERNAL --> OP_EXTERNAL_HARDWARE[HARDWARE]

    DEV_SOFTWARE_SOFTWARE --> DEV_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_SOFTWARE --> DEV_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_HARDWARE --> DEV_HARDWARE_HARDWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_HARDWARE --> DEV_HARDWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_HARDWARE_SOFTWARE --> OP_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE_HARDWARE --> OP_HARDWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE_HARDWARE --> OP_HARDWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_SOFTWARE_SOFTWARE --> OP_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_SOFTWARE --> OP_EXTERNAL_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE --> OP_EXTERNAL_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE[HARDWARE]

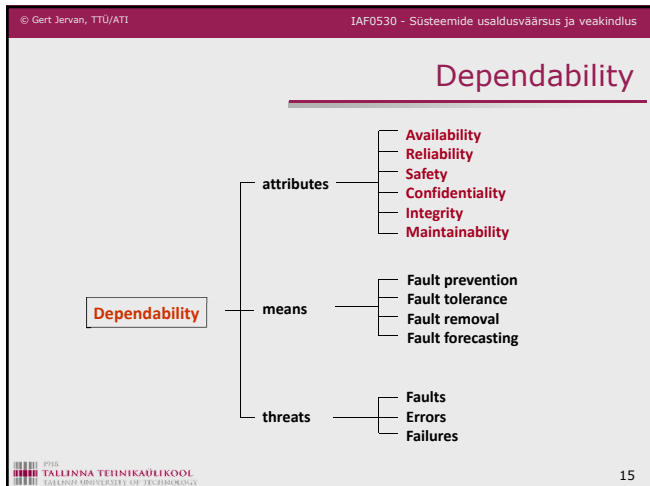
    DEV_SOFTWARE_SOFTWARE_SOFTWARE --> DEV_SOFTWARE_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_SOFTWARE_SOFTWARE --> DEV_HARDWARE_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_HARDWARE_SOFTWARE --> DEV_HARDWARE_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_HARDWARE_HARDWARE --> DEV_HARDWARE_HARDWARE_HARDWARE_SOFTWARE[SOFTWARE]
    DEV_HARDWARE_HARDWARE_HARDWARE --> DEV_HARDWARE_HARDWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_HARDWARE_SOFTWARE_SOFTWARE --> OP_HARDWARE_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE_SOFTWARE_HARDWARE --> OP_HARDWARE_SOFTWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE_SOFTWARE_HARDWARE --> OP_HARDWARE_SOFTWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_HARDWARE_HARDWARE_SOFTWARE --> OP_HARDWARE_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE_HARDWARE_SOFTWARE --> OP_HARDWARE_HARDWARE_SOFTWARE_HARDWARE[HARDWARE]
    OP_HARDWARE_HARDWARE_HARDWARE --> OP_HARDWARE_HARDWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_HARDWARE_HARDWARE_HARDWARE --> OP_HARDWARE_HARDWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_SOFTWARE_SOFTWARE_SOFTWARE --> OP_SOFTWARE_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_SOFTWARE_SOFTWARE --> OP_EXTERNAL_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_SOFTWARE_HARDWARE --> OP_EXTERNAL_SOFTWARE_SOFTWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_SOFTWARE_HARDWARE --> OP_EXTERNAL_SOFTWARE_SOFTWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_SOFTWARE_HARDWARE --> OP_EXTERNAL_SOFTWARE_SOFTWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_HARDWARE_SOFTWARE --> OP_EXTERNAL_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_SOFTWARE --> OP_EXTERNAL_HARDWARE_SOFTWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_SOFTWARE --> OP_EXTERNAL_HARDWARE_SOFTWARE_SOFTWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_HARDWARE_SOFTWARE --> OP_EXTERNAL_HARDWARE_SOFTWARE_SOFTWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_SOFTWARE --> OP_EXTERNAL_HARDWARE_SOFTWARE_SOFTWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_SOFTWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_SOFTWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_SOFTWARE_HARDWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_HARDWARE_SOFTWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_HARDWARE_SOFTWARE_HARDWARE[HARDWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_HARDWARE_SOFTWARE_HARDWARE_SOFTWARE[SOFTWARE]
    OP_EXTERNAL_HARDWARE_HARDWARE --> OP_EXTERNAL_HARDWARE_HARDWARE_HARDWARE_SOFTWARE_HARDWARE_HARDWARE[HARDWARE]
  
```

PTIS TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

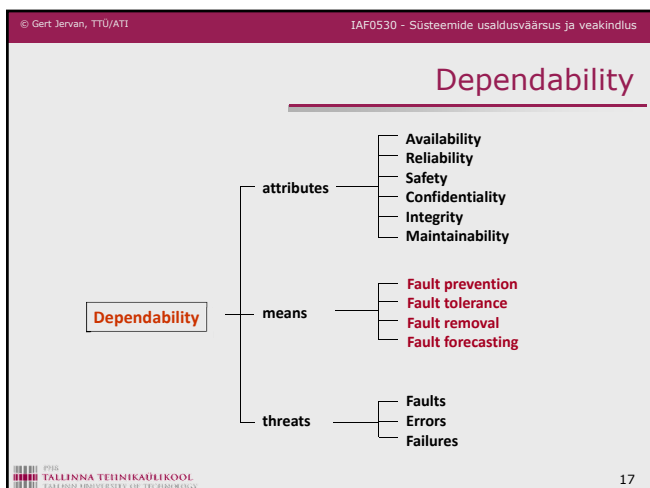
12



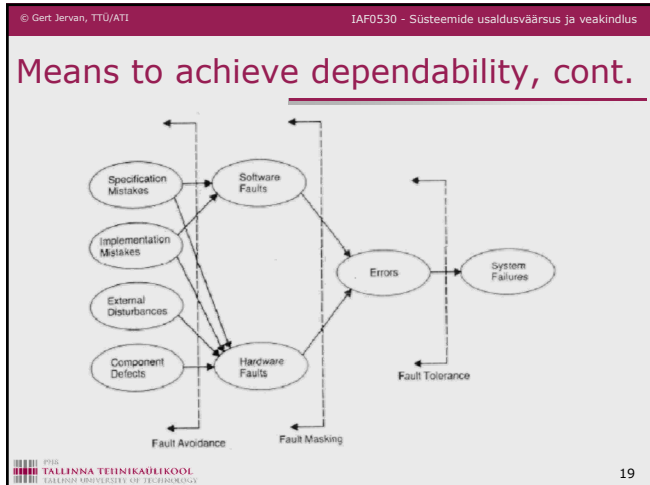
- © Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus
- ## Failure modes, cont.
