

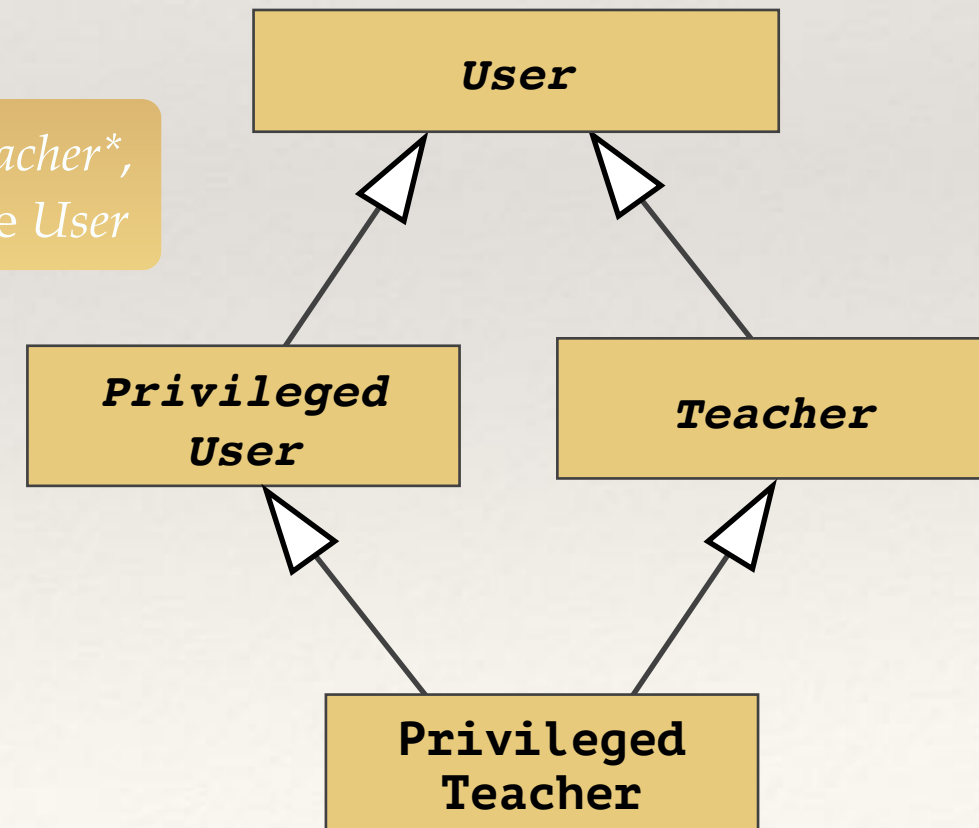
Hijerarhijska dekompozicija

- ❖ Međutim, kako C++ *uvek* implementira objekte izvedene klase tako da u njih ugrađuje po jedan objekat svake osnovne klase (i tako rekurzivno), sledi:
 - objekat klase *PrivilegedTeacher* u sebi ima podobjekat klase *PrivilegedUser*, a ovaj u sebi podobjekat klase *User*
 - objekat klase *PrivilegedTeacher* u sebi ima podobjekat klase *Teacher*, a ovaj u sebi podobjekat klase *User*
- ❖ Tako će objekat klase *PrivilegedTeacher* u sebi imati dva podobjekta osnovne klase *User*, što znači dva kompleta svih svojstava, što nije poželjno. Ova pojava naziva se “problem dijamanta” (*diamond problem*), zbog grafa nasleđivanja oblika romba (liči na dijamant i često se tako naziva na engleskom)
- ❖ Ovim podobjektima može se pristupiti navođenjem kvalifikovanog imena (staze) i operatora ::

```
this->PrivilegedUser::username = ...;
```

```
this->Teacher::username = ...;
```

this je tipa PrivilegedTeacher, a username je član klase User*



Hijerarhijska dekompozicija

- ❖ Da bi se rešio ovaj problem, osnovna klasa *User* treba da se deklarise kao tzv. *virtuelna osnovna klasa*:

```
class User {...};  
class PrivilegedUser : virtual public User {...};  
class NonprivilegedUser : virtual public User {...};  
  
class Teacher : virtual public User {...};  
class Student : virtual public User {...};  
  
class PrivilegedTeacher : public PrivilegedUser, public Teacher {...};
```

- ❖ Značenje virtuelne osnovne klase je sledeće: ako se objekat neke klase (*PrivilegedUser*, *Teacher*, ...) nađe kao podobjekat unutar objekta neke izvedene klase (*PrivilegedTeacher*), u njega ugrađen podobjekat virtuelne osnovne klase (*User*) će biti zajednički (deljen) za sve druge podobjekte unutar tog objekta koji takođe imaju tu osnovnu klasu kao virtuelnu
- ❖ Ovo se implementira na sledeći način: unutar objekta izvedene klase postoji pokazivač na podobjekat virtuelne osnovne klase, jer on ne mora uvek biti na unapred poznatom mestu unutar tog objekta (na njegovom početku), pošto se može nalaziti unutar nekog drugog objekta sa kojim se deli
- ❖ Zbog toga prevodilac ne može generisati kod za pristup članovima virtuelne osnovne klase statičkim vezivanjem (adresiranjem preko adresa, tj. pomeraja u odnosu na adresu početka objekta poznatim u vreme prevođenja), već dinamičkim vezivanjem, preko ovog pokazivača

