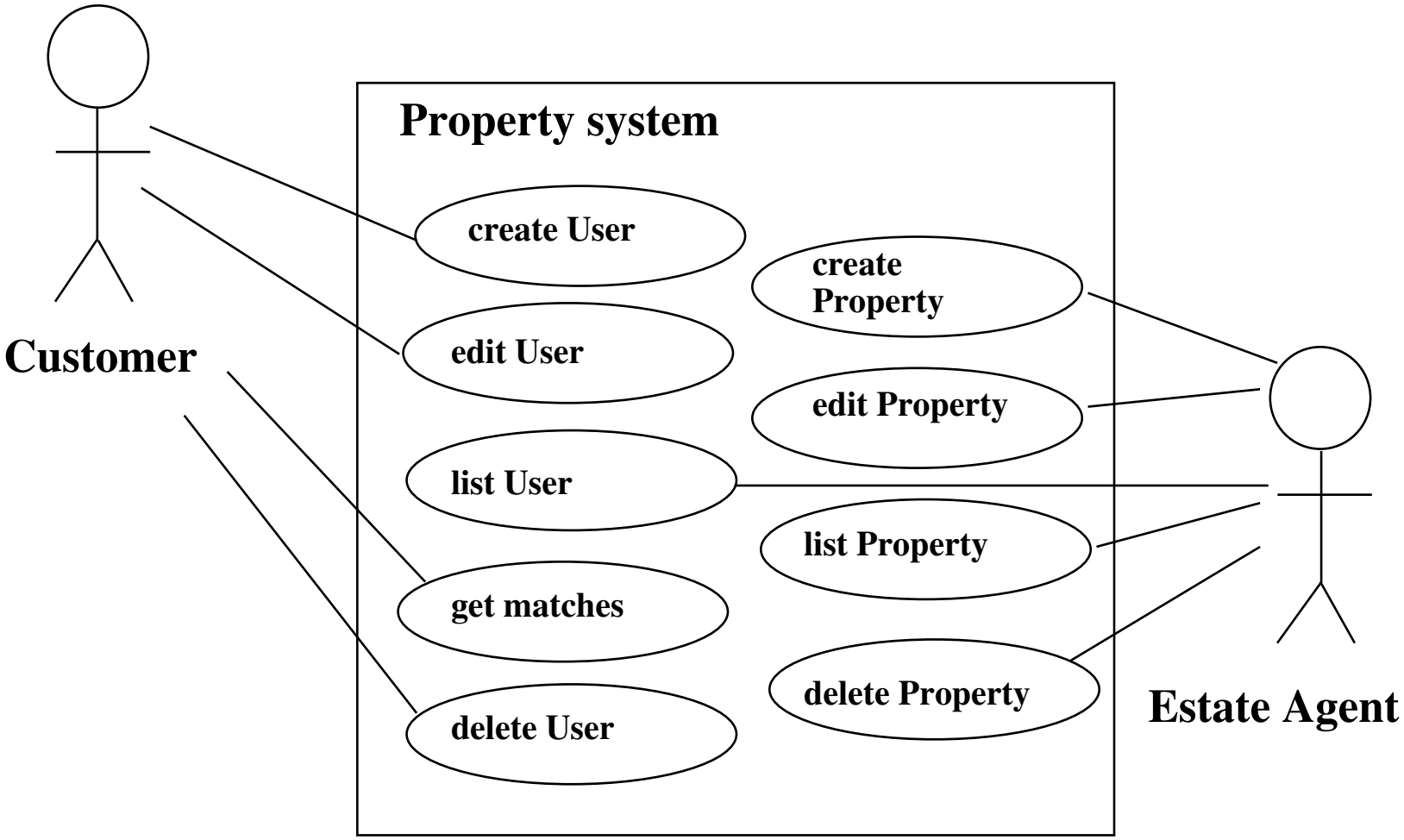
Property System

This is a simple online property search system for an estate agent: users may register their requirements for a property, and then carry out searches for properties that match these requirements.

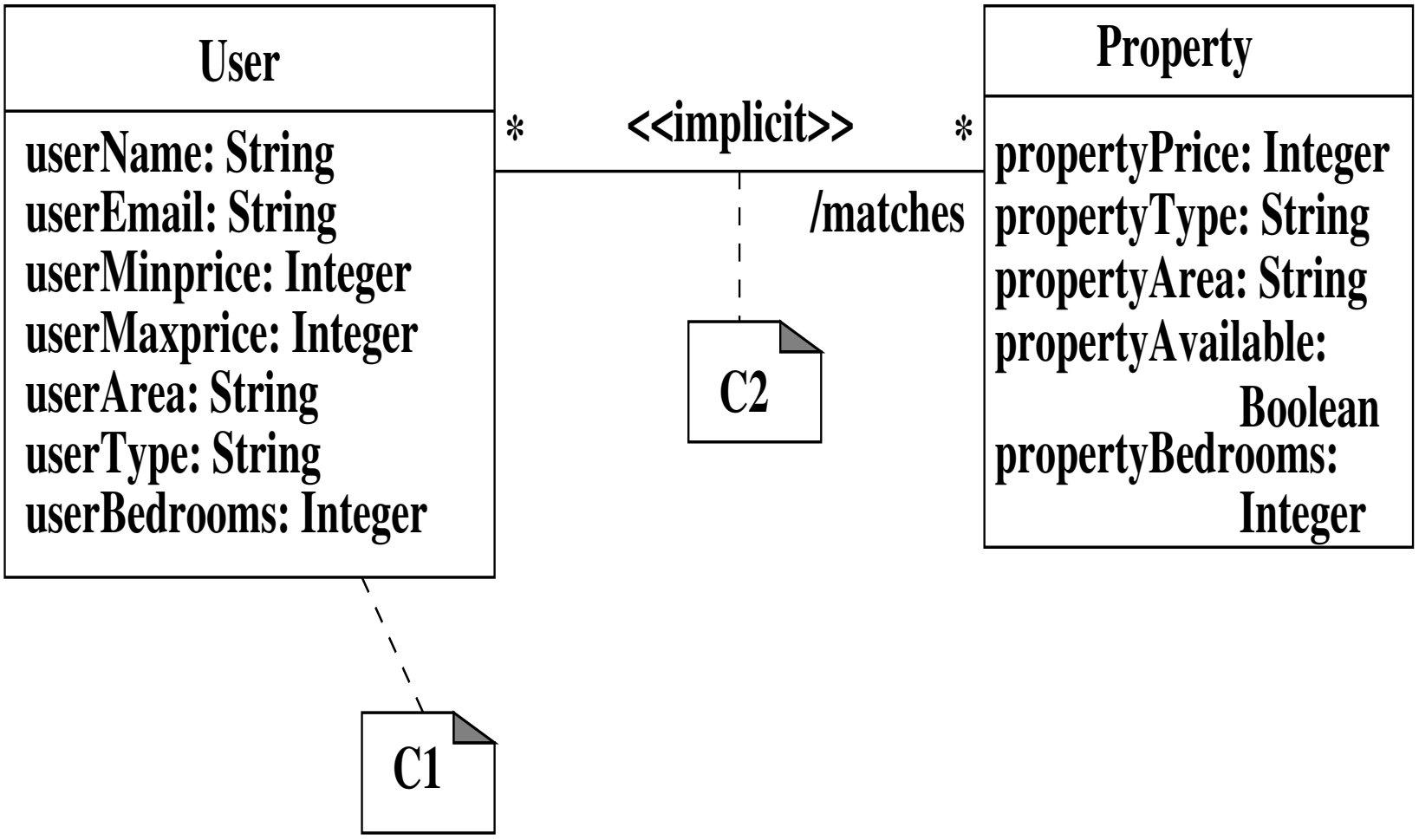
In property PIM $C1$ is an example of a class invariant constraint, of *User*:

$$userName.size > 0 \ \&$$
$$userMinprice \leq userMaxprice$$

This can be used to define data validation checks in the entity bean of *User*: for checking data which is input to system for creation of new instances of the class, or for modification of instances of the class.



Use cases of property system

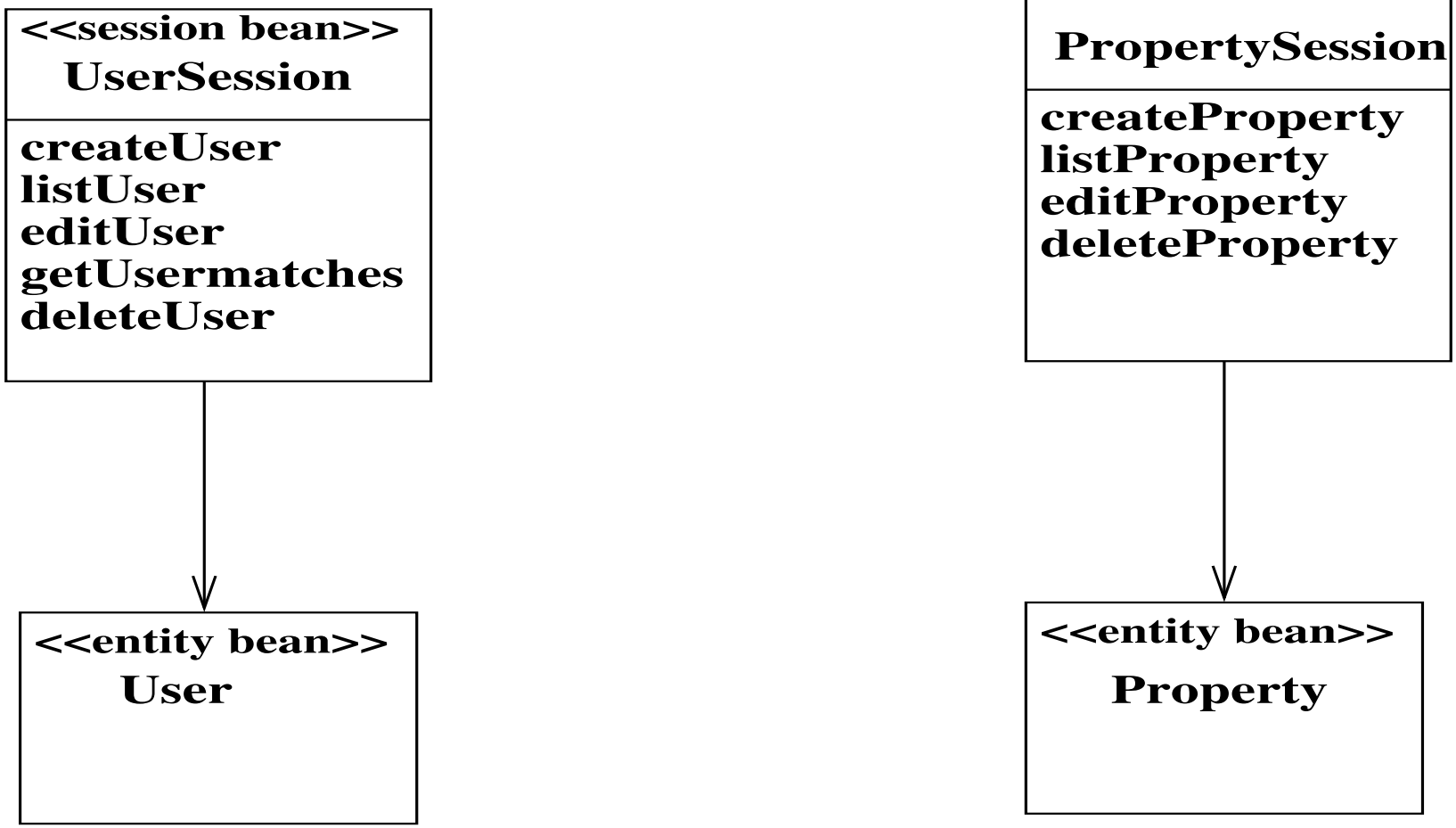


PIM of property system

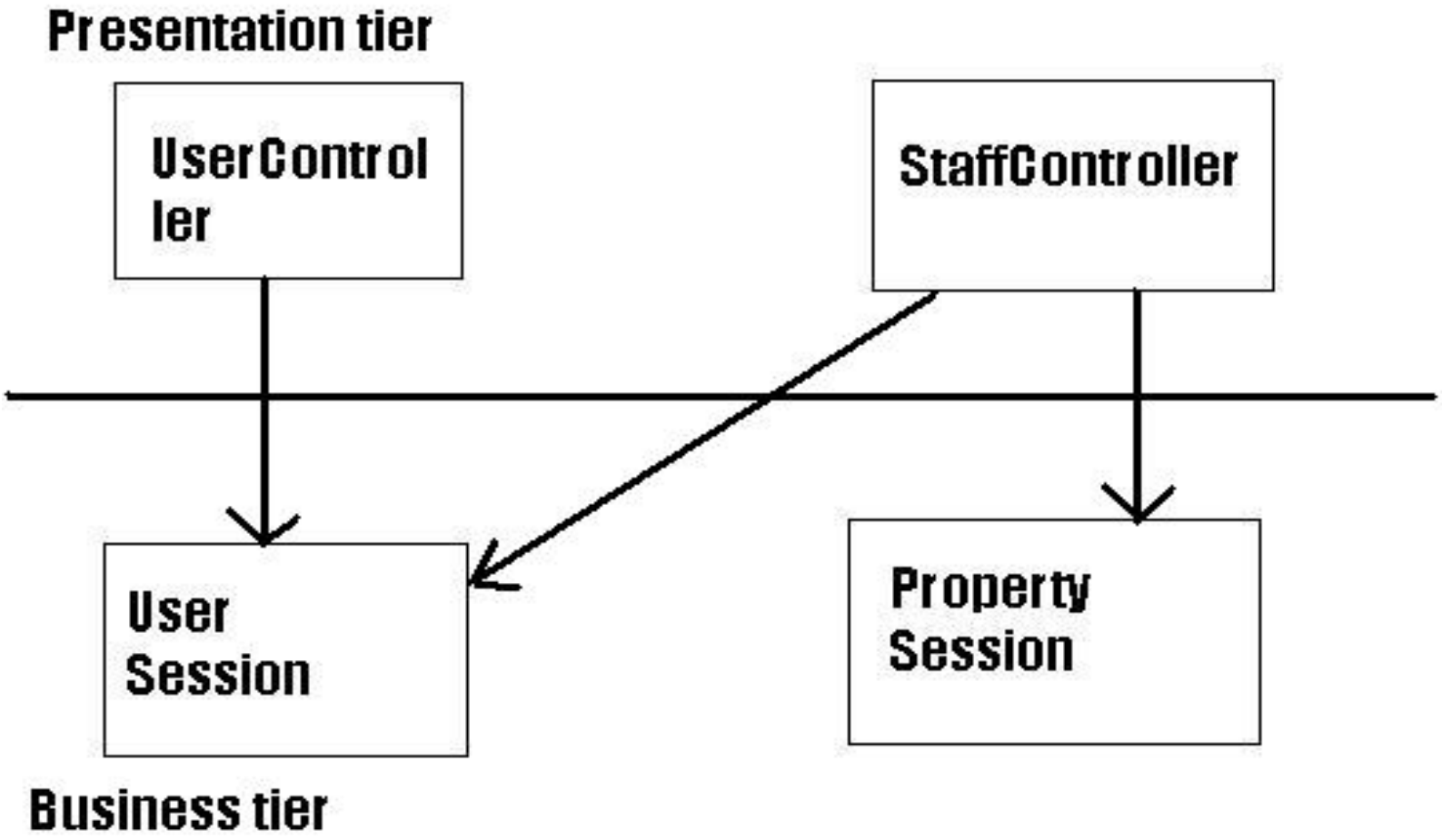
In contrast, constraint $C2$ attached to association defines set of elements that are linked by the association:

$$
\begin{aligned}
&userArea = propertyArea \ \& \\
&propertyPrice \leq userMaxprice \ \& \\
&userMinprice \leq propertyPrice \ \& \\
&userBedrooms \leq propertyBedrooms \ \& \\
&userType = propertyType \ \& \\
&propertyAvailable = true
\end{aligned}
$$

This is used to derive the code of *getUsermatches* operation to find all properties which meet a particular user's requirements.



