When we have interface hierarchies in both interfaces as well as implementations, then bridge design pattern is used to decouple the interfaces from implementation and hiding the implementation details from the client programs.

When we need bridge design pattern

The Bridge pattern is an application of the old advice, “prefer composition over inheritance”. It becomes handy when you must subclass different times in ways that are orthogonal with one another.

With Bridge Design Pattern

For example, you can solve the above problem by decoupling the Vehicle and Workshop interfaces in the below manner.

list of task:

-create four subclasses with their own interfaces

-implements two objects with bridge model

