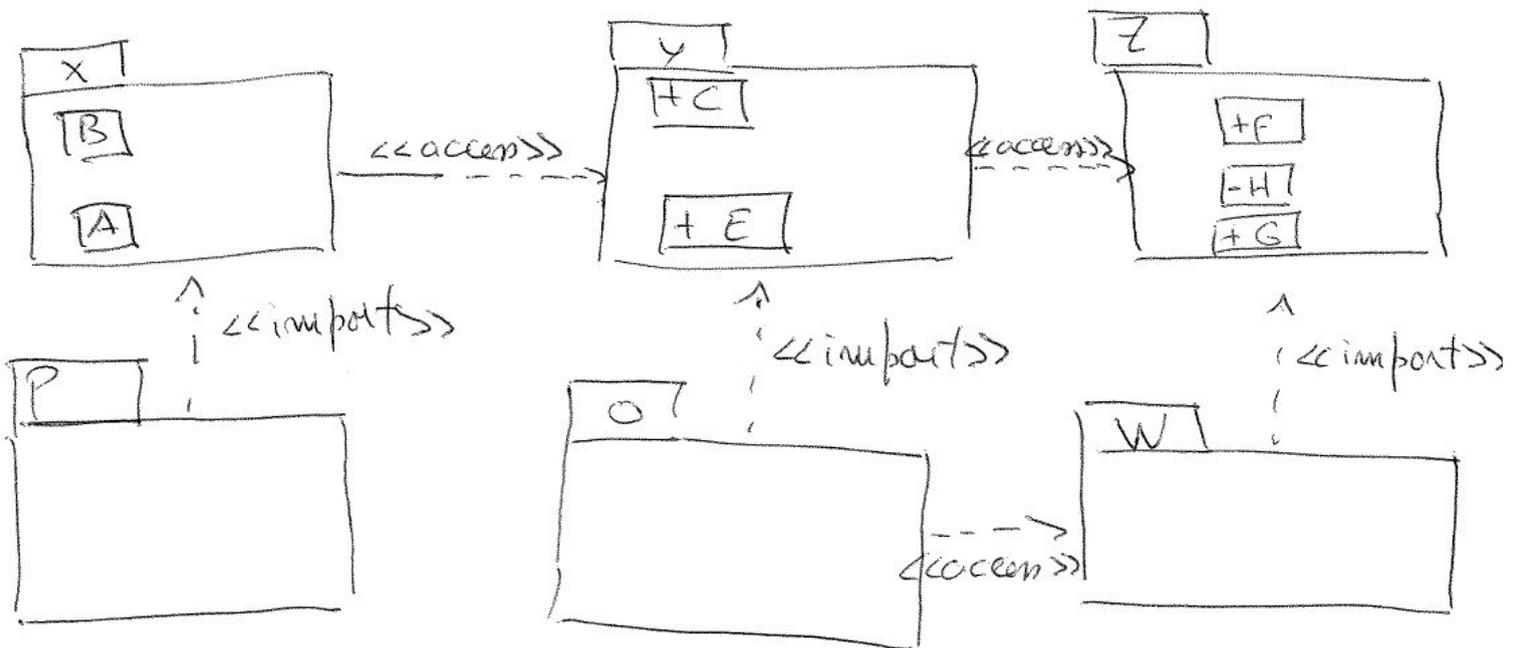


Lab 5: From Analysis to Design Class Diagrams
The Software Development Methods Team
October, 2019

I - Package Diagrams

Consider the following package diagram:



1) Which of the following statements is true?
(choose one)

- A. O can see C and E and F
- B. W can see F and H and G
- C. X can see H
- D. X can see C and F
- E. none of the above

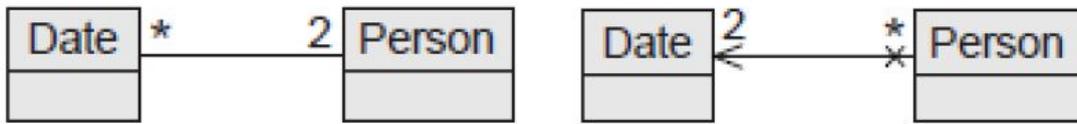
2) Consider now that we add an import relation where P imports W. Which of the following statements is true?
(choose one)

- A. P can see F
- B. P can see C and G
- C. P can see A and B
- D. O can see C and E
- E. O can see F and H and G

II - Design Class Diagrams

1 For each of the following, please select the most appropriate answer.

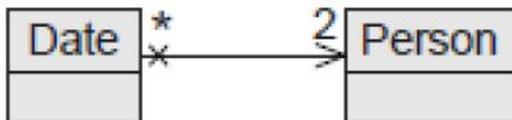
1.1 On a blind date exactly two people take part. A person can take part in multiple blind dates. It is not sufficient to know a person to derive all his/her dates.



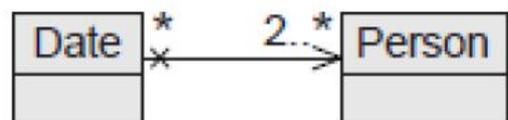
(a)



(b)

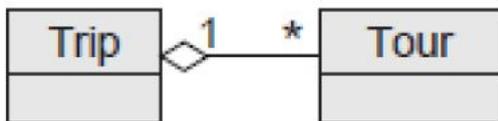


(c)

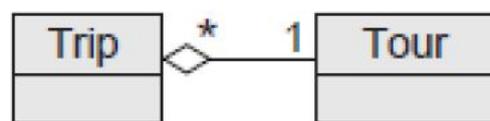


(d)

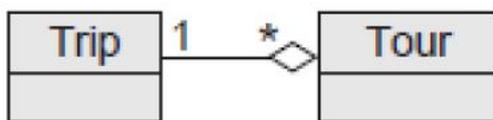
1.2 A trip might comprise of multiple tours, one tour can be included in several trips.



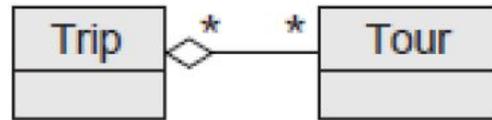
(a)



(b)

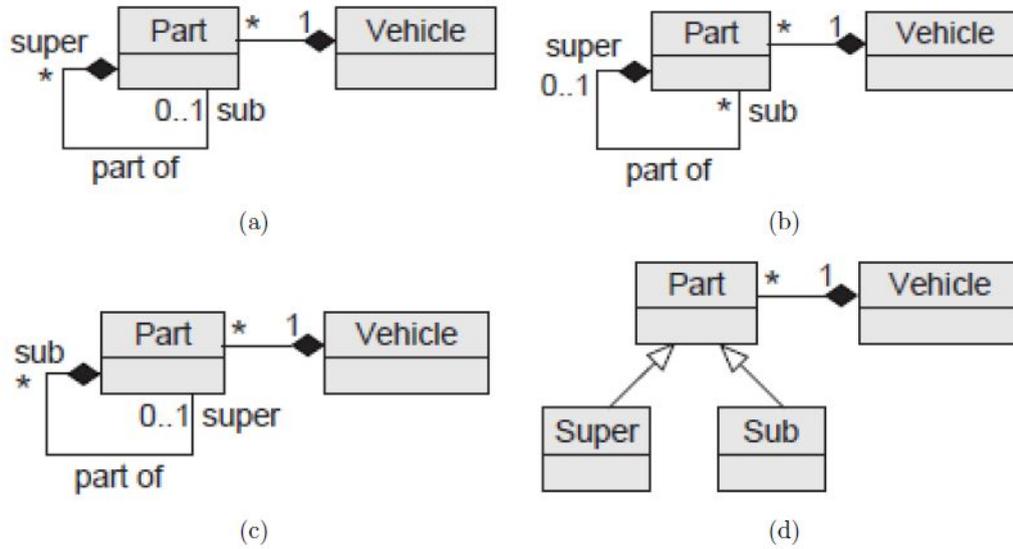


(c)

