Real Time System

A real-time system (RTS) means that the system is subjected to real-time, i.e., the response should be guaranteed within a specified timing constraint or the system should meet the specified deadline. For example, flight control systems, real-time monitors, etc.

Types of RTS based on timing constraints:

1. Hard real-time system: This type of system can never miss its deadline. Missing the deadline may have disastrous consequences. The usefulness of results produced by a hard real-time system decreases abruptly and may become negative if tardiness increases. Tardiness means how late a real-time system completes its task with respect to its deadline. Example: Flight controller system.
2. Soft real-time system: This type of system can miss its deadline occasionally with some acceptably low probability. Missing the deadline have no disastrous consequences. The usefulness of results produced by a soft real-time system decreases gradually with an increase in tardiness. Example: Telephone switches.
3. Firm Real-Time Systems: These are systems that lie between hard and soft real-time systems. In firm real-time systems, missing a deadline is tolerable, but the usefulness of the output decreases with time. Examples of firm real-time systems include online trading systems, online auction systems, and reservation systems.

Terms related to RTS

- Job: A job is a small piece of work that can be assigned to a processor and may or may not require resources.
- Task: A set of related jobs that jointly provide some system functionality.
- Release time of a job: It is the time at which the job becomes ready for execution.
- Execution time of a job: It is the time taken by the job to finish its execution.
- Deadline of a job: It is the time by which a job should finish its execution. Deadline is of two types: absolute deadline and relative deadline.

- Response time of a job: It is the length of time from the release time of a job to the instant when it finishes.

Advantages of RTS

- Real-time systems provide immediate and accurate responses to external events, making them suitable for critical applications such as air traffic control, medical equipment, and industrial automation.
- They can automate complex tasks that would otherwise be impossible to perform manually, thus improving productivity and efficiency.
- Real-time systems can reduce human error by automating tasks that require precision, accuracy, and consistency.
- They can help to reduce costs by minimizing the need for human intervention and reducing the risk of errors.
- Real-time systems can be customized to meet specific requirements, making them ideal for a wide range of applications.

Disadvantages of RTS

- Real-time systems can be complex and difficult to design, implement, and test, requiring specialized skills and expertise.
- They can be expensive to develop, as they require specialized hardware and software components.
- Real-time systems are typically less flexible than other types of computer systems, as they must adhere to strict timing requirements and cannot be easily modified or adapted to changing circumstances.
- They can be vulnerable to failures and malfunctions, which can have serious consequences in critical applications.
- Real-time systems require careful planning and management, as they must be continually monitored and maintained to ensure they operate correctly.

Real-Time Applications

A real-time application (or RTA), is an application that functions within a time frame that the user senses as immediate or current. The latency must be less than a defined value, usually measured in seconds. The use of RTA is part of real-time computing.

A RTA is designed to function within a time frame that is effectively immediate for the user. The processing response happens so swiftly that any interaction seems to be occurring in real-time. However, some common characteristics of RTA are:

1. Low Latency: Latency refers to the time it takes for a signal sent from a source to reach its destination. Low latency means this duration is extremely short, typically measured in milliseconds. For RTA to feel "real-time," the lag or delay between an action and its result should be virtually imperceptible to the user. Whether it's a gamer needing immediate feedback in a virtual world or a trader executing a stock buy, delays can compromise user experience or even have financial implications.

2. Synchronous Interaction: Synchronous interactions in RTA refer to the continuous and active two-way exchange of information, where both parties can send and receive data almost simultaneously. This characteristic mirrors face-to-face conversations where there's an ongoing exchange without significant pauses. In a RTA, such as a video call, any delay can make the conversation feel unnatural or disjointed.

3. Push Mechanism: Instead of waiting for the client to request data, in a push mechanism, the server sends data to the client proactively whenever there's new or updated information. The push mechanism ensures users receive timely updates without manual intervention. This is vital in scenarios like receiving instant messages or real-time notifications, where waiting for the user to request updates would defeat the purpose of "real-time" communication.

Other characteristics will depend on the specific type of RTA. For instance, state-fullness is important in RTA like chat or online games, as these need to maintain a user's state to function effectively. This could be the user's current position in a game or the last message they received in a chat. Whereas fault tolerance is more important in RTA like

video conferencing, where any disruption can be immediately noticeable. Thus, these applications must be designed to handle and recover from unexpected issues.

Some RTA examples:

Real-time System has applications in various fields of the technology. Here we will discuss the important applications of real-time system.

Industrial application

industries. Systems are built in real time to provide optimum and accurate output. Most industrial companies employ real-time technologies to do this. These solutions, in some way, result in higher performance and production in less time. Examples of industrial applications include: automated car assembly plants, chemical plants, and so on.

Medical Science application

In the field of medical science, real-time system has a huge impact on the human health and treatment. Due to the introduction of real-time system in medical science, many lives are saved and treatment of complex diseases has been turned down to easier ways. People specially related to medical, now feel more relaxed due to these systems. Some of the examples of medical science applications are: Robot, MRI Scan, Radiation therapy etc.

Peripheral Equipment applications

Real-time system has made the printing of large banners and such things very easier. Once these systems came into use, the technology world became more strong. Peripheral equipment's are used for various purposes. These systems are embedded with microchips and perform accurately in order to get the desired response. Some of the

examples of peripheral equipment applications are: Laser printer, fax machine, digital camera etc.

Telecommunication applications

Real-time system maps the world in such a way that it can be connected within a short time. Real-time systems have enabled the whole world to connect via a medium across internet. These systems make the people connect with each other in no time and feel the real environment of togetherness. Some examples of telecommunication applications of real-time systems are: Video Conferencing, Cellular system etc.

Defense applications

In the new era of atomic world, defense is able to produce the missiles which have the dangerous powers and have the great destroying ability. All these systems are real-time system and it provides the system to attack and also a system to defend. Some of the applications of defense using real time systems are: Missile guidance system, anti-missile system, Satellite missile system etc.

Aerospace applications

The most powerful use of real time system is in aerospace applications. Basically hard real time systems are used in aerospace applications. here the delay of even some Nano second is not allowed and if it happens, system fails. Some of the applications of real-time systems in aerospace are: Satellite tracking system, Avionics, Flight simulation etc.

RTAs behavior

Many real-time applications involve a continuous flow of information that is transferred from one location to another over an extended period of time. This type of interaction is called a stream and it implies the transfer of a sequence of related

information; for example, video and audio data. It is also used frequently to describe a sequence of data associated with real-world events, for instance, as emitted by sensors, devices, or other applications.

Processing these kinds of events is thus referred to as event stream processing or data stream processing. The latter is taken to mean processing over data carried in the stream of events, essentially with the same meaning as event stream processing.

The Need for RTS

Growing global connectivity, changing consumer demands for always-available data, and always-on, sensor-enabled enterprise environments are driving the creation, collection, and analysis of exponential amounts of data. By 2025, IDC estimates that there will be 79.41 zettabytes of data created and nearly 30 percent 2 of it will require real-time processing enabled by real-time systems.

The need for real-time processing is especially crucial for businesses in robotics, manufacturing, healthcare, and high-precision industries, such as oil and gas and power, that rely on real-time data for continuous improvement in safety, efficiency, and reliability.

One key factor in ensuring data is processed in real-time for businesses in these types of industries is a system's ability to prioritize, manage, and execute real-time workloads over non-real-time workloads.

For example, modern automotive manufacturers are highly reliant on robots to work together on a production line to assemble a car. The robots will pass each other parts, drill or weld, or perform safety inspections—all of which require a high level of precision and meticulous timing. In this use case, a real-time system must have the ability to not only process data in a defined, predictable time frame but also ensure that critical tasks, such as safety-related workloads, are completed prior to less critical tasks.