



**BITS Pilani**  
Pilani | Dubai | Goa | Hyderabad

# Early Middleware Technologies

---

**Srikanth Gunturu**

Guest Faculty  
BITS, WILP

# In this segment

## Early Middleware Technologies

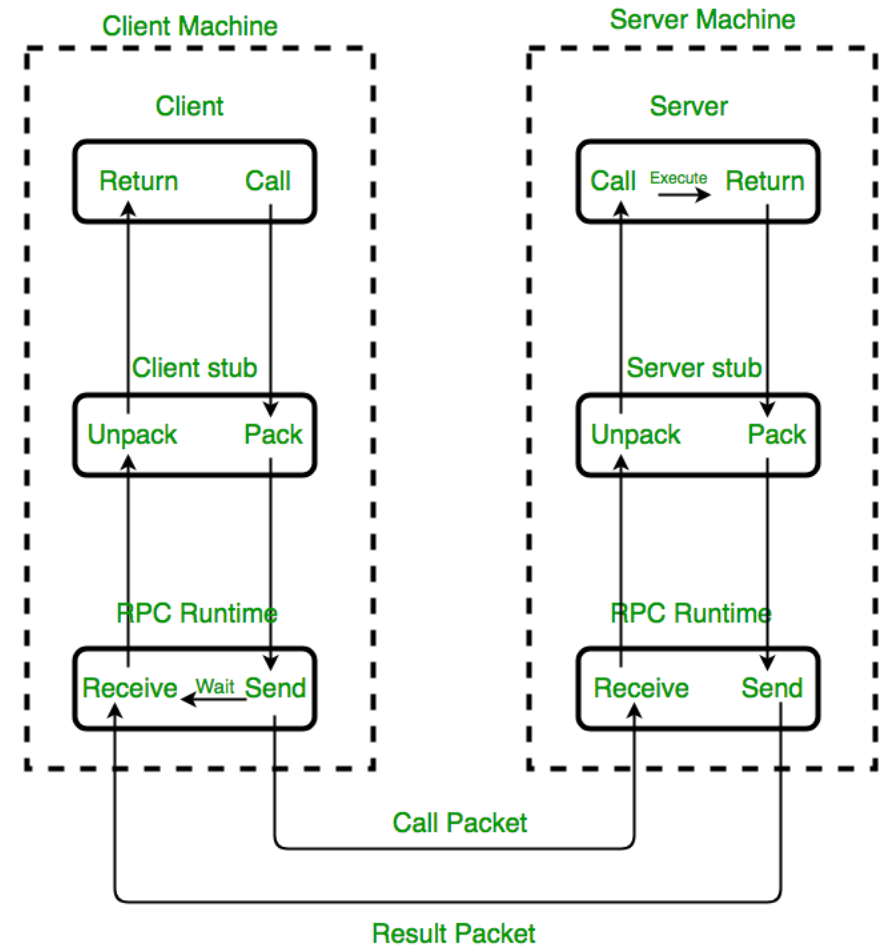
- Remote Procedure Calls
- Distributed Object Oriented Components
- Message Oriented Middleware (MOM)



# Remote Procedure Calls (RPC)

## Overview

- RPC is a method call across the network, which is treated same as a local procedure call (location transparency)
- Open Network Computing (ONC) RPC implementation (early attempt of middleware)
  - RPC methods defined in “.x” files
  - *rpcgen* translates .x files to stubs on client side and skeleton files on server side
- Stubs and skeletons handle packing/unpacking of data from application to network layer (bytes) – called **Marshaling**
- Applications on either side (client and server) are designed as if they run on the same machine, leaving underlying network / transport specifics to **RPC runtime**

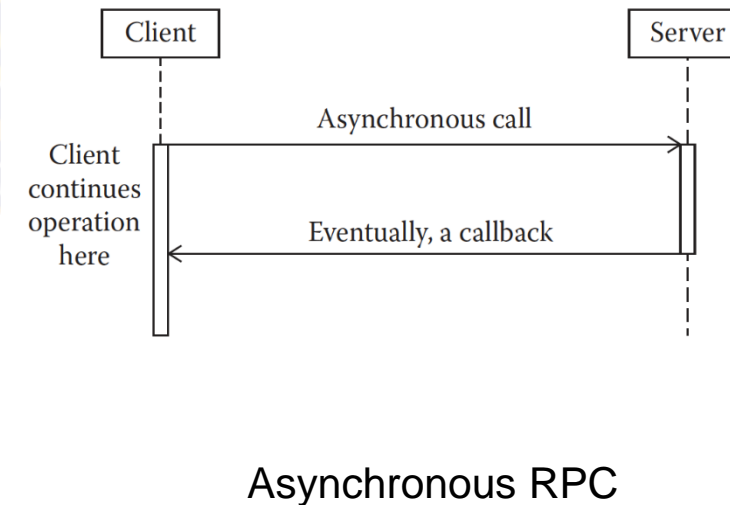
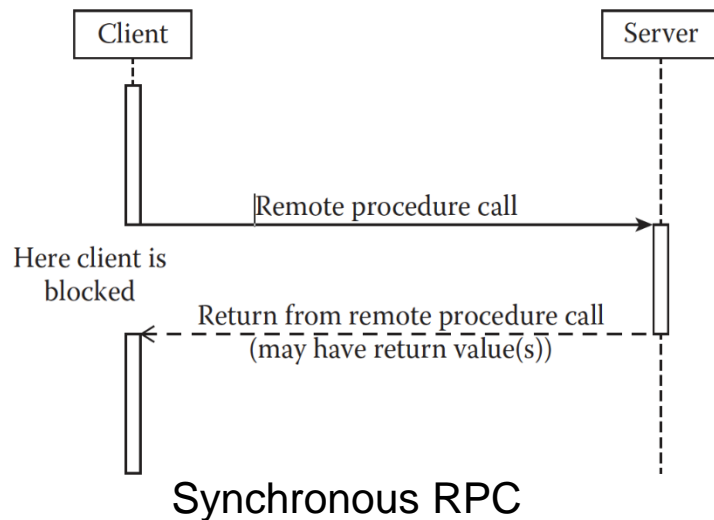