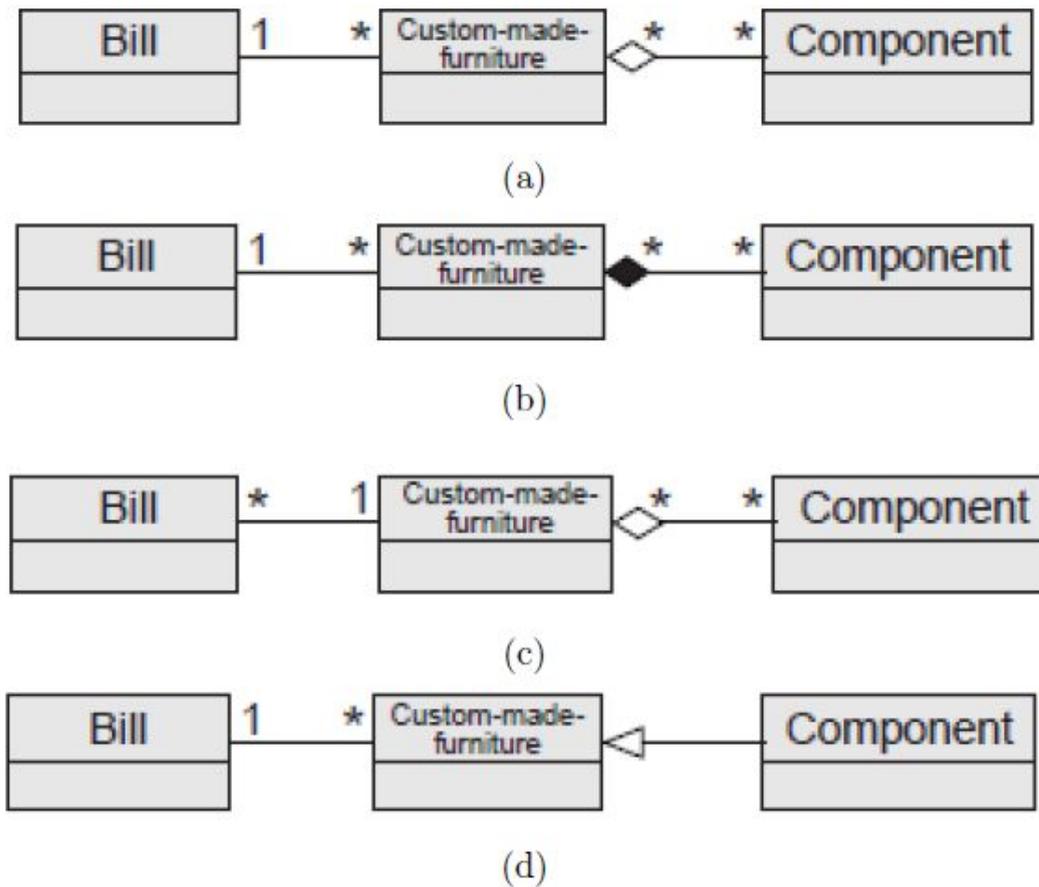


(d)

1.3 A vehicle comprises of multiple parts. Each part can be built in a maximum of one super-part. One super-part comprises of multiple sub-parts.



1.4 FurnitureFritz charges custom-made furniture which is assembled from prefabricated components.



2. Consider the following requirements description

An air traffic information system stores information about flight details, so that customers can keep track of flights. Each flight is operated by one or more airline companies (e.g. TAP and Air France may cooperate a flight from Lisbon to Paris). Each company operates an arbitrary number of flights. For each flight, the system needs to store the flight number, origin airport, destination airport, departure time, terminal and gate, landing time, terminal and gate. A trip has at least one flight, while a flight is a part of at least one trip. Each trip is made by exactly one passenger, but a passenger may make one or more trips. Flights are performed by an aircraft, of which we need to know the manufacturer, model and capacity for carrying passengers (i.e., how many passengers it can carry). Some aircrafts can only carry cargo. Others, can transport passengers. Of these, some are purely passenger carriers, while others are mixed, so they can carry passengers and other cargo. Of course, all these kinds of aircrafts have to fly, but only those with passenger capacity can be used for trips. Purely cargo flights are only owned by cargo aircrafts, but may not be used in trips.

2.1 Identify a list of potential domain classes

Start by identifying candidate classes. List them. Then, check if you may have synonyms, so that you end up with a list without repetitions.

2.2 Create an analysis class diagram for this description

In the analysis class diagram, make sure you identify domain classes, their associations, cardinalities and roles. You should also identify the main attributes of your classes (and association classes, if any), as described here. You do not need to add extra attributes you may be aware of, from your own domain knowledge, if they are not mentioned in the requirements, right now.

2.3 Rene your analysis class diagram into a design class diagram

Using the patterns discussed in the lectures, please derive a design class diagram. You may need to identify extra classes, merge classes that were identified as separate during analysis, and so on. At this stage, make sure you assign a type to your attributes, reify relationships, clarify navigability, visibility, etc. Assume you will be outsourcing the implementation to programmers with great development skills (e.g. assume they will know how to choose a decent collection type, should that need arise). Note: at the end of the day, your design class diagram has to be traceable to the analysis class diagram.

2.4 Generate the skeleton for your classes in Java.

As it turns out, your fantastic development team is not available right now and, instead, you have an inexperienced team of developers. To make sure they are able to implement this system properly, generate the skeleton for your classes in Java. Here, you will need to select adequate Java classes for your implementation. Note: at the end of the day, your code skeleton has to be traceable to the design class diagram.