

1- Are the problems of Denver Airport common?

Yes, these problems are not uncommon. Many large-scale software systems face similar issues such as delays, budget overruns, and operational failures. For example, two out of every six large software projects are canceled.

2 - What are the percentages of 'operating failures' Software as observed in the mid-1990s? What are the average increase in scheduling time?

In the mid-1990s, approximately 75% of large software systems were classified as "operational failures," meaning they either did not perform as intended or were not used. On average, project schedules were exceeded by 50%, with large-scale projects often faring worse.

3 - What are the Software Crises?

The Software Crisis refers to pervasive problems in software development, such as project delays, budget overruns, operational failures, and the unpredictability and difficulty in managing the development process.

4 - What is 'Software Engineering'? (answer from the paper)

Software Engineering is defined as "the systematic, disciplined, and quantifiable approach to the development, operation, and maintenance of software".

5 - Do you feel that outsourcing affects software standards and interoperability? Why or why not?

Outsourcing can affect software standards and interoperability because different teams may use varying methodologies and standards, increasing the complexity of integration. Additionally, the lack of unified management can result in quality issues.

6 - Is it easy to get the software right the first time? Maybe if we try harder and with military discipline?

It is extremely difficult to develop perfect software on the first attempt, even with increased effort and military-like discipline. The paper highlights that errors in real-time and complex distributed systems often remain undetected under certain conditions.

7 - What is unique about 'Realtime Systems'? And what is different in 'Distributed Systems'?

Realtime Systems are unique because they must respond to unpredictable events in real-time, with errors often only surfacing under specific conditions. Distributed Systems, on the other hand, involve multiple interconnected computers where the

failure of one node can disrupt the entire system.

8 - Did the Loral team experience go completely successful?

The Loral team achieved some success in the space shuttle project but faced challenges. For instance, the first shuttle launch was delayed by two days due to synchronization issues.

9 - Would formal methods or mathematics solve the problem?

Formal methods and mathematics can reduce errors during the design phase, but they cannot fully ensure the software performs correctly in real-world environments. These methods verify compliance with specifications but cannot address unforeseen situations.

10 - Why are we interested in component software? (from the paper)

Component software improves development efficiency and maintainability by reducing duplication of effort. It is considered a key step in the industrialization of software, akin to interchangeable parts in hardware manufacturing

.