Implementation of RPC mechanism

# Remote Procedure Calls (RPC)

## Synchronous Vs Asynchronous

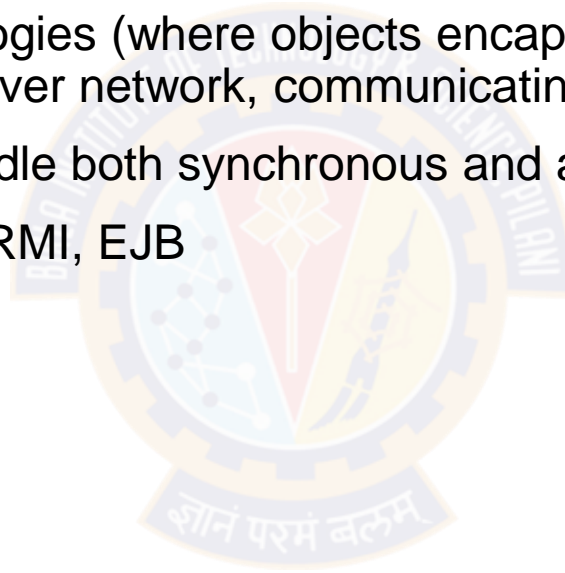
- Synchronous RPC **blocks** the client until server finishes the operation and returns response (or error)
- Asynchronous RPC allows client to continue its business while server processes RPC
  - Client may send return address (along with request) that server can use to **callback** to send response
  - Server may use a mediator (also called **broker**) to return response to the client when finished



# Distributed Object Oriented Components

## Overview

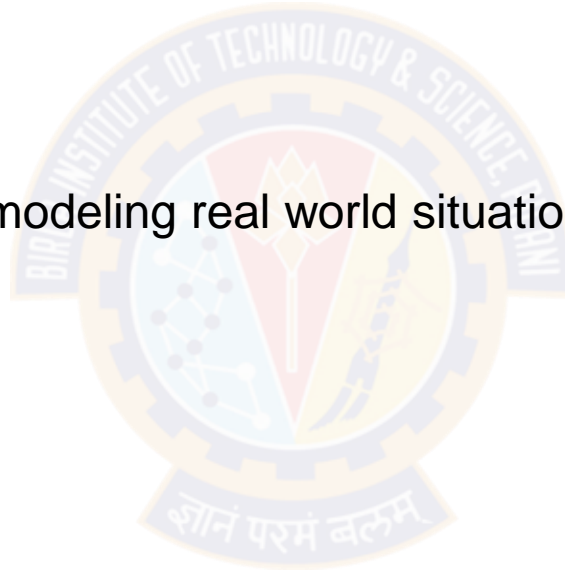
- Distributed component is a concurrent object with a well-defined interface, which is a logical unit of distribution and deployment
- Employs object oriented methodologies (where objects encapsulate data inside) to design a system, with components spread over network, communicating using middleware components
- Distributed Object middleware handle both synchronous and asynchronous communication
- Examples: CORBA, DCOM, Java RMI, EJB



# Distributed Object Oriented Components

## Component or Object Oriented Middleware ?

- Component based systems comprises of loosely coupled components (need not be object oriented), most of which could be existing components
  - CORBA
  - DCOM
  - EJB
- Object oriented systems focus on modeling real world situations, albeit using components at times
  - CORBA
  - DCOM

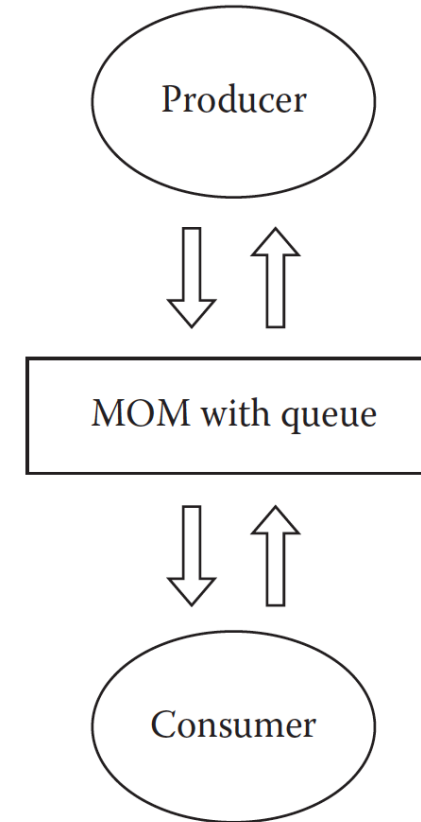




# Message Oriented Middleware (MOM)

## Overview

- Acts as a middleman (broker) between client and server for communication
- Brings in loose coupling in the design of client and server, by offloading communication to a third party
- Enables client to work asynchronously with server
- Generally (not necessarily always) uses queueing mechanism to deliver messages, in a publish/subscribe model
- MOM will handle auxiliary operations as well, like Quality of Service, Priority processing, Reliability, Recovery of lost messages etc.





# Thank You!

In our next session:  
Object Oriented Middleware – CORBA Basics