Bridge :-

What is Bridge ?

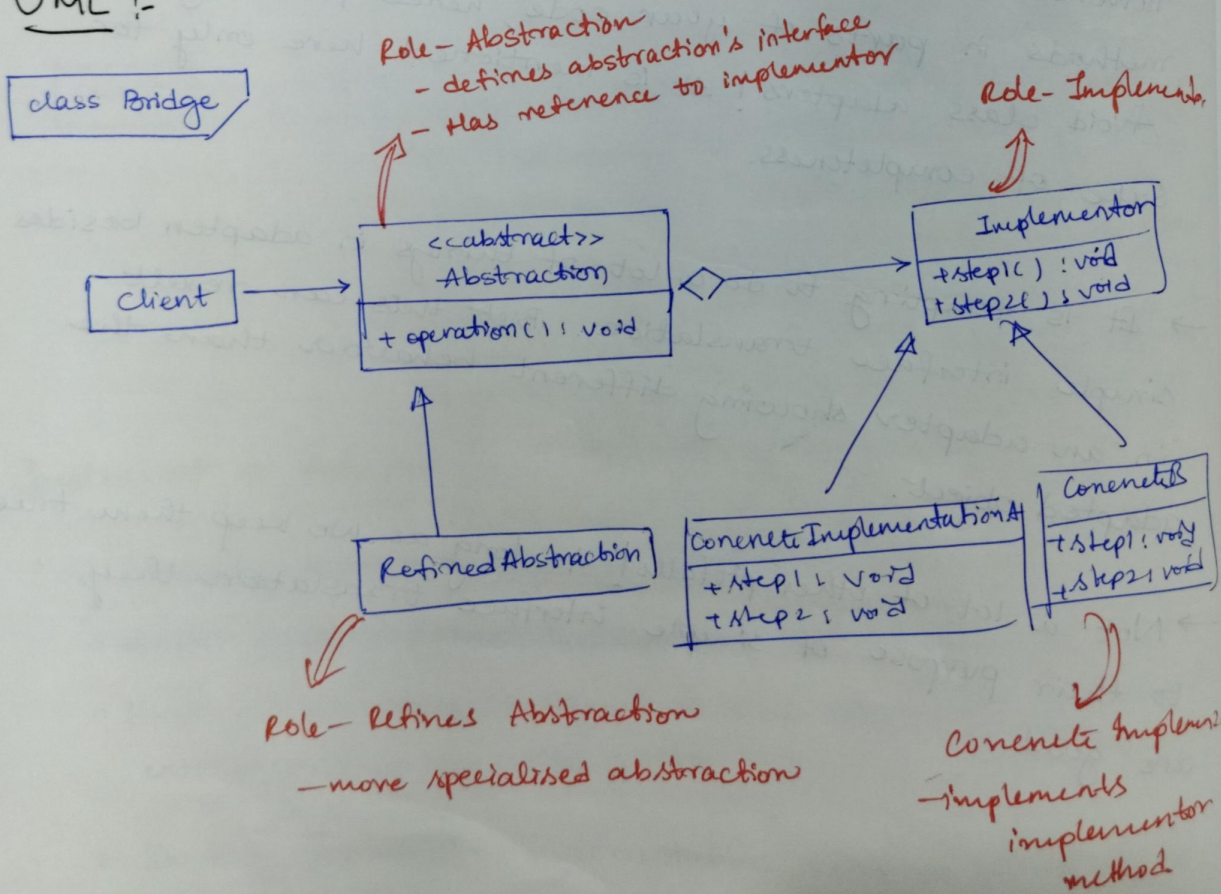
→ Our implementation & abstractions are generally coupled to each other in normal inheritance.

→ Using bridge pattern we can decouple them so they can both change without affecting each other.

→ We achieve this feat by creating two seperate inheritance hierarchies; one for implementation and another for abstraction.

→ We use composition to bridge these two hierarchies.

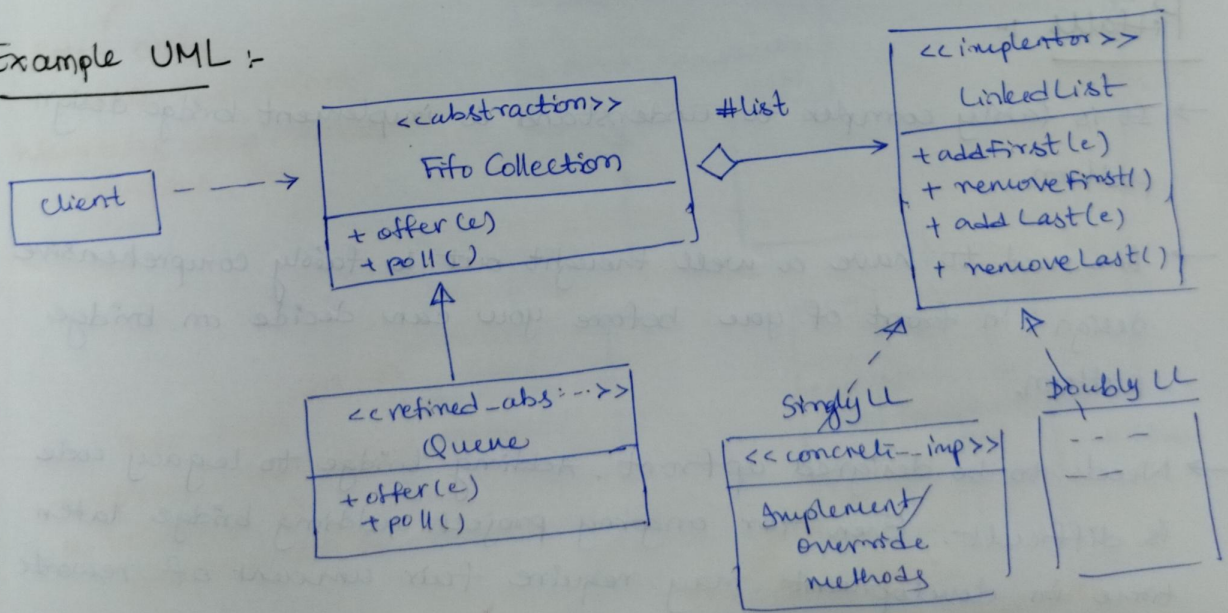
UML :-



Implement a Bridge :-

- We start by defining our abstraction as needed by client.
  - We determine the common base operations and define them in abstraction.
  - We can optionally also define a refined abstraction & provide more specialised operation.
  - Then we define our implementor next. Implementor methods do NOT have to match with the abstracton however abstraction can carry out its work by using implementor methods
  - Then we write one or more concrete implementor providing implementation.
- Abstractions are created by composing them with an instance of concrete implementor which is used by methods in abstraction.

Example UML :-



Implementation Considerations :-

In case we are ever going to have a single implementation then we can skip creating abstract implementor.

Abstraction can decide on its own which concrete implementor to use in its constructor or we can delegate that decision to a third class. In last approach abstraction remains unaware of concrete implementors & provides greater de-coupling.

Design Considerations :-

- Bridge provides great extensibility by allowing us to change abstraction and implementor independently. You can build & package them seperately to modularize overall system.
- By using abstract factory pattern to create abstraction objects with correct implementation you can de-couple concrete implementors from abstraction.

Pitfalls :-

→ It is fairly complex to understand & implement bridge design pattern.

→ You need to have a well thought out & fairly comprehensive design in front of you before you can decide on bridge pattern.

→ Needs to be designed upfront. Adding bridge to legacy code is difficult. Even for ongoing project adding bridge later time in development may require fair amount of rework.