

Component diagram

A Component diagram falls under the structural diagramming family. A Component diagram describes logical components that make up the system.

A [Component](#) contains information about the logical [Class](#) or Classes that it implements, thus creating a mapping from a logical view to a component view. Dependencies between the Components makes it easy to analyze how a change in one component affects the others. The Components may also be shown with any of the [Interfaces](#) that they expose. They, as with almost any other model elements, can be grouped into [Packages](#), much like Classes or [Use Cases](#).

The Component diagrams are used in the later phases of the software development, when there is a need to divide up Classes among different Components. When working with the CASE facilities, the Components are used for file-class mapping during code generation, reverse engineering, and round-trip engineering operations.



As of MagicDraw 17.0.1, the Component diagram (or the [Deployment diagram](#)) replaces the Implementation diagram, which is no longer supported in UML standard. An Implementation diagram created with earlier versions of MagicDraw, now opens as

Related pages

- [Deployment diagram](#), if Nodes *were used* in the Implementation diagram
- [Creating diagrams](#)
- Component diagram, if Nodes *were not used* in the Implementation diagram

Customized diagrams that were based on the Implementation diagram are now based on the Component diagram.