



Time Sharing Operating System

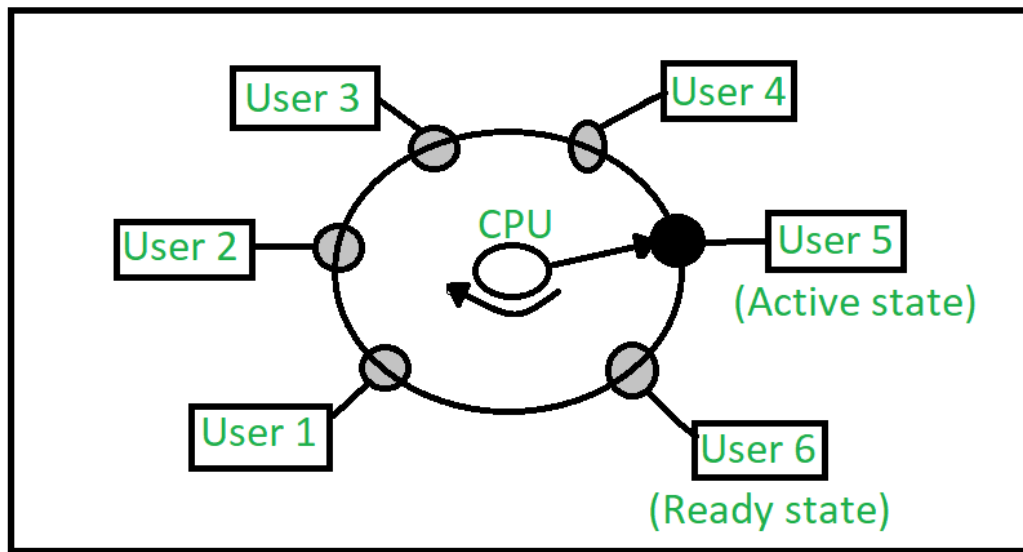


Last Updated : 23 Jun, 2023

Multiprogrammed, batched systems provide an environment where various system resources were used effectively, but it did not provide for user interaction with computer systems. Time-sharing is a logical extension of [multiprogramming](#). The CPU performs many tasks by switches that are so frequent that the user can interact with each program while it is running. A time-shared operating system allows multiple users to share computers simultaneously. With each action or order at a time the shared system becomes smaller, so only a little CPU time is required for each user. As the system rapidly switches from one user to another, each user is given the impression that the entire computer system is dedicated to its use, although it is being shared among multiple users.

A time-shared operating system uses CPU scheduling and [multi-programming](#) to provide each user with a small portion of a shared computer at once. Each user has at least one separate program in memory. A program is loaded into memory and executes, it performs a short period of time either before completion or to complete I/O. This short period of time during which the user gets the attention of the [CPU](#) is known as **time slice, time slot, or quantum**. It is typically of the order of 10 to 100 milliseconds. Time-shared operating systems are more complex than multiprogrammed operating systems. In both, multiple jobs must be kept in memory simultaneously, so the system must have memory management and security. To achieve a good response time, jobs may have to swap in and out of disk from the [main memory](#) which now serves as a backing store for the main memory. A common method

to achieve this goal is virtual memory, a technique that allows the execution of a job that may not be completely in memory.



In the above figure the user 5 is **active state** but user 1, user 2, user 3, and user 4 are in a **waiting state** whereas user 6 is in a **ready state**.

1. **Active State** – The user's program is under the control of the CPU. Only one program is available in this state.
2. **Ready State** – The user program is ready to execute but it is waiting for its turn to get the CPU. More than one user can be in a ready state at a time.
3. **Waiting State** – The user's program is waiting for some input/output operation. More than one user can be in a waiting state at a time.

Requirements of Time Sharing Operating System: An alarm clock mechanism to send an interrupt signal to the CPU after every time slice. Memory Protection mechanism to prevent one job's instructions and data from interfering with other jobs.

What are some of the key features of a Time-Sharing Operating System?

A Time-Sharing Operating System's key characteristics include the capacity to support multiple concurrent users and the capacity to reduce response times for all users. Additionally, because they permit multiple users to use the system without needing to purchase individual licenses, [time-sharing](#) operating systems may be more cost-effective for businesses.

What are some benefits of the Time-sharing operating System?

The ability for multiple users to use the system at various terminals simultaneously is one advantage of using a time-sharing operating system. All users' response times can be cut down, and the system's resources can be used more effectively.

Additionally, because they permit multiple users to use the system without needing to purchase individual licenses, time-sharing operating systems may be more cost-effective for businesses.

Advantages

1. Each task gets an equal opportunity.
2. Fewer chances of duplication of software.
3. CPU idle time can be reduced.

Disadvantages

1. Reliability problem.
2. One must have to take of the security and integrity of user programs and data.
3. Data communication problem.

