

## Asynchronous Communication

Now, when we are already aware of synchronous communication, the process is blocked until the request is complete. Asynchronous is opposite to it. You mail your colleague regarding some office work. You are not supposed to halt your daily activities, wait for the email response, and then continue your routine. When you do not need to wait and can do anything, communication is an example of asynchronous communication.

Asynchronous communication is “Communication not in sync”.

### What is asynchronous communication?

Asynchronous communication is a type of communication when the client has the freedom to start or pause other processes instead of waiting for the response. The client can do anything on the application without waiting for the response, as it can take some time for the response to arrive.

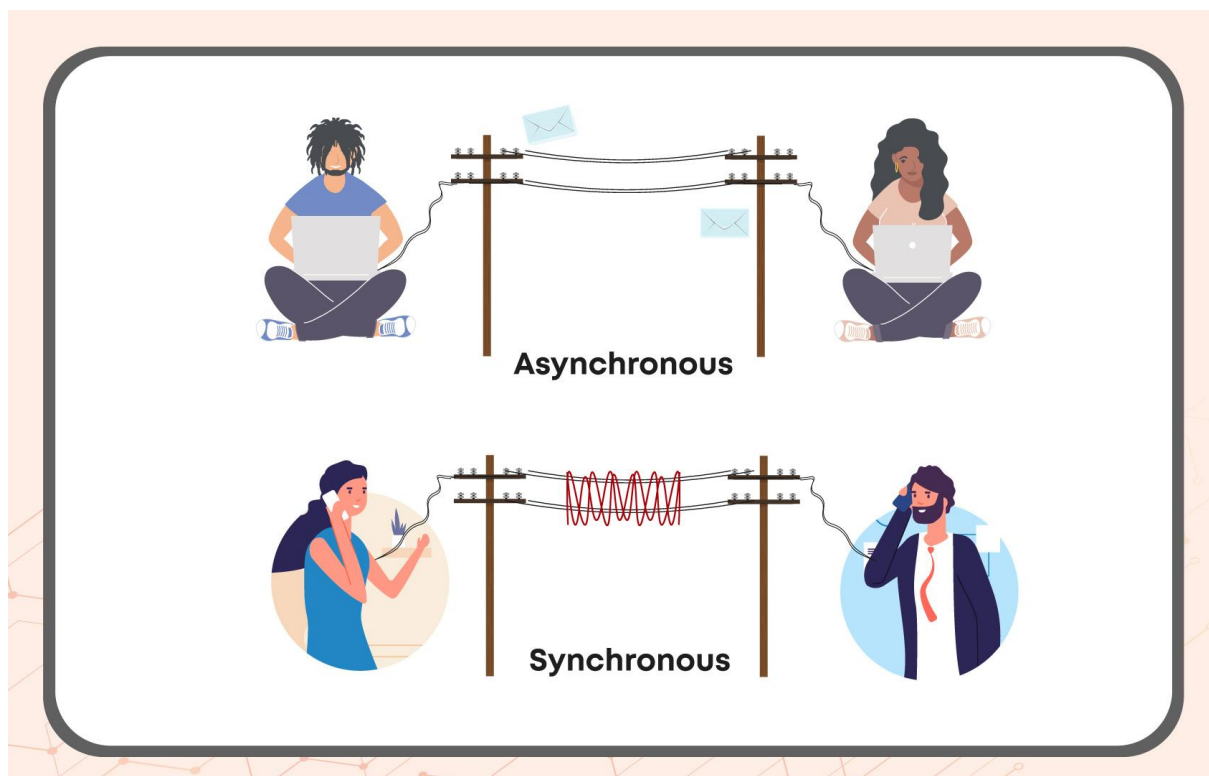


Fig: Asynchronous communication vs Synchronous Communication

Examples of asynchronous communication:

- Messaging
- Email
- Intranet
- Social Media
- NewsGroups
- World Wide Web

### **Why is asynchronous communication needed?**

1. Computation takes a lot of time:

If the computation and pre-processing for a response takes a lot of time, it would be more efficient for the client to start some other process and revisit when possible or available.

Example: The notification service in a company can be slow because of network congestion or numerous requests. In such a situation instead of creating an open connection, it is better to use asynchronous communication that would save time and resources.

2. Scalability/Performance of the application:

Since the client or the process is not blocked or doesn't have to wait for the response, time and resources are saved, leading to better performance of the application.

Example: You have bought some product from amazon, and the last step of ordering is the confirmation email. It might take a lot of time to receive the notification email depending on many factors like traffic etc.; the client can resume other activities instead of sitting idle and waiting for the email.

3. Huge Rate Differences:

If the request rate of the client is much higher than the response rate, then the other part will lead to cascading failure(Failure due to overloading systems) because of increased load.

Example: Application A makes requests with a rate of 1000 requests/second. At the same time, application B can only deliver 200 responses/second. After some time, application B would be overloaded and breakdown leading to cascading failure because of the considerable differences in the rates.

### **How do Asynchronous Communication works?**

1. Request-Response Pattern/Tracking ID Based:

A tracking id is issued by the application to the request. The requesting application/client can check the job's progress using that tracking id, and a response is delivered once the computation is completed.

Example: Application A makes a request to application B. Suppose the request would take 1 hour to complete. Application B gives application A a tracking id, say S1. Application A keeps checking the status using id S1. After one hour, when the computation or pre-processing is completed, the response is delivered by application B.

2. Message Queue Based:

There is no direct communication between both ends. The request is pushed in the queue for application 2. Once the request is completed, Application B notifies application A that the job is complete.

**Applications of Asynchronous Communication:**

1. Ecommerce/Booking Involving:

Any application involving Booking confirmation emails/messages/notifications use asynchronous communication because there is no fixed time that the user needs to wait to receive the confirmation message.

2. Bank:

A bank has multiple sections like transactional service and reporting service. Once a transaction is made, the reporting service has to generate a report for the entire working day. Since it is not an immediate required service with no fixed time, it is also completed using asynchronous communication.

3. Sentiment-prediction system:

The sentiment prediction system can be developed for predicting the sentiments based on tweets on Twitter. Twitter has a dynamic load depending on various factors. Synchronous communication cannot happen here because sometimes the prediction system would have numerous requests, leading to overloading and cascading failure and very few requests. Therefore, we can use queue-based asynchronous communication here.