



UNIVERSITÀ  
DI TORINO

# Client Server Architectures

Prof. Fabio Ciravegna  
Dipartimento di Informatica  
Università di Torino  
[fabio.ciravegna@unito.it](mailto:fabio.ciravegna@unito.it)



# Introduction to Client-Server Architectures

- In the world of computer science and networking,
  - client-server architectures play a fundamental role in enabling communication and collaboration across various devices and systems
- A client-server architecture is a model for **distributed computing**
  - where two distinct entities, the "client" and the "server," interact to fulfil specific tasks or services
  - This model is often used in networked systems, allowing multiple devices to work together efficiently

# The Client

- The client's primary role is to request services or data from the server
  - it is typically a user-facing device or application, such as a personal computer, smartphone, or web browser
    - When you open a web browser and access a website, your browser acts as the client,
      - sending requests to a remote server that hosts the website's content
- However a server can also play the role of client to another server

# The Server

- The server is computer or software application designed to respond to client requests
  - It stores and manages resources, data, or services that clients need
  - Servers can vary widely in their functions,
    - from web servers that deliver web pages
    - to database servers that store and retrieve data

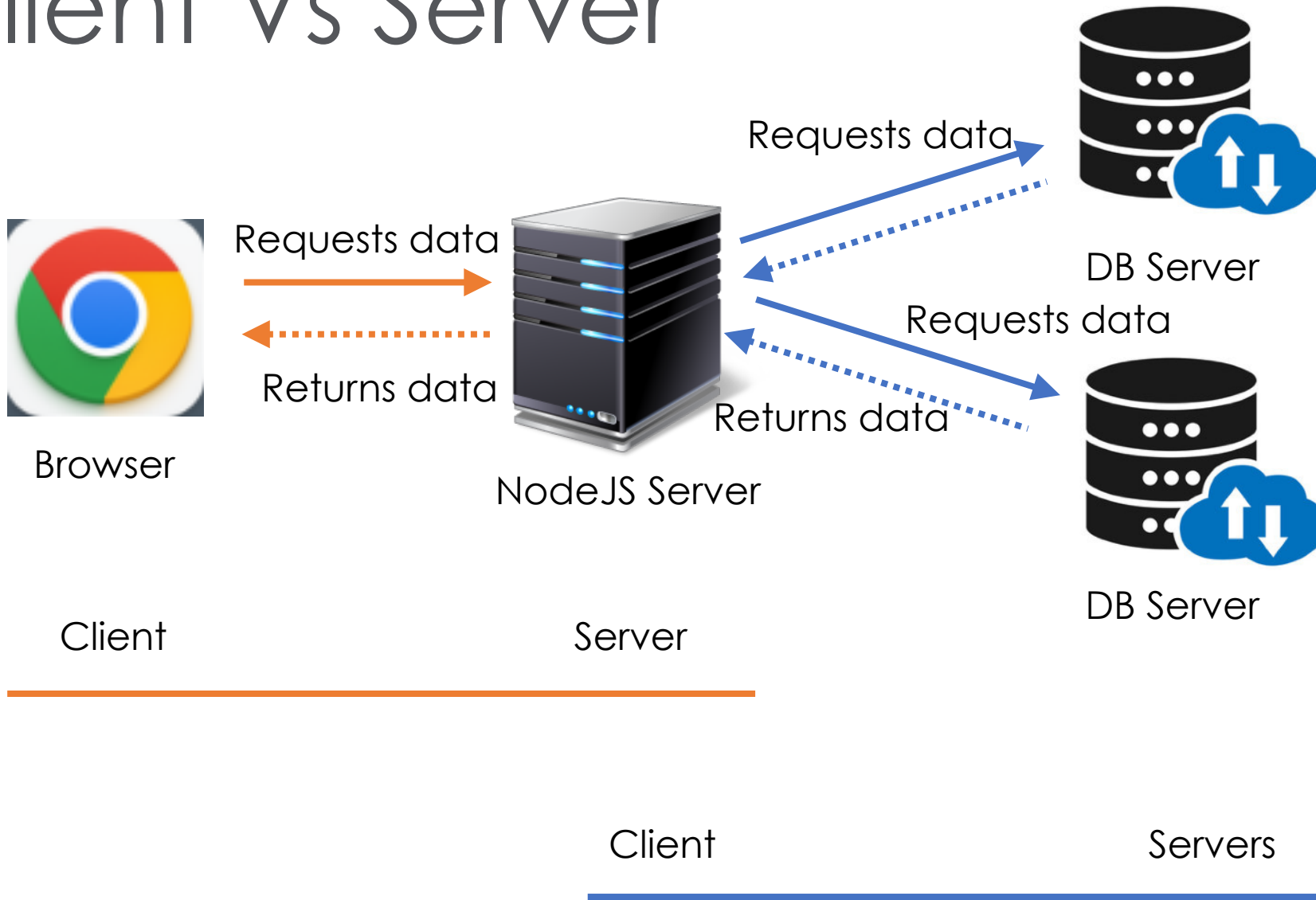
# How it works

- The client-server architecture operates through a simple request-response mechanism
  - When a client needs something,
    - it sends a request to the server, specifying what it wants
  - The server processes the request
    - and sends back a response containing the requested information
- This interaction can happen in real-time, creating a seamless user experience
- We will see different strategies of interaction
  - original (REST) client-server interaction
  - proactive servers
  - streaming, etc.

# Key Characteristics

- Scalability:
  - Client-server architectures are highly scalable, allowing additional clients and servers to be added as needed to handle increased demand
- Centralised Control:
  - Servers typically have centralised control over resources or data, ensuring data consistency and security
- Efficiency:
  - By distributing tasks between clients and servers, client-server architectures can optimise resource utilisation and enhance overall system performance
- Reliability:
  - Redundancy and failover mechanisms can be implemented to ensure high availability and reliability of services

# Client Vs Server







UNIVERSITÀ  
DI TORINO

# Next: the HTTP Protocol

