Dependable Systems

Idaho State University | Computer Science

Isaac Griffith

CS 3321
Department of Computer Science
Idaho State University

ROAR

Topics Covered

- Dependability properties
- Sociotechnical systems
- Redundancy and diversity

System dependability

- For many computer-based systems, the most important system property is the dependability of the system.
- The dependability of a system reflects the user's degree of trust in that system. It reflects the extent of the user's confidence that it will operate as users expect and that it will not 'fail' in normal use.
- Dependability covers the related systems attributes of reliability, availability and security. These are all inter-dependent.

Importance of dependability

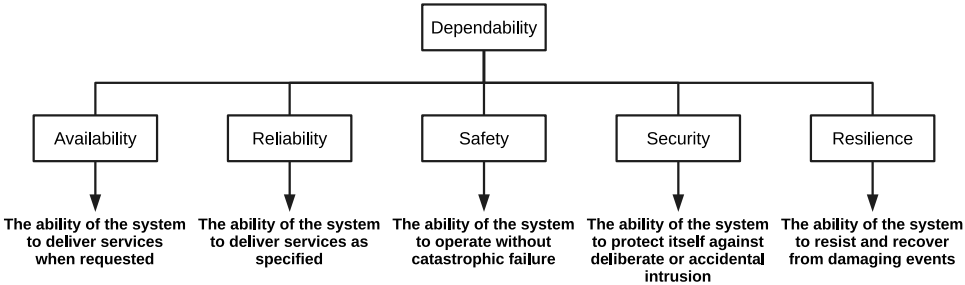
- System failures may have widespread effects with large numbers of people affected by the failure.
- Systems that are not dependable and are unreliable, unsafe or insecure may be rejected by their users.
- The costs of system failure may be very high if the failure leads to economic losses or physical damage.
- Undependable systems may cause information loss with a high consequent recovery cost.

Causes of failure

- Hardware failure
  - Hardware fails because of design and manufacturing errors or because components have reached the end of their natural life.
- Software failure
  - Software fails due to errors in its specification, design or implementation.
- Operational failure
  - Human operators make mistakes. Now perhaps the largest single cause of system failures in socio-technical systems.

Dependability properties

The principal dependability properties



Principal properties

- Availability
  - The probability that the system will be up and running and able to deliver useful services to users.
- Reliability
  - The probability that the system will correctly deliver services as expected by users.
- Safety
  - A judgment of how likely it is that the system will cause damage to people or its environment.

Principal properties

- Security
  - A judgment of how likely it is that the system can resist accidental or deliberate intrusions.
- Resilience
  - A judgment of how well a system can maintain the continuity of its critical services in the presence of disruptive events such as equipment failure and cyberattacks.

Other dependability properties

- Repairability
  - Reflects the extent to which the system can be repaired in the event of a failure
- Maintainability
  - Reflects the extent to which the system can be adapted to new requirements;
- Error tolerance
  - Reflects the extent to which user input errors can be avoided and tolerated.

Dependability attribute dependencies

- Safe system operation depends on the system being available and operating reliably.
- A system may be unreliable because its data has been corrupted by an external attack.
- Denial of service attacks on a system are intended to make it unavailable.
- If a system is infected with a virus, you cannot be confident in its reliability or safety.

Dependability achievement

- Avoid the introduction of accidental errors when developing the system.
- Design V & V processes that are effective in discovering residual errors in the system.
- Design systems to be fault tolerant so that they can continue in operation when faults occur
- Design protection mechanisms that guard against external attacks.

