



Deployment Diagram

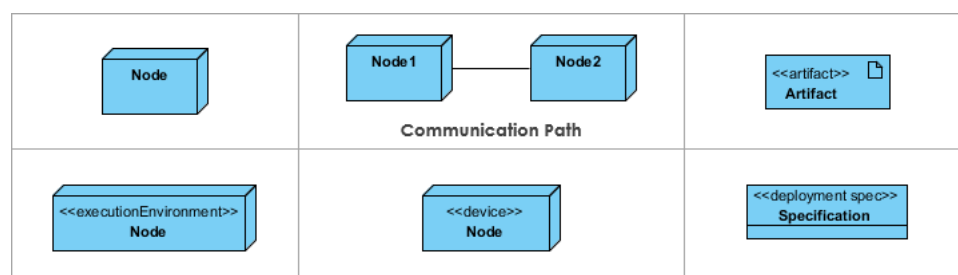
Written Date : September 02, 2016

A deployment diagram is a diagram that shows the configuration of run time processing nodes and the components (hardware, software & middleware on hardware) that live on them that is used to model the static deployment view of a system (topology of the hardware).

1. A special kind of class diagram that focuses on a system's nodes.
2. Show the structure of the run-time system.
3. It one of the two kinds of UML diagrams used for the physical aspects of an OO system.

Graphically, a deployment diagram is a collection of vertices and arcs. Deployment diagrams commonly contain **Nodes** and **Dependency & association relationships**. It may also contain notes and constraints.

1. 3-D box represents a node, either software or hardware
2. Hardware node can be signified with <<stereotype>> i.e. <<artifact>> or <<mainframe>>
3. Connections between nodes are represented with a line, with optional <<stereotype>>
4. Nodes can reside within a node



Deployment Planning

1. Design and plan how will your system be installed?
2. Determine if different versions of the system will be deployed at the same time, how will you resolve differences?
3. What physical sites do you need to deploy to and in what order?
4. How will you train your users?
5. What backups do you need before installation?

6. Do you need to do a data conversion?

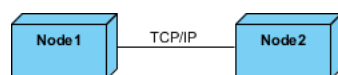
How to Develop a Deployment diagram?

Firstly, identify the nodes that represent your system's client and server processors and then highlight those devices that are relevant to the behavior of your system.

For example, you'll want to model special devices, such as credit card readers, badge readers, and display devices other than monitors, because their placement in the system's hardware topology are likely to be architecturally significant. # Provide visual cues for these processors and devices via stereotyping. # Model the topology of these nodes in a deployment diagram. Similarly, specify the relationship between the components in your system's implementation view and the nodes in your system's deployment view.

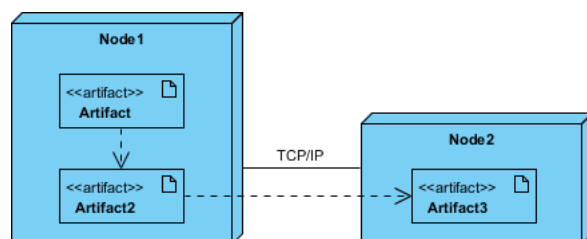
Nodes

A node usually represents a piece of hardware in the system. A connection depicts the communication path used by the hardware to communicate usually indicates the method i.e. TCP/IP

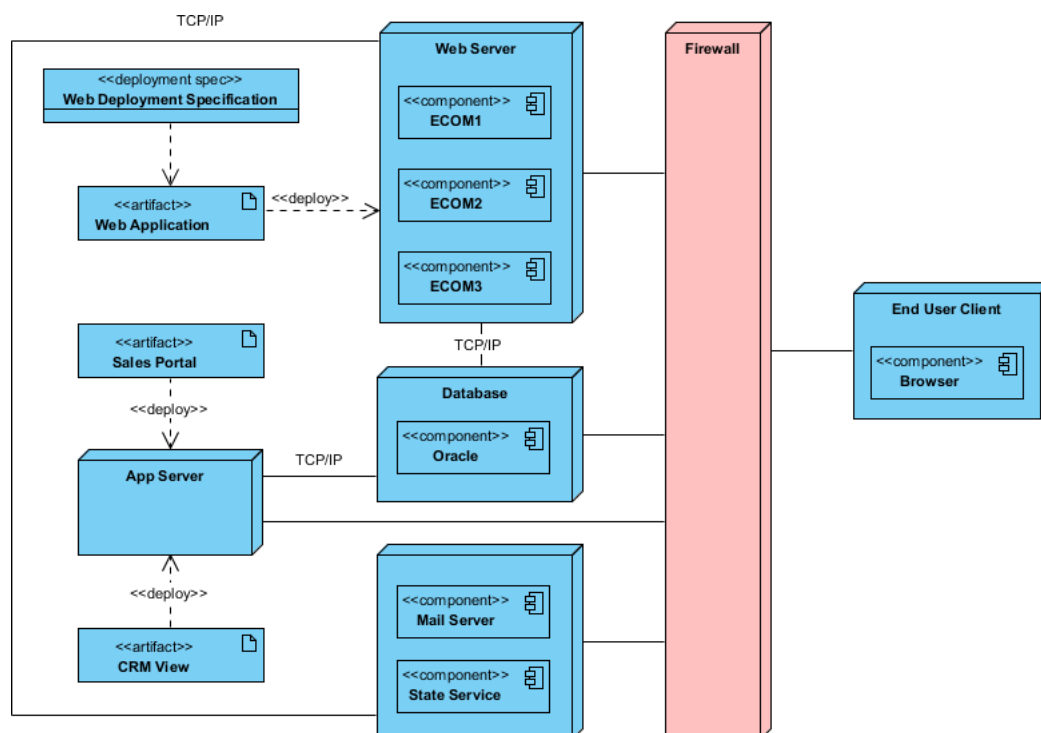


Artifacts

1. An artifact is the specification of a physical piece of information, such as, source files, binary executable files, table in a database system.
2. An artifact is defined by the user represents a concrete element in the physical world.



Deployment Diagram Example



Visual Paradigm supports deployment diagram and other UML diagram types. You can find all the tools you need in modeling deployment views and other aspects of software systems.

[Try now for FREE](#)

References

1. [How to draw a Deployment Diagram in UML - Visual Paradigm User's Guide](#)



Visual Paradigm home page
(<https://www.visual-paradigm.com/>)

Visual Paradigm handbooks
(<https://www.visual-paradigm.com/learning/handbooks/>)