Business tier architecture of property system (1st version)

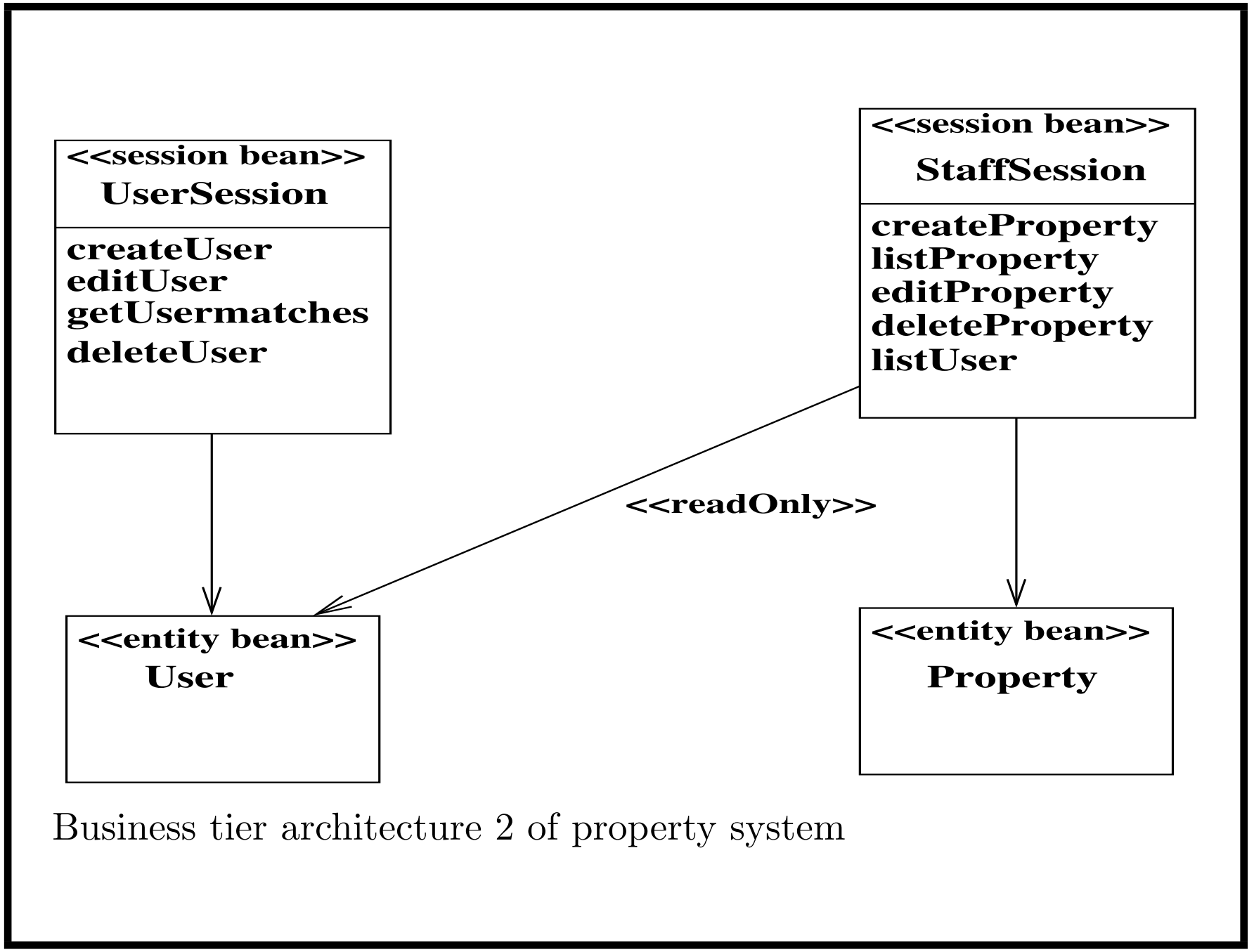


Presentation tier/Business tier architecture (1st version)

