

# CASE STUDY NO : 04

Date	1 / 20
Page No.	22

Case Study On :- Study of clock synchronization in distributed OS.

## Synchronization in Distributed System -

Distributed System is a collection of computers connected via the high speed communication networks. In the distributed system, the hardware and software components communicate and coordinate their actions by message passing. Each node in distributed systems can share their resources with other nodes. So, the resources with other nodes. there is need of proper allocation of resources to preserve the state of resources and help coordinate between the several processes. To resolve such conflicts, synchronization is used. Synchronization in distributed systems is achieved via clock.

The physical clocks are used to adjust the time of nodes. Each node in the system can share its local time with other nodes in the system. The time is set based on UTC (Universal Time Coordination). UTC is used as a reference time clock for the nodes in the system.

The clock synchronization can be achieved by 2 ways: External and Internal Clock Synchronization.

## 1. External clock synchronization -

External clock synchronization is the one in which an external reference clock is present. It is used as a reference and the nodes in the system can set and adjust their time accordingly.

## 2. Internal clock synchronization -

Internal clock synchronization is the one in which each node shares its time with other nodes and all the nodes set and adjust their times accordingly.

- \* There are 2 types of clock synchronization algorithms: Centralized and Distributed.

### 1. Centralized -

Centralized is the one in which a time server is used as a reference. The single time server propagates its time to the nodes and all the nodes adjust the time accordingly. It is dependent on single time server so if that node fails, the whole system will lose synchronization. Examples of centralized are - Berkeley Algorithm, Passive Time Server, Active Time Server etc.

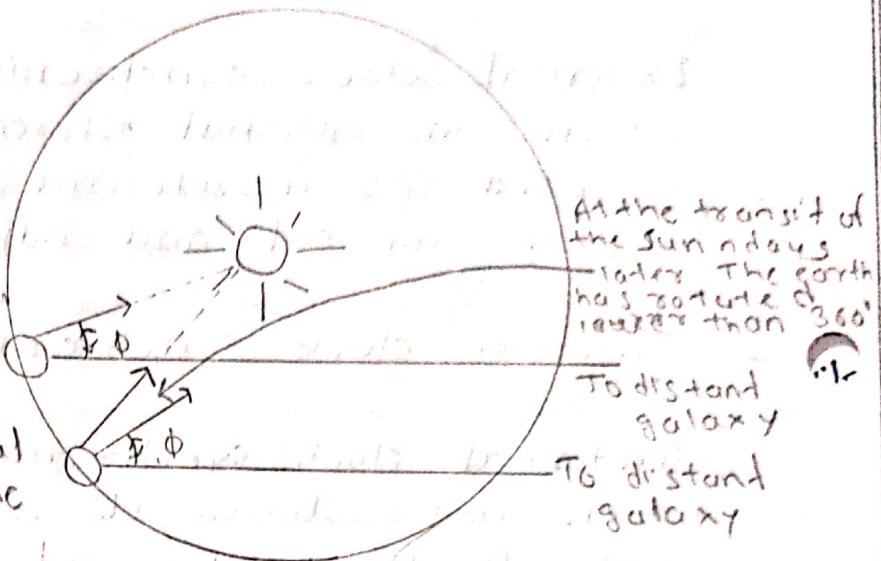
and still maintaining orbit around the sun.

A transit of the sun occurs when the sun reaches the highest point of the day (meridian) in relation to the earth's orbit and the orbital period is 365 days.

For on day 0 of the transit of the sun the sun is at the meridian and the earth has rotated 180°. At the transit of the sun 1 day later the sun has rotated more than 360°.

Earth on day 0 of the transit of the sun

Earth on day 1 of the transit of the sun



Earth's rotation is also responsible for the apparent motion of the stars across the sky. A celestial object's position in the sky is defined by its Right Ascension (RA) and Declination (Dec). RA is measured along the ecliptic, and Dec is measured relative to the celestial equator. As Earth rotates, the stars appear to move across the sky from east to west. This apparent motion is called stellar parallax. The angle of parallax is the angle between the direction of the star and the direction of the Sun. The angle of parallax is proportional to the distance of the star from Earth.

Apparent magnitude is the brightness of a star as seen from Earth. It is a logarithmic scale where brighter stars have lower magnitudes. A star with a magnitude of 1 is 100 times brighter than a star with a magnitude of 6.

Astronomers often use a logarithmic scale to measure the brightness of stars. The apparent magnitude of a star is a logarithmic measure of its brightness. A star with a magnitude of 1 is 100 times brighter than a star with a magnitude of 6.

Astronomers often use a logarithmic scale to measure the brightness of stars. The apparent magnitude of a star is a logarithmic measure of its brightness. A star with a magnitude of 1 is 100 times brighter than a star with a magnitude of 6.

Astronomers often use a logarithmic scale to measure the brightness of stars. The apparent magnitude of a star is a logarithmic measure of its brightness. A star with a magnitude of 1 is 100 times brighter than a star with a magnitude of 6.

Astronomers often use a logarithmic scale to measure the brightness of stars. The apparent magnitude of a star is a logarithmic measure of its brightness. A star with a magnitude of 1 is 100 times brighter than a star with a magnitude of 6.

Astronomers often use a logarithmic scale to measure the brightness of stars. The apparent magnitude of a star is a logarithmic measure of its brightness. A star with a magnitude of 1 is 100 times brighter than a star with a magnitude of 6.

## 2. Distributed -

Distributed is the one in which there is no centralized time server present. Instead the nodes adjust their time by using their local time and then, taking the average of the differences of time with other nodes. Distributed algorithms overcome the issue of centralized algorithms like the scalability and single point failure. Examples of Distributed algorithms are - Global Averaging Algorithm, Localized Averaging Algorithm, NTP (Network time protocol) etc.

### \* Physical Clocks -

Although Lamport's algorithm gives an unambiguous event ordering, the time values assigned to events are not necessarily close to the actual times at which they occur. In some systems (eg. real time systems), the actual clock time is important. For these systems external physical clocks are required. For reasons of efficiency and redundancy, multiple physical clocks are generally considered desirable, which with real-world clocks, and (2) How do we synchronize the clocks with each other.

### \* Clock Synchronization Algorithms -

All the algorithms have the same underlying model of the system, which we will now describe. Each machine is assumed to have a timer that causes

$$\frac{dc}{dt} > 1$$

clocks will tick faster than the standard clock.

Fast clock  $\rightarrow \frac{dc}{dt} < 1$ , slow clock  $\rightarrow \frac{dc}{dt} > 1$ .

Relative time with respect to the standard clock.

Not all clocks tick precisely at correct rate.

Not all clocks tick precisely at correct rate.

Date	/ /20
Page No.	25

an interrupt 11 times a second. When this timer goes off, the interrupt handler adds 1 to a software clock that keeps track of the number of ticks (interrupts) since some agreed-upon time in the past. Let us call the value of this clock  $C$ . More specifically, when the UTC time is  $t$ , the value of the clock on machine  $p$  is  $C_p(t)$ . In a perfect world, we would have  $C_p(t) = t$  for all  $p$  and all  $t$ . In other words  $dC/dt$  ideally should be 1.

#### \* Use of Synchronized Clocks -

Only quite recently has the necessary hardware and software for synchronizing clocks on a wide scale (e.g., over the entire Internet) become easily available. With this new technology, it is possible to keep millions of clocks synchronized to within a few milliseconds of UTC. New algorithms that utilize synchronized clocks are just starting to appear. Below we summarize two of the examples discussed by Liskov (1993).