- ✓ **Failure domain**
    - Value failures : incorrect value delivered at interface
    - Timing failures : right result at the wrong time (usually late)
  - ✓ **Failure consistency**
    - Consistent failures : all nodes see the same, possibly wrong, result
    - Inconsistent failures : different nodes see different results
  - ✓ **Failure consequences**
    - Benign failures : essentially loss of utility of the system
    - Malign failures : significantly more than loss of utility of the system; catastrophic, e.g. airplane crash
  - ✓ **Failure oftenness (failure frequency and persistency)**
    - Permanent failure : system ceases operation until it is repaired
    - Transient failure : system continues to operate
      - Frequently occurring transient failures are called intermittent
- PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY
- 14



- © Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus
- ## Dependability attributes
- ✓ **Availability**: readiness for correct service
  - ✓ **Reliability**: continuity of correct service
  - ✓ **Safety**: absence of catastrophic consequences on the user(s) and the environment
  - ✓ **Confidentiality**: absence of unauthorized disclosure of information
  - ✓ **Integrity**: absence of improper system alterations
  - ✓ **Maintainability**: ability to undergo, modifications, and repairs
  - ✓ **Security**: the concurrent existence of (a) availability for authorized users only, (b) confidentiality, and (c) integrity with 'improper' taken as meaning 'unauthorized'.
- PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY
- 16



- © Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus
- ## Means to achieve dependability
- ✓ **Fault-prevention**: how to prevent, by **construction**, fault occurrence.
  - ✓ **Fault-tolerance**: how to provide, by **redundancy**, service complying with the specification in spite of faults having occurred or occurring.
  - ✓ **Fault-removal**: how to minimize, by **verification and validation**, the presence of latent faults.
  - ✓ **Fault-forecasting**: how to minimize, by **evaluation**, the presence, the creation and the consequences of faults.
- PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY
- 18



© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Fault prevention

- ✓ Attained by quality control techniques
  - Software
    - Structured/object oriented programming
    - Information hiding
    - Modularization
  - Hardware
    - Rigorous design rules
    - Shielding
    - Radiation hardening
    - "Foolproof" packaging
- ✓ Note:
  - Malicious faults can also be prevented; Example: firewalls

TALLINNA TEHNIKAKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

20

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Fault tolerance

- ✓ **Fault tolerance** is the ability of a system to continue to perform its functions (deliver correct service), even when one or more components have failed.
  - **Masking**: the use of sufficient redundancy may allow recovery without explicit error detection.
  - **Reconfiguration**: eliminating a faulty entity from a system and restoring the system to some operational condition or state.
    - Error **detection**: recognizing that an error has occurred
    - Error **location**: determining which module produced the error
    - Error **containment**: preventing the errors from propagating
    - Error **recovery**: regaining operational status

TALLINNA TEHNIKAKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

21

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### The concept of redundancy

- ✓ Definition
  - **Redundancy** is the addition of information, resources, or time beyond what is needed for normal system operation.
- ✓ Digital filter example
  - Software redundancy: lines of software to perform a validity checks
  - Hardware redundancy: if more memory needed for the software checks
  - Time redundancy: each filter calculation performed twice to detect faults
  - Information redundancy: output using with a simple parity bit

Input → Analog-to-digital converter → Microprocessor → Digital-to-analog converter → Output

TALLINNA TEHNIKAKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

22

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Error detection

- ✓ Two ways to detect errors:
  1. a priori knowledge about intended state
  2. comparing results of two redundant computational channels
- ✓ Notes
  - Errors can happen in the **value domain** and/or in the **time domain**.
  - The probability that an error is detected, provided it is present, is called the **error detection coverage**.
  - The time interval between the start of an error and the detection of an error is the **error detection latency**.

TALLINNA TEHNIKAKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

23

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### A priori Knowledge flexibility vs. error-detection coverage

- ✓ **Syntactic knowledge about code space**
  - Parity bits, CRC
- ✓ **Assertions and acceptance tests**
  - Valid data values, properties of the controlled object
    - Development of physical processes, plausibility of data sets
- ✓ **Activation patterns of computation**
  - Regularity in execution pattern, e.g., frequency of updates
    - Limited by the update frequency and clock synchronisation
    - Event every second, on the second → detect missing event
- ✓ **Worst case execution time of tasks**
  - Must be known to calculate real-time schedules
  - A priori information about the execution of a task can be used for detecting task errors

TALLINNA TEHNIKAKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

24

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Redundant Computations

Type of Redundancy	Implementation	Type of Detected Errors
Time redundancy	Same software executed on the same hardware during two different time-intervals	Errors caused by transient physical faults in hardware with a duration less than one execution time slot
Hardware redundancy	The same software executes on two independent hardware channels	Errors caused by transient and permanent physical hardware errors
Diverse software on the same hardware	Different software versions are executed on the same hardware during two different time intervals	Errors caused by independent software faults and transient physical faults in the hardware with a duration less than one execution time slot
Diverse software on diverse hardware	Two different versions of software are executed on two independent hardware channels	Errors caused by independent software faults and by transient and permanent physical hardware faults

PTIS TALLINNA TEHNIKALÜKÜÜL TALLINN UNIVERSITY OF TECHNOLOGY 25

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Recovery

- ✓ Definition
  - **Recovery** transforms a system state that contains one or more errors and (possibly) faults into a state without detected errors and faults that can be activated again.
