

Component Diagram

Ex.No 7

20CS2050 L – Software Engineering Lab

2021-1-BIV

Component Diagram

Introduction

- Implementation diagram
- Used to visualize the organization and relationships among components in a system
- Used during implementation phase, however, prepared well in advance to visualize the implementation details
- Component – (file, package, library, activeXcomponent, applet, dll, exe)



Component Diagram

Introduction

Application

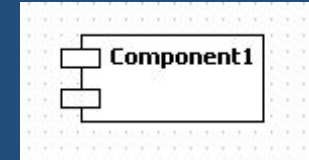
- Component diagrams can be used to –
 - Model the components of a system.
 - Model the database schema.
 - Model the executables of an application.
 - Model the system's source code.



Description

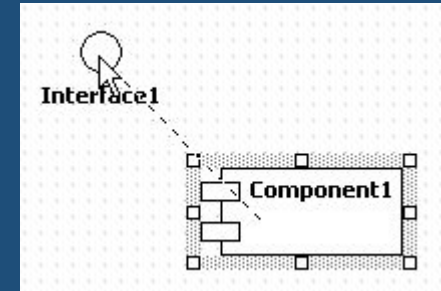
- Component

- A component represents a software entity in a system.
- Examples include source code files, programs, documents and resource files.
- A component is represented using a rectangular box, with two rectangles protruding from the left side



- Interfaces

- Interfaces represent the places where the groups of classes in the component communicate with other system components.

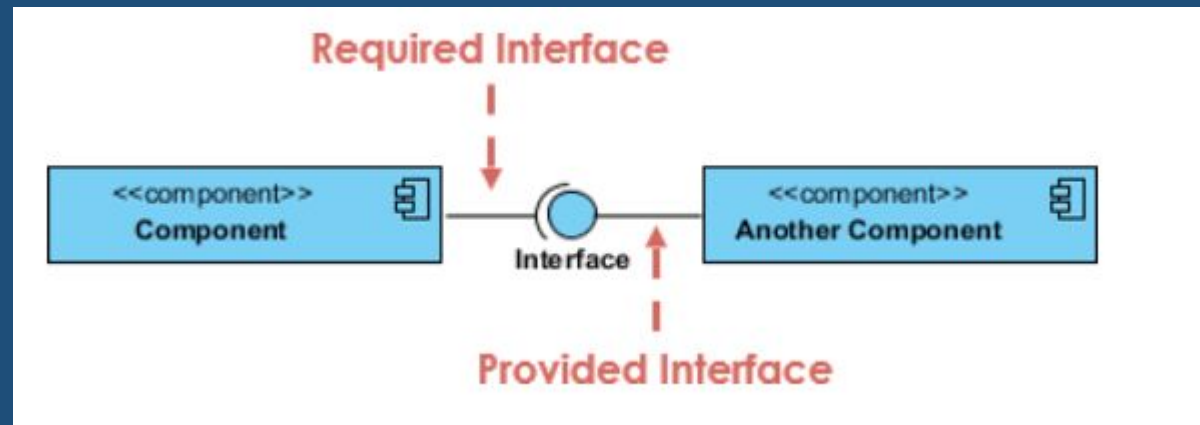


- Dependency

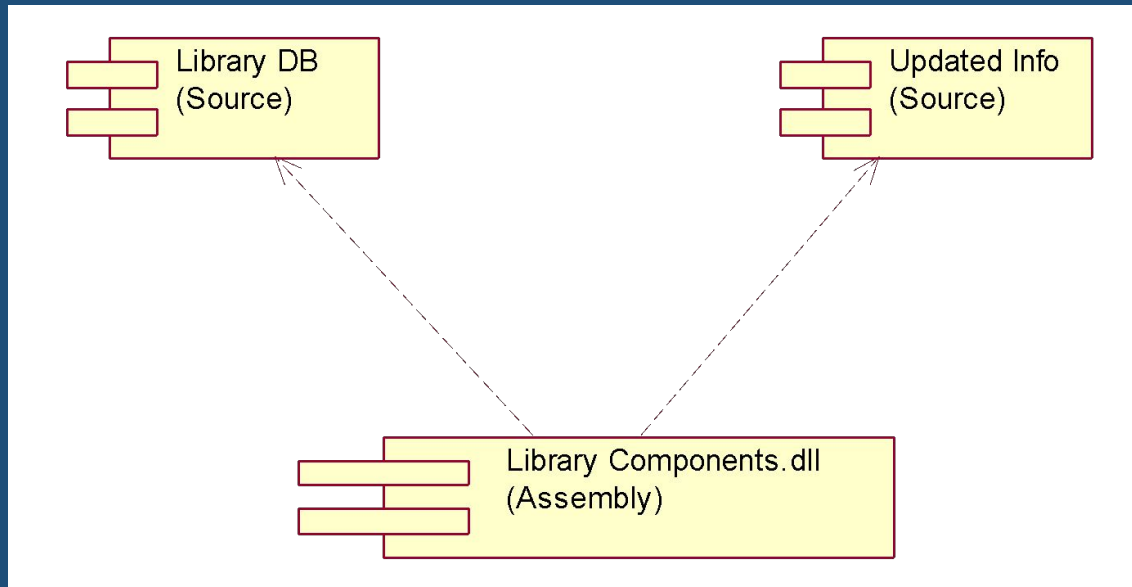
- A Dependency is used to model the relationship between two components
- The notation for a dependency relationship is a dotted arrow, pointing from a component to the component it depends on.

Description

- **Provided interface** symbols with a complete circle at their end represent an interface that the component provides - this "lollipop" symbol is shorthand for a realization relationship of an interface classifier.
- **Required Interface** symbols with only a half circle at their end (a.k.a. sockets) represent an interface that the component requires (in both cases, the interface's name is placed near the interface symbol itself).



Basic Example



Example