Dependability achievement

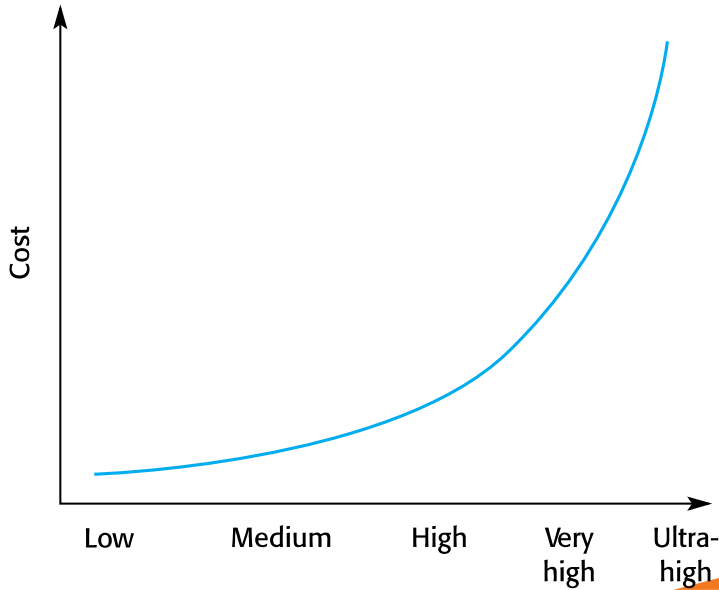
- Configure the system correctly for its operating environment.
- Include system capabilities to recognize and resist cyberattacks.
- Include recovery mechanisms to help restore normal system service after a failure.

Dependability costs

- Dependability costs tend to increase exponentially as increasing levels of dependability are required.
- There are two reasons for this
  - The use of more expensive development techniques and hardware that are required to achieve the higher levels of dependability.
  - The increased testing and system validation that is required to convince the system client and regulators that the required levels of dependability have been achieved.

Idaho State University
Computer Science

Cost/dependability curve



ROAR

Dependability economics

- Because of very high costs of dependability achievement, it may be more cost effective to accept untrustworthy systems and pay for failure costs
- However, this depends on social and political factors. A reputation for products that can't be trusted may lose future business
- Depends on system type - for business systems in particular, modest levels of dependability may be adequate

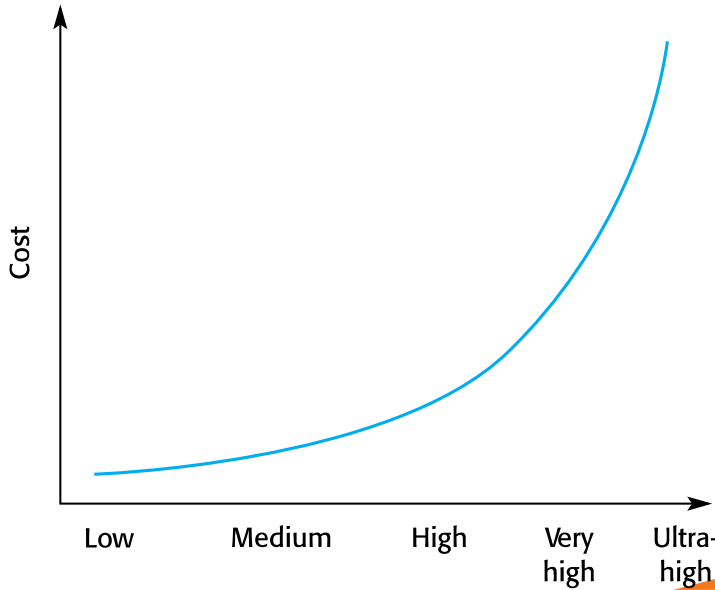
Sociotechnical systems

Systems and software

- Software engineering is not an isolated activity but is part of a broader systems engineering process.
- Software systems are therefore not isolated systems but are essential components of broader systems that have a human, social or organizational purpose.
- Example
  - The wilderness weather system is part of broader weather recording and forecasting systems
  - These include hardware and software, forecasting processes, system users, the organizations that depend on weather forecasts, etc.