- ✓ Consists of
  - Error handling
    - **Rollback**: returning to a saved state (checkpoint)
    - **Compensation**: enough redundancy to eliminate the error
    - **Rollforward**: the state without errors is a new state
  - Fault handling
    - **Fault diagnosis**: identifies the cause of errors, location and type
    - **Fault isolation**: physical or logical exclusion of the faulty components
    - **System reconfiguration**: switches in spares or re-assigns tasks
    - **System reinitialization**: checks, updates and records the new configuration

PTIS TALLINNA TEHNIKALÜKÜÜL TALLINN UNIVERSITY OF TECHNOLOGY 26

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Fault removal

- ✓ **Verification**: "Are we building the system right?"
  - Static: does not exercise the system
    - Static analysis: inspections, walkthroughs, model checking
  - Dynamic
    - Symbolic execution: inputs are symbolic
    - Testing: actual inputs
  - Fault injection
- ✓ **Validation**: "Are we building the right system?"
  - Checking the specification

PTIS TALLINNA TEHNIKALÜKÜÜL TALLINN UNIVERSITY OF TECHNOLOGY 27

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Fault Forecasting

- ✓ Evaluation of the system behavior with respect to fault occurrence
  - **Qualitative** evaluation
    - Identifies, classifies, ranks the failure modes or the event combinations that lead to system failures
    - Example methods: Failure mode and effect analysis, fault-tree analysis
  - **Quantitative** evaluation
    - Evaluates in terms of probabilities the extent to which some of the dependability are satisfied (measures dependability)
    - Example methods: Markov chains, reliability block diagrams

PTIS TALLINNA TEHNIKALÜKÜÜL TALLINN UNIVERSITY OF TECHNOLOGY 28

PTIS TALLINNA TEHNIKALÜKÜÜL TALLINN UNIVERSITY OF TECHNOLOGY Department of computer Engineering ati.ttu.ee

## Safety Requirements

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Definitions of Safety

- ✓ Informally
  - "Nothing bad will happen"
- ✓ N. Leveson, Safeware
  - "Freedom from accidents or losses"
  - But no system can be completely safe in absolute sense...
  - Focus is on making systems safe enough, given limited resources
  - Emphasis on accidents, rather than risk
- ✓ N. Storey, Safety-Critical Computer Systems:
  - "System will not endanger human life or environment"
  - More emphasis on removing hazards than actual accidents...
- ✓ Safety-critical system
  - System that has the potential to cause accidents

PTIS TALLINNA TEHNIKALÜKÜÜL TALLINN UNIVERSITY OF TECHNOLOGY 30

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Safety requirements

- ✓ In order to determine safety requirements:
  - Identification of the hazards associated with the system
  - Classification of these hazards
  - Determination of methods for dealing with the hazards
  - Assignment of appropriate reliability and availability requirements
  - Determination of an appropriate safety integrity level
  - Specification of development methods appropriate to this integrity level

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

31

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## The Role of Standards

- ✓ Helping staff to ensure that a product meets a certain level of quality
- ✓ Helping to establish that a product has been developed using methods of known effectiveness
- ✓ Promoting a uniformity of approach between different teams
- ✓ Providing guidance on design and development techniques
- ✓ Providing some legal basis in the case of a dispute

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

32

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Conflicting requirements

- ✓ High performance v low cost
- ✓ Reliability  $\neq$  safety

BUT

- ✓ System must be reliable AND safe
- ✓ Hazard analysis and risk analysis to identify *acceptable* levels of safety and reliability

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

33

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

Department of computer Engineering  
ati.ttu.ee

## Hazard Analysis

Hazards & Risk Definitions

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Definitions

- ✓ Hazard
  - Situation with actual or potential danger to people, environment or material, of a certain severity
  - e.g. lock that prevents elevator door from opening is not activated
- ✓ Incident (near miss)
  - Unplanned event that involves no damage or loss, but has the potential to be an accident in different circumstances
  - e.g. elevator door opens while the elevator is missing but nobody is leaning against it

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

35

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Definitions (cont.)

- ✓ Accident
  - Unplanned event that results in a certain level of damage or loss to human life or the environment
  - e.g. elevator door opens and someone falls to the shaft
- ✓ Risk
  - Combination of the severity of a specified hazardous event with its probability of occurrence over a specified duration

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

36

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Risk Assessment

- ✓ Risk = penalty x likelihood
  - Penalty can be measured in money, lives, injuries, amount of deadline...
  - Likelihood is the probability that a particular hazard will be activated and result in an undesirable outcome
  - Pareto ranking: 80% of problems are from 20% of the risks...

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

37

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Risk Assessment (cont.)

- ✓ Example of risk calculation
  - Failure of a particular component results in chemical leak that could kill 500 people
  - Estimate that component will fail once every 10,000 years
 
$$\text{risk} = \text{penalty} \times (\text{probability per year})$$

$$= 500 \times (0.0001)$$

$$= 0.05 \text{ deaths per year}$$
- ✓ But rare and costly events are a problem
  - E.g. infinite penalty multiplied by near-zero probability?
  - Must guard against catastrophic penalties event for near-zero probability

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

38

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Acceptability of Risk

- ✓ ALARP (As Low As is Reasonably Possible)
  - If risk can be easily reduced, it should be
  - Conversely, a system with significant risk may be acceptable if it offers sufficient benefit and if further reduction of risk is impractical
- ✓ Ethical considerations
  - Determining risk and its acceptability involves moral judgement
  - Society's view not determined by logical rules
  - Perception that accidents involving large numbers of deaths are perceived as more serious than smaller accidents, though they may occur less frequently

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

39

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Conflicting Requirements – Safety and Reliability

- ✓ A system can be unreliable but safe
  - If it does not behave according to specification but still does not cause an accident
- ✓ A system can be unsafe but reliable
  - If it can cause harm but faults occur with very low probability
- ✓ Fail Safe
  - System designed to fail in a safe state  
e.g. trains stop in case of signal failure
  - affects availability – 100% safe but 0% available..
