Log In Join

Back To Module Home

Distributed Systems

0% completed

Search Module

Introduction to Distributed Systems

□

Basic Concepts and Theorems

□

- ○ Partitioning
- ○ Algorithms for Horizontal Partitioning
- ○ Replication
- ○ Single-Master Replication Algorithm
- ○ Multi-Master Replication Algorithm
- ○ Quorums in Distributed Systems

Conclusion

□

Mark Module as Completed

Safety Guarantees in Distributed Systems

In this lesson, we will explore the properties that guarantee safety in distributed systems, and their relation with difficulties in designing distributed systems.

We'll cover the following □

Since distributed systems involve a lot of complexity, some safety guarantees ensure that the system will behave in specific, predictable ways. This makes it easier for people to reason about a system and any potential anomalies that can occur. This will allow them to build proper safeguards to prevent these anomalies from occurring.

Safety guarantors#

The main safety guarantees that systems provide are around the three properties shown in the illustration.

Three properties which guarantee safety in a distributed system

The concepts of **atomicity** and **isolation** originate from database research and ACID transactions. When we mention **consistency** in this course, we will mostly refer to the notion of consistency made popular by the CAP theorem.

Before going any further, it is useful to look at these topics. We will study these two topics in detail in the next two lessons.

It is interesting to observe that each of these safety guarantees is tightly related to the aforementioned reasons that make distributed systems hard to design.

Achieving atomicity#

It is challenging to achieve atomicity in a distributed system because of the possibility of **partial failures**.

A partial failure occurs when some components in the system fail. The following illustration shows this.

Partial failures

Achieving consistency#

It is challenging to achieve consistency because of the **network asynchrony**.

Network asynchrony occurs when different nodes in a network have different values for the current time. The following illustration shows this.

Network asynchrony

Achieving isolation#

It is challenging to achieve isolation because of the inherent concurrency of distributed systems.

Concurrency occurs when multiple things happen at the same time. The following illustration shows this.

Concurrency

In the above illustration, two pens are trying to write on a single resource at the same time.

Quorums in Distributed Systems

Next

ACID Transactions

Mark as Completed

Report an Issue