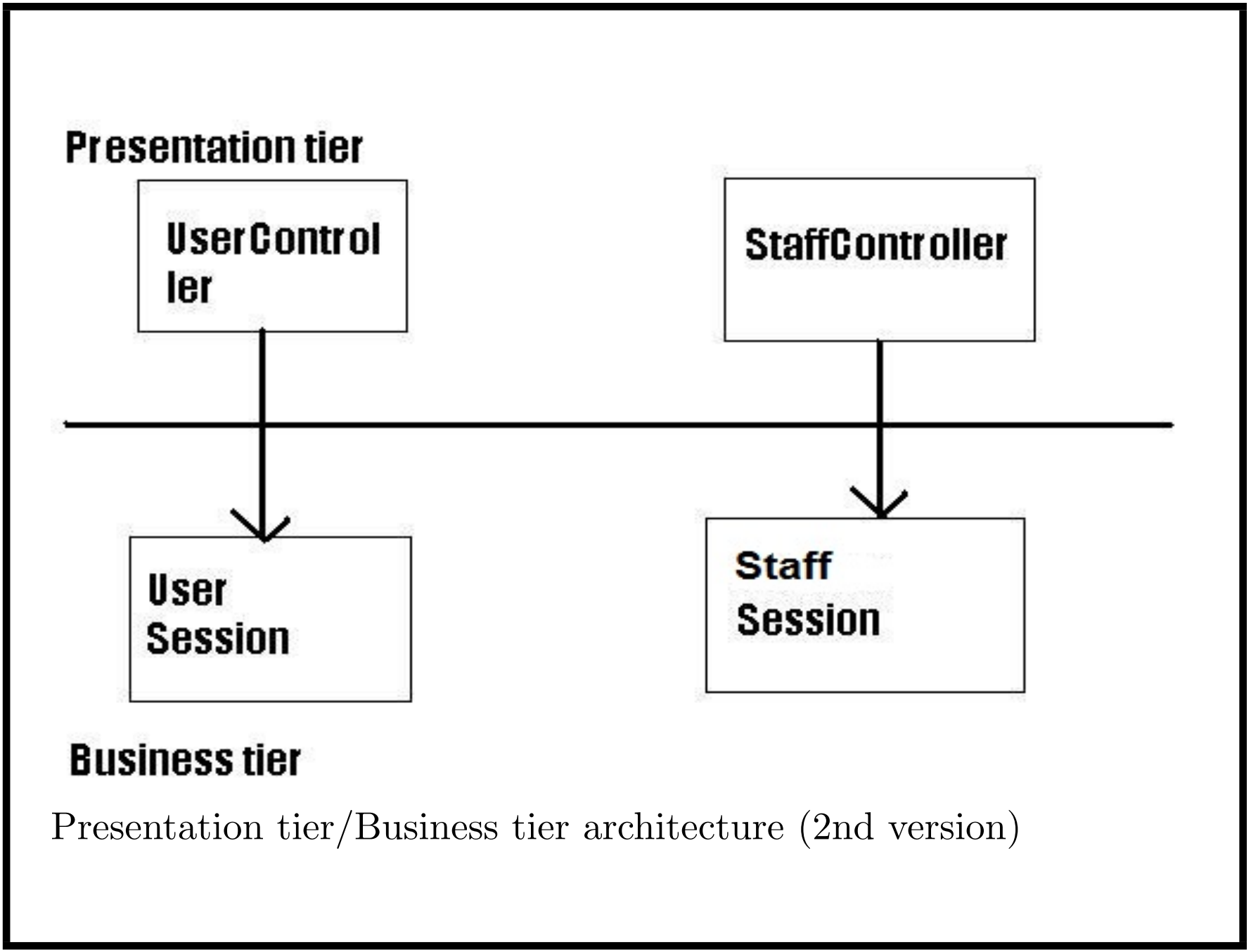
Separate session beans can be used, since none of use cases operating on *User* involve updating *Property*, and use cases on *Property* do not involve *User*. There are no constraints connecting the two classes.

In this case we have grouped use cases into beans on basis of what entity they operate on. An alternative would be to group them according to the actor of the use case: this would place *listUser* in the *StaffSession* session bean, and require read-only access from this to the *User* entity bean.

This has benefit that a single session bean can be used by each interface (actor) of the system, although it slightly increases number of dependencies in business tier. Also, users should not have access to *listUser*.



Business tier architecture 2 of property system



Presentation tier/Business tier architecture (2nd version)