- ✓ Fail Operational
  - System designed to keep working even if something fails
  - usually using redundancy
- ✓ Fail-over to reduced capability system
  - Mechanical backup

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

40

1918 TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

Department of computer Engineering  
ati.ttu.ee

## Hazards

### Hazards Overview

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Hazards

- ✓ A Hazard is a system state that could lead to:
  - Loss of life
  - Loss of property
  - Release of energy
  - Release of dangerous materials
- ✓ Hazards are the *states* we have to avoid
- ✓ An accident is a loss event:
  - System in hazard state, **and**
  - Change in the operating environment
- ✓ Classification
  - Severity
  - Nature

TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

42

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Hazard Categories for Civil Aircraft

DESCRIPTION	CATEGORY	DEFINITION	PROBABILITY
CATASTROPHIC	I	Loss of Lives, Loss of Aircraft	$10^{-9}/\text{hr}$
HAZARDOUS	II	Severe Injuries, Major aircraft Damage	$10^{-7}/\text{hr}$
MAJOR	III	Minor injury, minor aircraft or system damage	$10^{-5}/\text{hr}$
MINOR	IV	Less than minor injury, less than minor aircraft or system damage	$10^{-3}/\text{hr}$
NO EFFECT	V	No change to operational capability	$10^{-2}/\text{hr}$

© G.F. Marsters

PTIS TALLINNA TEHNIKAÜLIKOOL TALLINN UNIVERSITY OF TECHNOLOGY

43

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Hazard Categories for Civil Aircraft

Frequency of Occurrence	Level	Specific Item	Fleet or Inventory	Failure Probability per Flight Hour
Frequent	A	Likely to occur frequently	Continuously experienced	$\geq 1 \times 10^{-3}$
Reasonably Probable	B	Will occur several times in the life of each item	Will occur frequently	$< 1 \times 10^{-3}$ to $\geq 1 \times 10^{-5}$
Remote	C	Unlikely but possible to occur in the life of an item	Unlikely but can reasonably be expected to occur	$< 1 \times 10^{-5}$ to $\geq 1 \times 10^{-7}$
Extremely Remote	D	So unlikely it can be assumed that the occurrence may not be experienced	Unlikely to occur, but possible	$< 10^{-7}$ to $\geq 1 \times 10^{-9}$
Extremely Improbable	E	Should never happen in the life of all the items in the fleet	Not expected to occur during life of all aircraft of this type	$< 1 \times 10^{-9}$

© G.F. Marsters

Risk from lightning is  $5 \times 10^{-7}$  deaths per person year

PTIS TALLINNA TEHNIKAÜLIKOOL TALLINN UNIVERSITY OF TECHNOLOGY

44

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Hazard Risk Index

Probability	Severity Classification			
	Catastrophic	Hazardous	Major	Minor
Frequent	1	3	7	13
Reasonably Probable	2	5	9	16
Remote	4	6	11	18
Extremely Remote	8	10	14	19
Extremely Improbable	12	15	17	20

■ Acceptable - only ALARP actions considered  
■ Acceptable - use ALARP principle and consider further investigations  
■ Not acceptable - risk reducing measures required

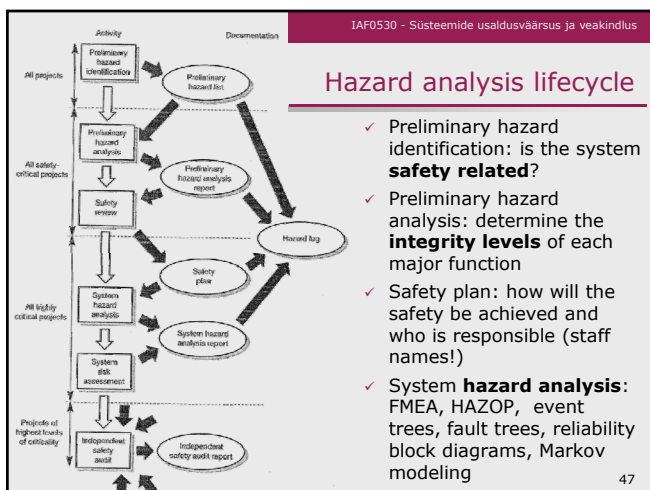
PTIS TALLINNA TEHNIKAÜLIKOOL TALLINN UNIVERSITY OF TECHNOLOGY

45

1918 TALLINNA TEHNIKAÜLIKOOL TALLINN UNIVERSITY OF TECHNOLOGY Department of computer Engineering [ati.ttu.ee](http://ati.ttu.ee)

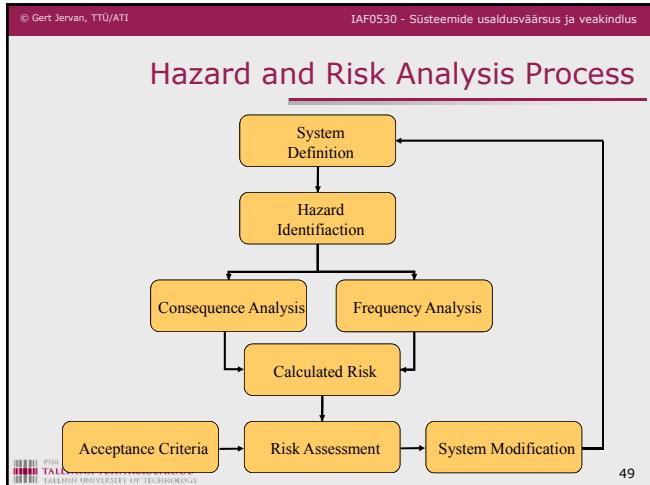
## Hazards

### Hazard Analysis



- © Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus
- ### Hazard Analysis
- ✓ The purpose
    - Identify events that may lead to accidents
    - Determine impact on system
    - Performed throughout the life cycle
  - ✓ Analytical Techniques
    - Failure modes and effects analysis (FMEA)
    - FMECA: Failure modes, effects and criticality analysis (FMECA)
    - ETA: Event tree analysis (ETA)
    - FTA: Fault tree analysis (FTA)
    - HAZOP: Hazard and operability studies (HAZOP)
  - ✓ Standards
- PTIS TALLINNA TEHNIKAÜLIKOOL TALLINN UNIVERSITY OF TECHNOLOGY
- 48





© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Preliminary Hazard Identification

- ✓ First activity in safety process, performed during early requirements analysis (concept definition)
- ✓ Identifies potential hazard sources and accidents
- ✓ Sources of information include
  - system concept and operational environment
  - incident data of previous in-service operation and similar systems
  - technology and domain specific analyses and checklists
- ✓ Method is group-based and dependent on experience
- ✓ Process is largely informal
- ✓ Output is Preliminary Hazard List

50

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Preliminary Hazard Analysis

- ✓ Refines hazards and accidents based on design proposal
- ✓ Performed using a system model that defines
  - scope and boundary of system
  - operating modes
  - system inputs, outputs and functions
  - preliminary internal structure
- ✓ Techniques for Preliminary Hazard Analysis include
  - Hazard and Operability Studies
  - Functional Failure Analysis
- ✓ Output is initial Hazard Log

51

1918 TALLINNA TEHNIKAUÜKÜÜK TALLINN UNIVERSITY OF TECHNOLOGY Department of computer Engineering at.ttu.ee

### Hazard Analysis

Failure Mode and Effects Analysis (FMEA)

Failure Modes, Effects and Criticality Analysis (FMECA)

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Failure Mode and Effects Analysis

- ✓ **Failure modes and effects analysis (FMEA)** considers the failure of any component within a system and tracks the effects of this failure to determine its ultimate consequences.