Idaho State University
Computer Science

The sociotechnical systems stack



ROAR

Layers in the STS stack

- Equipment
  - Hardware devices, some of which may be computers. Most devices will include an embedded system of some kind.
- Operating system
  - Provides a set of common facilities for higher levels in the system.
- Communications and data management
  - Middleware that provides access to remote systems and databases.
- Application systems
  - Specific functionality to meet some organization requirements.

Layers in the STS stack

- Business processes
  - A set of processes involving people and computer systems that support the activities of the business.
- Organizations
  - Higher level strategic business activities that affect the operation of the system.
- Society
  - Laws, regulation and culture that affect the operation of the system.

Holistic system design

- There are interactions and dependencies between the layers in a system and changes at one level ripple through the other levels
  - Example: Change in regulations (society) leads to changes in business processes and application software.
- For dependability, a systems perspective is essential
  - Contain software failures within the enclosing layers of the STS stack.
  - Understand how faults and failures in adjacent layers may affect the software in a system.

Regulation and compliance

- The general model of economic organization that is now almost universal in the world is that privately owned companies offer goods and services and make a profit on these.
- To ensure the safety of their citizens, most governments regulate (limit the freedom of) privately owned companies so that they must follow certain standards to ensure that their products are safe and secure.

Regulated systems

- Many critical systems are regulated systems, which means that their use must be approved by an external regulator before the systems go into service.
  - Nuclear systems
  - Air traffic control systems
  - Medical devices
- A safety and dependability case has to be approved by the regulator. Therefore, critical systems development has to create the evidence to convince a regulator that the system is dependable, safe and secure.

Safety regulation

- Regulation and compliance (following the rules) applies to the sociotechnical system as a whole and not simply the software element of that system.
- Safety-related systems may have to be certified as safe by the regulator.
- To achieve certification, companies that are developing safety-critical systems have to produce an extensive safety case that shows that rules and regulations have been followed.
- It can be as expensive develop the documentation for certification as it is to develop the system itself.

Redundancy and diversity

Redundancy and diversity

- Redundancy
  - Keep more than a single version of critical components so that if one fails then a backup is available.
- Diversity
  - Provide the same functionality in different ways in different components so that they will not fail in the same way.
- Redundant and diverse components should be independent so that they will not suffer from ‘common-mode’ failures
  - For example, components implemented in different programming languages means that a compiler fault will not affect all of them.

Diversity and redundancy examples

- **Redundancy.** Where availability is critical (e.g. in e-commerce systems), companies normally keep backup servers and switch to these automatically if failure occurs.
- **Diversity.** To provide resilience against external attacks, different servers may be implemented using different operating systems (e.g. Windows and Linux)

Process diversity and redundancy

- Process activities, such as validation, should not depend on a single approach, such as testing, to validate the system.
- Redundant and diverse process activities are important especially for verification and validation.
- Multiple, different process activities the complement each other and allow for cross-checking help to avoid process errors, which may lead to errors in the software.

Problems with redundancy and diversity

- Adding diversity and redundancy to a system increases the system complexity.
- This can increase the chances of error because of unanticipated interactions and dependencies between the redundant system components.
- Some engineers therefore advocate simplicity and extensive V & V as a more effective route to software dependability.
- Airbus FCS architecture is redundant/diverse; Boeing 777 FCS architecture has no software diversity

Key points

- System dependability is important because failure of critical systems can lead to economic losses, information loss, physical damage or threats to human life.
- The dependability of a computer system is a system property that reflects the user's degree of trust in the system. The most important dimensions of dependability are availability, reliability, safety, security and resilience.
- Sociotechnical systems include computer hardware, software and people, and are situated within an organization. They are designed to support organizational or business goals and objectives.

Key points

- The use of a dependable, repeatable process is essential if faults in a system are to be minimized. The process should include verification and validation activities at all stages, from requirements definition through to system implementation.
- The use of redundancy and diversity in hardware, software processes and software systems is essential to the development of dependable systems.

Are there any questions?