

# Bridge DP

4-Jul-22 10:16 PM

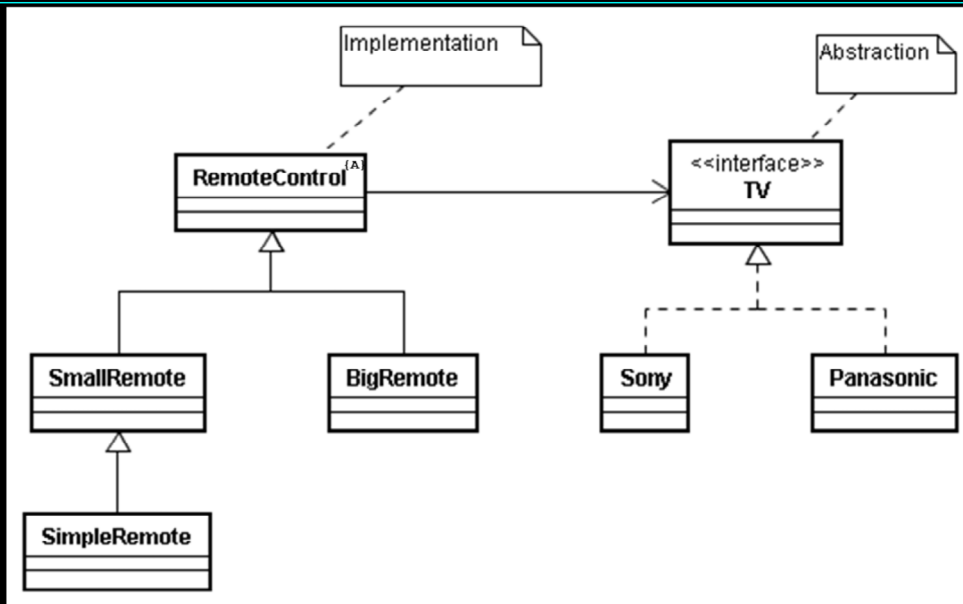
53

## Problem

- Simulate the remote controls for different TVs
  - Sony has 3 different models of TVs and each model has 2 different remote models.
  - Panasonic has 5 different models of TVs and each model has 2 different remote models.



# Solution



4-Jul-22 10:16 PM

55

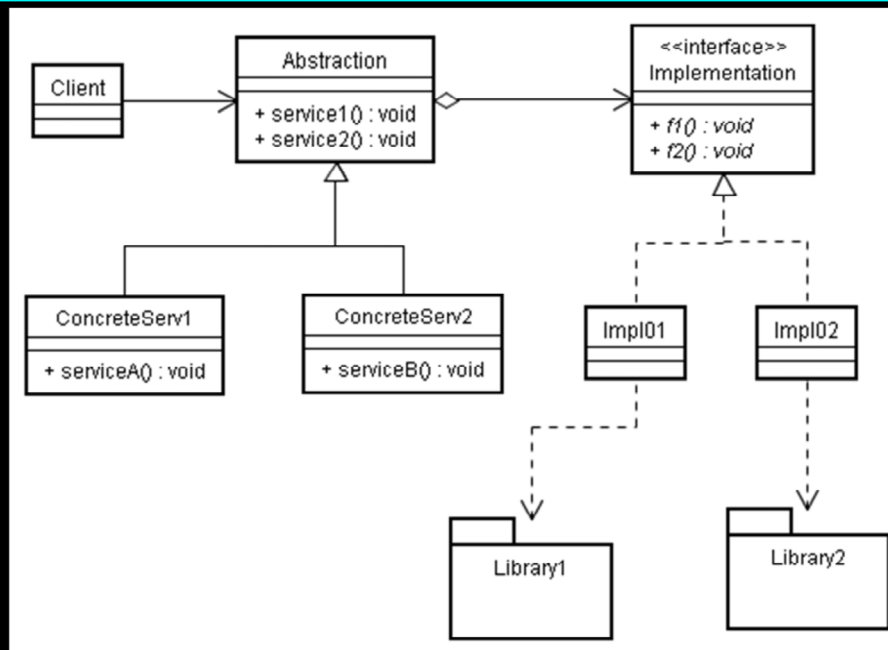
Two class hierarchies evolve independently.

We want to avoid a permanent binding between an front end object (abstraction) and a back end object (implementation).

Both the abstraction and implementation should be extensible.

Changes in implementation should not cause change to the clients using the abstraction.

## Bridge diagram



4-Jul-22 10:16 PM

56

The ConcreteServ extend/provide the functionality of abstraction class.

The abstraction class maintains a reference to the Implementation(s).

The Front-end objects implement their functionality using back-end objects.

=====

Decouple an abstraction (back-end objects) from its implementation (front-end objects) so that the two can vary independently.

We can easily add new kinds of implementations without impacting the abstractions.

The front-end classes can (and usually do) have completely different interfaces from the back end objects

Without this DP there would be a combinatorial explosion of possibilities.

## Bridge and other DPs

- How does Bridge relate to
  - Adapter
  - Abstract Factory

4-Jul-22 10:16 PM

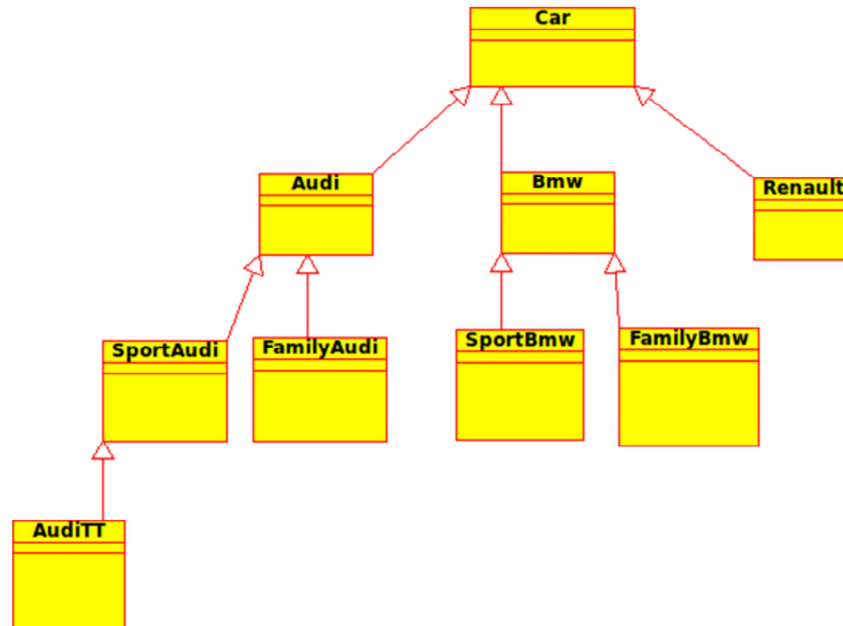
57

Adapter makes things work after they're designed; Bridge makes them work before.

The Bridge design usually uses the Abstract Factory pattern to create the implementation objects.

Bridge DP usually uses Adapter DP for adapting the interface of the concrete classes.