  - Probably the most commonly used technique
  - Looks for consequences of component failures (forward chaining technique)

53

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### FMEA

- ✓ Manual analysis
  - Identify component, module or system failures
  - Determine consequences
  - Performed bottom-up
- ✓ Outputs
  - Spreadsheet noting each
    - failure mode
    - possible causes
    - consequences
    - possible remedies
  - Usually computer records kept
- ✓ Standardised by IEC (International Electrotechnical Commission)

54

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## FMEA

- ✓ Notes
  - Can be applied at any stage of the design process and at any level within the system
  - Teams of four to eight engineers
- ✓ Limitations:
  - Lot of unnecessary work, it considers all components/failure modes
  - Requires expert knowledge to decide what to analyze
  - Usually do not consider multiple failures

55

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## FMEA Example

Ref No.	Unit	Failure mode	Possible cause	Local effects	System effects	Remedial action
1	Tool guard switch	Open-circuit contacts	(a) faulty component (b) excessive current (c) extreme temperature	Failure to detect tool guard in place	Prevents use of machine – system fails safe	Select switch for high reliability and low probability of dangerous failure Rigid quality control on switch procurement
2		Short-circuit contacts	(a) faulty component (b) excessive current	System incorrectly senses guard to be closed – dangerous failure	Allows machine to be used when guard is absent – dangerous failure	Modify software to detect switch failure and take appropriate action
3		Excessive switch-bounce	(a) ageing effects (b) prolonged high currents	Slight delay in sensing state of guard	Negligible	Ensure hardware design prevents excessive current through switch

56

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Failure Modes, Effects and Criticality Analysis

- ✓ FMECA:
  - Extension to FMEA
  - Takes into account importance of each component
  - Determines probability/frequency of occurrence of failures
- ✓ Problems
  - Measuring reliability of components difficult
  - Models often too simplistic
  - Tool support needed
- ✓ Used as input to fault tree analysis
  - Standardised

57

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Background

- ✓ FMECA was one of the first systematic techniques for failure analysis
- ✓ FMECA was developed by the U.S. Military. The first guideline was Military Procedure MIL-P-1629 "Procedures for performing a failure mode, effects and criticality analysis" dated November 9, 1949
- ✓ FMECA is the most widely used reliability analysis technique in the initial stages of product/system development
- ✓ FMECA is usually performed during the conceptual and initial design phases of the system in order to assure that all potential failure modes have been considered and the proper provisions have been made to eliminate these failures

58

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## What can FMECA be used for?

- ✓ Assist in selecting design alternatives with high reliability and high safety potential during the early design phases
- ✓ Ensure that all conceivable failure modes and their effects on operational success of the system have been considered
- ✓ List potential failures and identify the severity of their effects
- ✓ Develop early criteria for test planning and requirements for test equipment
- ✓ Provide historical documentation for future reference to aid in analysis of field failures and consideration of design changes
- ✓ Provide a basis for maintenance planning
- ✓ Provide a basis for quantitative reliability and availability analyses

59

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Types of FMECA

- ✓ **Design FMECA** is carried out to eliminate failures during equipment design, taking into account all types of failures during the whole life-span of the equipment
- ✓ **Process FMECA** is focused on problems stemming from how the equipment is manufactured, maintained or operated
- ✓ **System FMECA** looks for potential problems and bottlenecks in larger processes, such as entire production lines

60

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## FME(C)A Chart

Failure Modes and Effect Analysis								
Product Name: DeWalt Tradesman Drill				Part name: Rear Vent				
Function	Failure Mode	Effects of Failure	Causes of Failure	Current Controls	S	O	D	RPN
Allow Additional Air Flow	Filter Blocked	Overheated Motor	User Error	Visual Inspection	4	1	5	20
Prevent Dangerous Usage	Filter Not In Place	Larger Opening to Motor	User Error	Visual Inspection	8	4	1	32
Filter dust	Defective Filter	Additional dust flows into shell	Poor Materials	Visual Inspection	1	1	7	7

S = Severity rating (1 to 10)  
 O = Occurrence frequency (1 to 10)  
 D = Detection Rating (1 to 10)  
 RPN = Risk Priority Number (1 to 1000)

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

61

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Severity Rating

Rank	Severity class	Description
10	Catastrophic	Failure results in major injury or death of personnel.
7-9	Critical	Failure results in minor injury to personnel, personnel exposure to harmful chemicals or radiation, or fire or a release of chemical to the environment.
4-6	Major	Failure results in a low level of exposure to personnel, or activates facility alarm system.
1-3	Minor	Failure results in minor system damage but does not cause injury to personnel, allow any kind of exposure to operational or service personnel or allow any release of chemicals into the environment

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

62

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Detection Rating

Rank	Description
1-2	Very high probability that the defect will be detected. Verification and/or controls will almost certainly detect the existence of a deficiency or defect.