Conclusion

A time-sharing operating system enables concurrent use of the computer by multiple users. By distributing the computer's resources among the users, this is accomplished. In this manner, the response time is reduced and each user receives an equitable share of the resources. The limited number of terminals that can be linked to the system at once and the added complexity of multiple instances of the operating system running on each terminal are some drawbacks. Conflicts between users who attempt to use the system at the same time could also arise.

Frequently Asked Question

Q1: What is a time-sharing operating system?

Answer:

A time-sharing operating system is an operating system design that allows multiple users or processes to concurrently share the same system resources, such as the CPU, memory, and peripherals. It enables each user or process to have the illusion of having dedicated access to the system while effectively sharing resources in a time-sliced manner.

Q2: How does a time-sharing operating system work?

Answer:

In a time-sharing operating system, the CPU time is divided into small time intervals called time slices or quantum. Each user or process is allocated a time slice during which it can execute its tasks. The operating system switches between processes rapidly, giving the illusion of parallel execution.

The operating system uses scheduling algorithms, such as round-robin or priority-based scheduling, to determine which process should be allocated the CPU next. Context switching, which involves saving and restoring the execution context of each process, is performed to switch between processes.

Q3: Can a time-sharing system prioritize certain processes or users?

Answer:

Yes, time-sharing systems can prioritize certain processes or users based on specific criteria. Scheduling algorithms often include priority-based scheduling, where processes with higher priority are

allocated more CPU time or given preference in execution. This allows critical or interactive tasks to receive faster response times. However, it's important to balance priorities to ensure fairness and prevent lower priority processes from being starved of resources.

"GeeksforGeeks helped me ace the GATE exam! Whenever I had any doubt regarding any topic, GFG always helped me and made my concepts quiet clear." - Anshika Modi | AIR 21

Choose GeeksforGeeks as your perfect GATE 2025 Preparation partner with these newly launched programs

[GATE CS & IT- Online](#)

[GATE DS & AI- Online](#)

[GATE Offline \(Delhi/NCR\).](#)

Over 150,000+ students already trust us to be their GATE Exam guide.
Join them & let us help you in opening the GATE to top-tech IITs & NITs!



rajku...

45

Previous Article

Multiprogramming in Operating System

Next Article

What is a Network Operating System?

Similar Reads

Difference between Time Sharing OS and Real-Time OS

Prerequisite - Types of Operating Systems Time sharing operating system allows concurrent execution of programs through rapid switching thereby...

2 min read

Difference between Time Sharing OS and Multiprogramming OS

Prerequisite - Different types of Operating System 1. Time Sharing : Time Sharing is the logical extension of multiprogramming, in this time sharing...

3 min read

Real Time Operating System (RTOS)

Real-time operating systems (RTOS) are used in environments where a large number of events, mostly external to the computer system, must be...

4 min read

System Protection in Operating System

Introduction: System protection in an operating system refers to the mechanisms implemented by the operating system to ensure the security...

3 min read

User View Vs Hardware View Vs System View of Operating System

User View of Operating System: The Operating System is an interface, hides the details which must be performed and present a virtual machine ...

1 min read

[View More Articles](#)

Article Tags :

[Operating Systems](#)

[Write From Home](#)



Corporate & Communications Address:- A-143, 9th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)
| Registered Address:- K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305



Company

- About Us
- Legal
- In Media
- Contact Us
- Advertise with us
- GFG Corporate Solution
- Placement Training Program
- GeeksforGeeks Community

DSA

- Data Structures
- Algorithms
- DSA for Beginners
- Basic DSA Problems
- DSA Roadmap
- Top 100 DSA Interview Problems
- DSA Roadmap by Sandeep Jain
- All Cheat Sheets

Web Technologies

- HTML
- CSS
- JavaScript
- TypeScript
- ReactJS
- NextJS

Languages

- Python
- Java
- C++
- PHP
- GoLang
- SQL
- R Language
- Android Tutorial
- Tutorials Archive

Data Science & ML

- Data Science With Python
- Data Science For Beginner
- Machine Learning Tutorial
- ML Maths
- Data Visualisation Tutorial
- Pandas Tutorial
- NumPy Tutorial
- NLP Tutorial
- Deep Learning Tutorial

Python Tutorial

- Python Programming Examples
- Python Projects
- Python Tkinter
- Web Scraping
- OpenCV Tutorial
- Python Interview Question

Bootstrap
Web Design

Django

Computer Science

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths
Software Development
Software Testing

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

Interview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

GeeksforGeeks Videos

DSA
Python
Java
C++
Web Development
Data Science
CS Subjects

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved