**Component Diagram:**

In the Unified Modeling Language, a **component diagram** depicts how components are wired together to form larger components and or software systems. They are used to illustrate thestructure of arbitrarily complex systems.

## Overview:

Component diagrams are different in terms of nature and behavior. Component diagrams are used to model physical aspects of a system.

Now the question is what are these physical aspects Physical aspects are the elements like executables, libraries, files, documents etc which resides in a node.

So component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

**Purpose:**

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.

So from that point component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files etc.

Component diagrams can also be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment.A single component diagram cannot represent the entire system but a collection of diagrams are used to represent the whole.

So the purpose of the component diagram can be summarized as:Visualize the components of a system.Construct executables by using forward and reverse engineering.Describe the organization and relationships of the components.

Now the question is what are these physical aspects? Physical aspects are the elements like executables, libraries, files, documents etc which resides in a node.

So component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

**Data Base Server**

**Desktop Client**

**Machine**

**User Interface**

**User**

**View**

**Utility**

**Data**

**Processing**

**Users**

**Resides**