3-4	High probability that the defect will be detected. Verification and/or controls have a good chance of detecting the existence of a deficiency/defect.
5-7	Moderate probability that the defect will be detected. Verification and/or controls are likely to detect the existence of a deficiency or defect.
8-9	Low probability that the defect will be detected. Verification and/or control not likely to detect the existence of a deficiency or defect.
10	Very low (or zero) probability that the defect will be detected. Verification and/or controls will not or cannot detect the existence of a deficiency/defect.

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

63

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Risk Ranking

- ✓ Risk Matrix
- ✓ Risk Ranking:
  - O = the rank of the occurrence of the failure mode
  - S = the rank of the severity of the failure mode
  - D = the rank of the likelihood the the failure will be detected before the system reaches the end-user/customer.
  - All ranks are given on a scale from 1 to 10. The risk priority number (RPN) is defined as
$$RPN = S \times O \times D$$
  - The smaller the RPN the better – and – the larger the worse.

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

64

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

Department of computer Engineering  
ati.ttu.ee

## Hazard Analysis

### Hazard & Operability Analysis (HAZOP)

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Hazard & Operability Analysis

- ✓ HAZOP:
  - Developed in Chemical industry
  - Applied successfully in other domains
  - "What if" analysis for system parameters
  - E.g., suppose "temperature" of "reactor" "rises", what happens to system?
  - System realization of perturbation or sensitivity analysis
  - Requires flow model of operating plant

PTIS  
TALLINNA TEHNIKAÜLIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

66

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Hazard & Operability Analysis

- ✓ Flowing items are "entities"
- ✓ Entities have characteristic properties known as "attributes"
- ✓ Analysis based on possible deviations of attribute values
- ✓ "Guide words" used to guide the analysis— designed to capture dimensions of variation
- ✓ Supplementary adjectives add temporal element
- ✓ Different word sets for different applications

TALLINNA TEHNIKALIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

67

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## HAZOP examples

- ✓ Guide words:
  - no, more, less, early, late, before, ...

Interpretation examples:

- Signal arrives too late
- Incomplete data transmitted / only part of the intended activity occurs

- ✓ Attributes:
  - Data flow, data rate, response time, ...

TALLINNA TEHNIKALIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

68

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## HAZOP guide word interpretations

Guide word	Chemical plant	Computer-based system
No	No part of the intended result is achieved	No data or wanted signal exchanged
More	A quantitative increase in the physical quantity	A signal magnitude or a data rate is too high
Less	A quantitative decrease in the physical quantity	A signal magnitude or a data rate is too low
As well as	The intended activity occurs, but with additional results	Redundant data sent in addition to intended value
Part of	Only part of the intended activity occurs	Incomplete data transmitted
Reverse	The opposite of what was intended occurs, for example reverse flow within a pipe	Polarity of magnitude changes reversed
Other than	No part of the intended activity occurs, and something else happens instead	Data complete but incorrect
Early	Not used	Signal arrives too early with reference to clock time
Late	Not used	Signal arrives too late with reference to clock time
Before	Not used	Signal arrives earlier than intended within a sequence
After	Not used	Signal arrives later than intended within a sequence

TALLINNA TEHNIKALIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

69

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## HAZOP attributes

Attribute	Guide word	Possible meaning
Data flow	More	More data is passed than expected
	Less	Less data is passed than expected
Data rate	More	The data rate is too high
	Less	The data rate is too low
Data value	More	The data value is too high
	Less	The data value is too low
Repetition time	More	The time between output updates is too high
	Less	The time between output updates is too low
Response time	More	The response time is longer than required
	Less	The response time is shorter than required

TALLINNA TEHNIKALIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

70

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## HAZOP Example

Item	Inter-connection	Attribute	Guide word	Cause	Consequence	Recommendation
1	Sensor supply line	Supply voltage	No	PSU, regulator or cable fault	Lack of sensor signal detected and system shuts down	
2			More	Regulator fault	Possible damage to sensor	Consider overvoltage protection
3			Less	PSU or regulator fault	Incorrect temperature reading	Include voltage monitoring
4	Sensor current	Sensor current	More	Sensor fault	Incorrect temperature reading, possible loading of supply	Monitor supply current
5			Less	Sensor fault	Incorrect temperature reading	As above
6						
7	Sensor output	Voltage	No	PSU, sensor or cable fault	Lack of sensor signal detected and system shuts down	
8			More	Sensor fault	Temperature reading too high - results in decrease in plant efficiency	Consider use of duplicate sensor
			Less	Sensor mounted incorrectly or sensor failure	Temperature reading too low - could result in overloading and possible plant failure	As above

TALLINNA TEHNIKALIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

71

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

TALLINNA TEHNIKALIKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

Department of computer Engineering  
ati.ttu.ee

## Hazard Analysis

### Fault Tree Analysis (FTA)

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Fault Tree Analysis

- ✓ Fault tree analysis (FTA) is a top-down approach to failure analysis, starting with a potential undesirable event (accident) called a TOP event, and then determining all the ways it can happen.
- ✓ The analysis proceeds by determining how the TOP event can be caused by individual or combined lower level failures or events.
- ✓ The causes of the TOP event are "connected" through logic gates
- ✓ FTA is the most commonly used technique for causal analysis in risk and reliability studies.

TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

73

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## History

- ✓ FTA was first used by Bell Telephone Laboratories in connection with the safety analysis of the Minuteman missile launch control system in 1962
- ✓ Technique improved by Boeing Company
- ✓ Extensively used and extended during the Reactor safety study (WASH 1400)

TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

74

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Preparations for FTA

TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

75

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Boundary Conditions

- ✓ The physical boundaries of the system (Which parts of the system are included in the analysis, and which parts are not?)
- ✓ The initial conditions (What is the operational stat of the system when the TOP event is occurring?)
- ✓ Boundary conditions with respect to external stresses (What type of external stresses should be included in the analysis – war, sabotage, earthquake, lightning, etc?)
- ✓ The level of resolution (How detailed should the analysis be?)

TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

76

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

## Fault Tree Construction

- ✓ Define the TOP event in a clear and unambiguous way.  
Should always answer:  
What e.g., "Fire"  
Where e.g., "in the process oxidation reactor"  
When e.g., "during normal operation"
- ✓ What are the immediate, necessary, and sufficient events and conditions causing the TOP event?
- ✓ Connect via a logic gate
- ✓ Proceed in this way to an appropriate level (= basic events)
- ✓ Appropriate level:
  - Independent basic events
  - Events for which we have failure data

TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

77

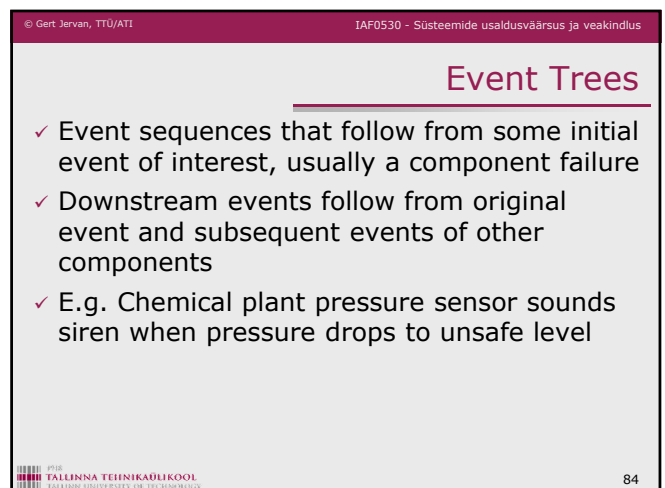
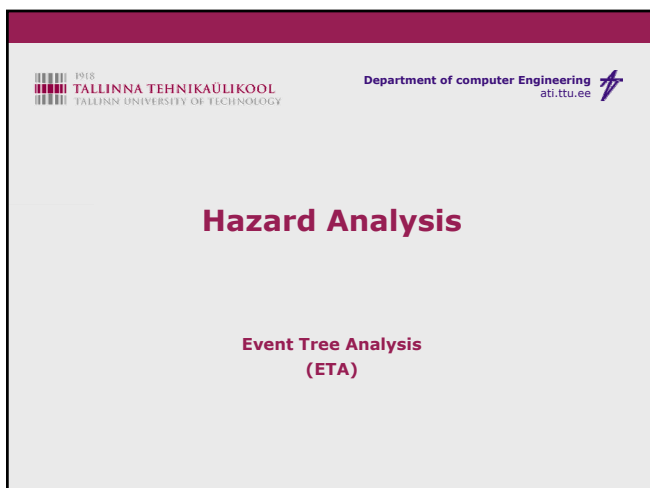
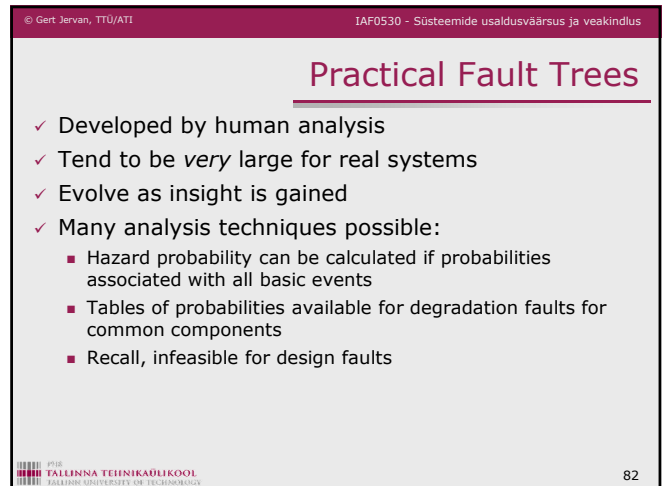
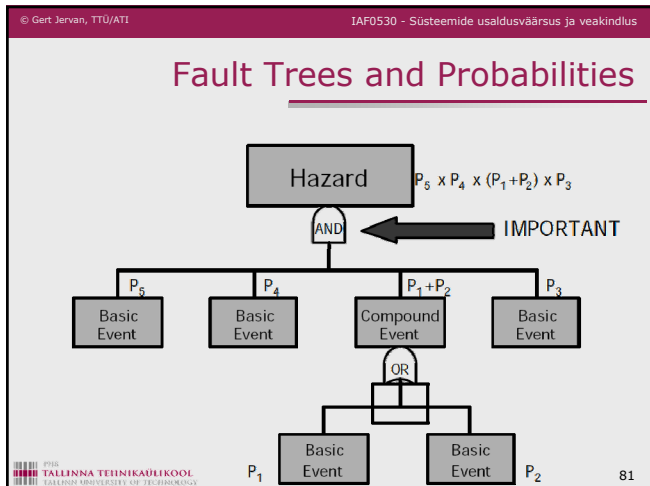
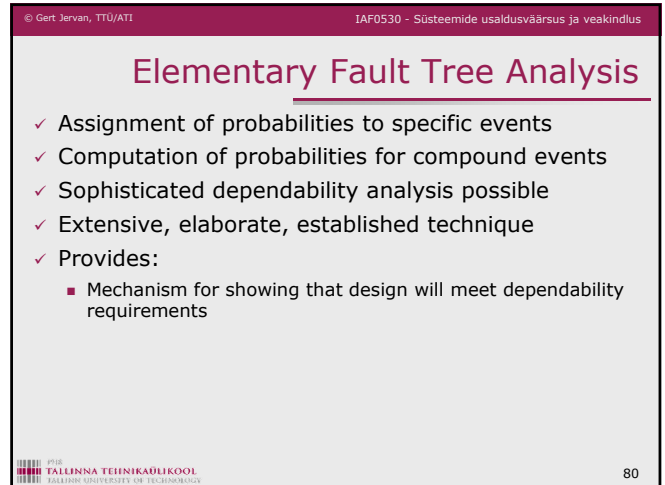
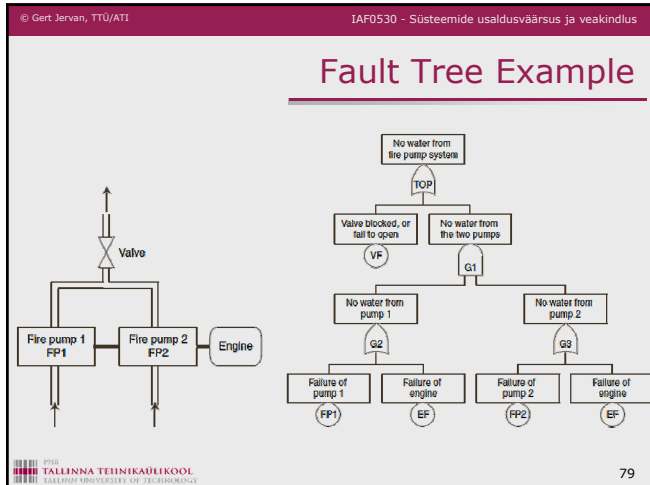
© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

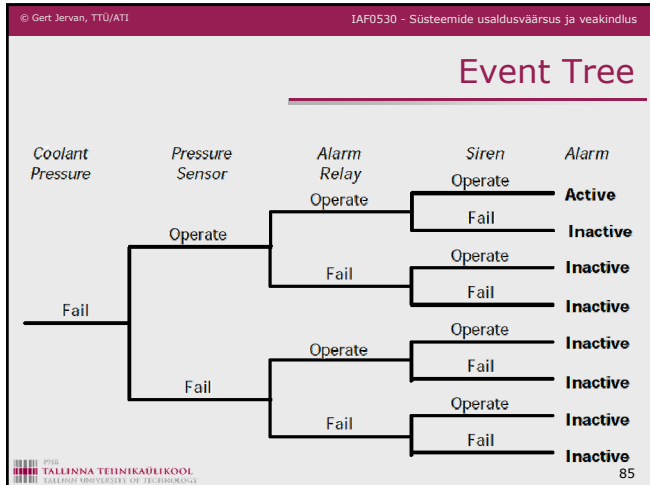
## Fault Tree Symbols

Logic gates	 OR-gate  AND-gate	<p>The OR-gate indicates that the output event occurs if any of the input events occur</p> <p>The AND-gate indicates that the output event occurs only if all the input events occur at the same time</p>
Input events (states)	  	<p>The basic event represents a basic equipment failure that requires no further development of failure causes</p> <p>The undeveloped event represents an event that is not examined further because information is unavailable or because its consequences are insignificant</p> <p>The comment rectangle is for supplementary information</p>
Transfer symbols	 Transfer out  Transfer in	<p>The transfer-out symbol indicates that the fault tree is developed further at the occurrence of the corresponding transfer-in symbol</p>

TALLINNA TEHNIKAKÜLKOOL  
TALLINN UNIVERSITY OF TECHNOLOGY

78





© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Barriers

- ✓ Most well designed systems have one or more barriers that are implemented to stop or reduce the consequences of potential accidental events. The probability that an accidental event will lead to unwanted consequences will therefore depend on whether these barriers are functioning or not.
- ✓ The consequences may also depend on additional events and factors. Examples include:
  - Whether a gas release is ignited or not
  - Whether or not there are people present when the accidental event occurs
  - The wind direction when the accidental event occurs
- ✓ Barriers may be technical and/or administrative (organizational).

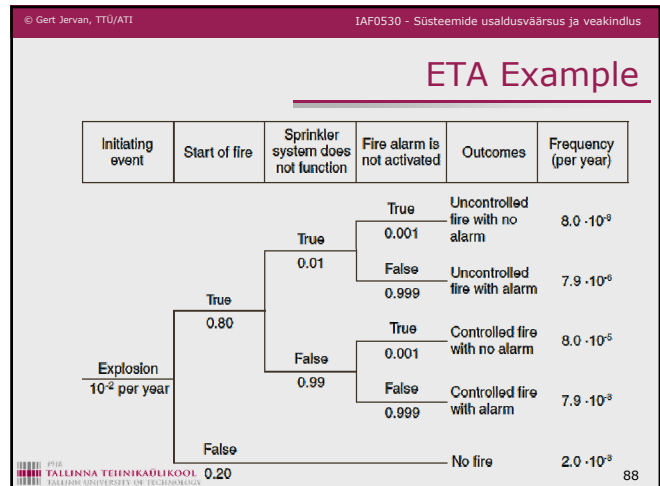
PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY 86

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Event Tree Analysis

- ✓ An event tree analysis (ETA) is an inductive procedure that shows all possible outcomes resulting from an accidental (initiating) event, taking into account whether installed safety barriers are functioning or not, and additional events and factors.
- ✓ By studying all relevant accidental events (that have been identified by a preliminary hazard analysis, a HAZOP, or some other technique), the ETA can be used to identify all potential accident scenarios and sequences in a complex system.
- ✓ Design and procedural weaknesses can be identified, and probabilities of the various outcomes from an accidental event can be determined.

PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY 87



© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### ETA Pros and Cons

- ✓ Positive
  - Visualize event chains following an accidental event
  - Visualize barriers and sequence of activation
  - Good basis for evaluating the need for new / improved procedures and safety functions
- ✓ Negative
  - No standard for the graphical representation of the event tree
  - Only one initiating event can be studied in each analysis
  - Easy to overlook subtle system dependencies
  - Not well suited for handling common cause failures in the quantitative analyses
  - The event tree does not show acts of omission

PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY 89

© Gert Jervan, TTÜ/ATI IAF0530 - Süsteemide usaldusväärsus ja veakindlus

### Hazard Analysis in the Life Cycle

- ✓ FME(C)A
  - Used to generate event trees and fault trees
- ✓ FME(C)A, FTA, ETA
  - Appropriate when functional design complete
- ✓ Preliminary HAZOP
  - Early in the life-cycle
  - Identify hazards, take account of them in the design
- ✓ Full HAZOP
  - Later in the life-cycle
  - Identify further hazards, feed back into design design

PTIS TALLINNA TEHNIKAKÜLKOOL TALLINN UNIVERSITY OF TECHNOLOGY 90