# Improve the Design

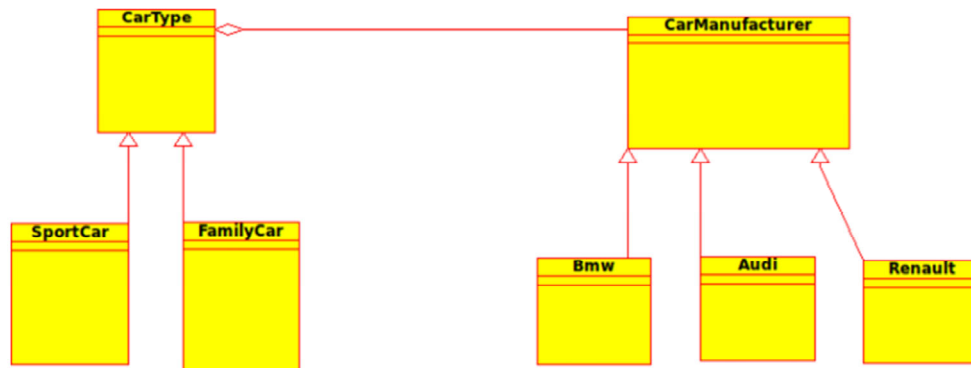


4-Jul-22 10:16 PM

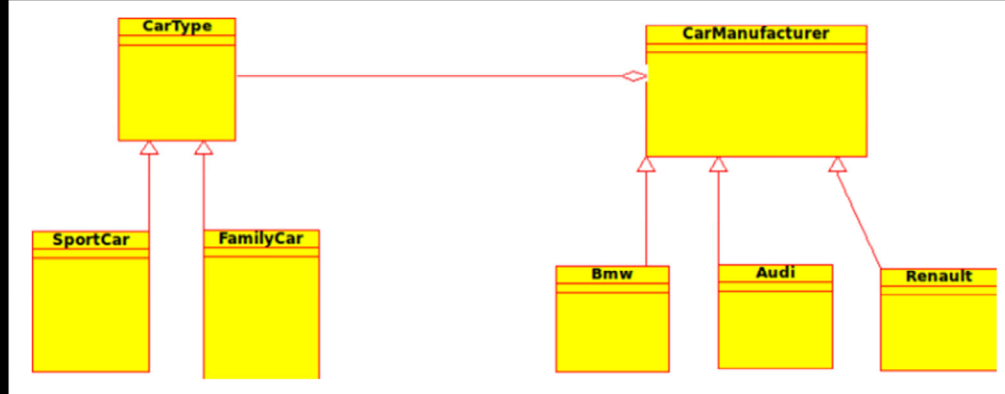
58

?

# Solution - 1



## Solution - 2



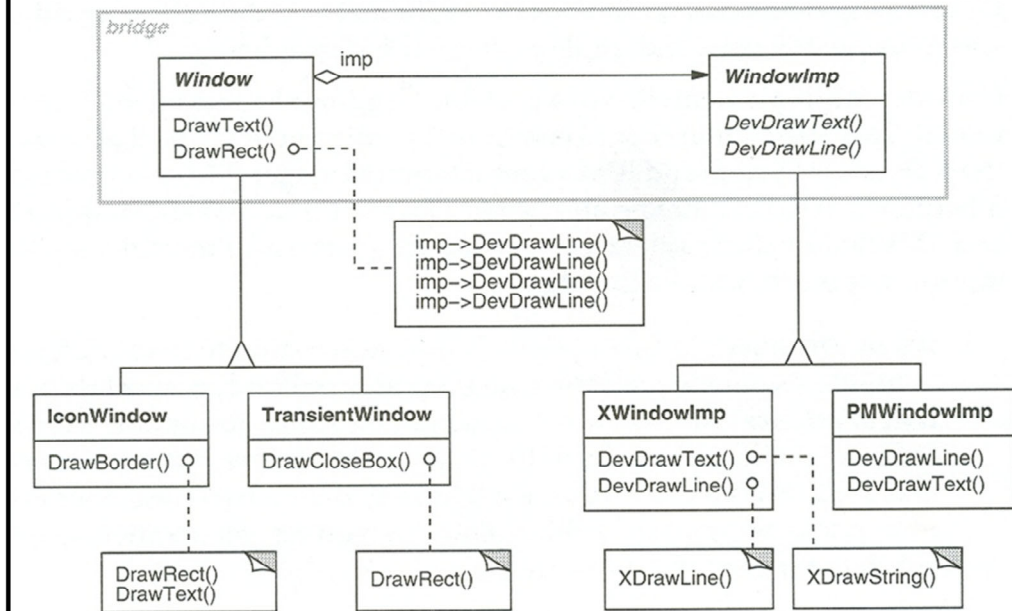


## Assignment

- Design a class diagram to separate the implementation of Window drawing from the display of windows.



# Solution



4-Jul-22 10:16 PM

62

## Non Software Analog

A family of four going to different Restaurants e.g. Chinese, Seafood, North-Indian, etc.

## NOTES:

### Non-Software Analog

My friends and I like to go to together dinner a lot. From the waitress's point of view, we are all the same – she considers us to be *patrons*, and she wants the same question answered no matter which of us she is speaking with: "whatlyahave?"

We are actually all different from each other, of course, but that is not her concern. One difference that influences the outcome of this interaction is our ordering preference.

I usually get what's on special for the day. I like adventure. Christopher (also my younger brother) says I'm crazy, because he says everyone knows "the special" is the thing that's just about to go bad. He says "get the specialty of the house; they take pride it doing that right." Sure, I say, but you always get the same thing if you do that. It's an old argument.

Andrea (also my wife), likes salads, and usually orders one. However, some places don't have them, so then she needs to get something meatless, whatever that might be.

Brenner always gets the most expensive thing on the menu. He's noticed that we always divide the check equally... and well, that's Brenner.

That's one variation, or really, two. The variation of who we are, and the variation of how we order food. These do not vary independently, however. The variation in our ordering preferences is what makes us different, in the context of being restaurant patrons.

The independent variation has to do with the restaurant we're currently visiting on a given evening. For example:

**Ocean Breeze:** Seafood, primarily, and very experimental cuisine.

**Redrocks:** Barbecue, mostly meat, very large portions: not Andrea's favorite place.

So, who we are varies, and the food available varies, but each of us can use the menu in our own way to get what we want. From the waitress's point of view, she just wants an answer.