
CyberResidencesOCL Documentation

Release 3.1.0

megaplanet

October 03, 2016

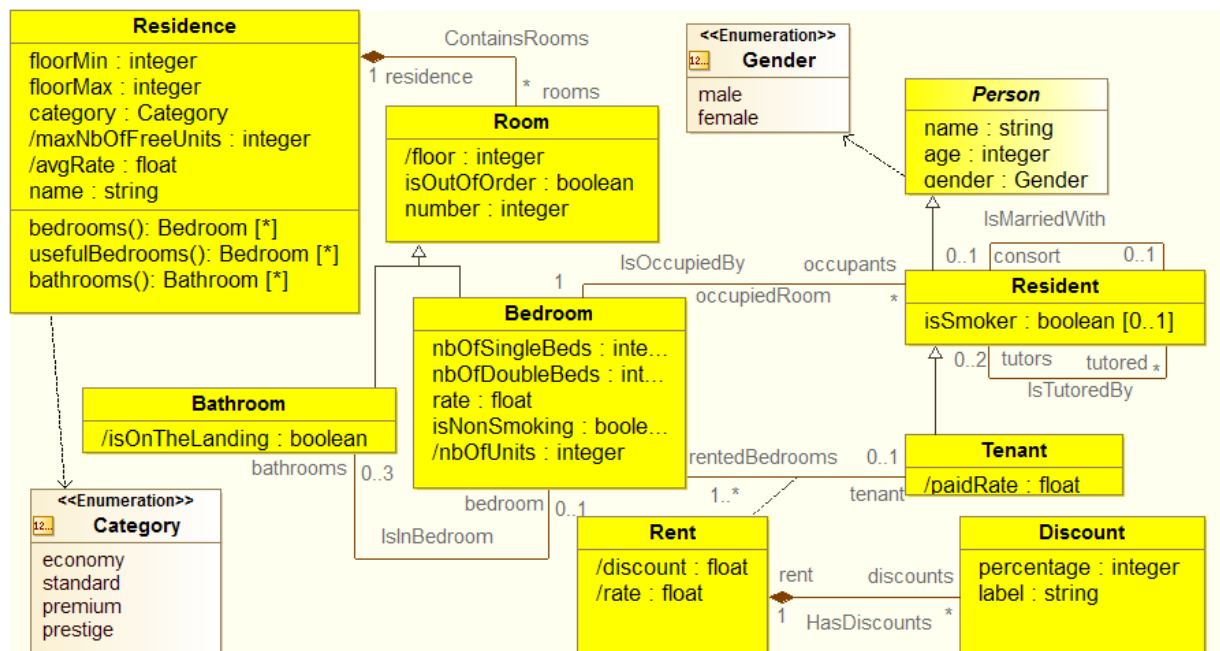
1	Overview	3
2	“Buildings” package	5
3	“Rents” package	7
4	“Rates” package	9
5	Constraints	11
6	USE OCL Specification	13
7	Questions?	15

This case study provides a UML model dealing with long-stay “*Residences*” such as university residences or so. The *model* represented by various class *diagrams*. Each diagram is providing information about the CyberResidence *system* and according to a different perspective.

Attention: If after reading this document you have some **questions**, or if you find some **errors** please do not hesitate to [post an issue](#). You should also have a look there to see if some questions are already answered.

Overview

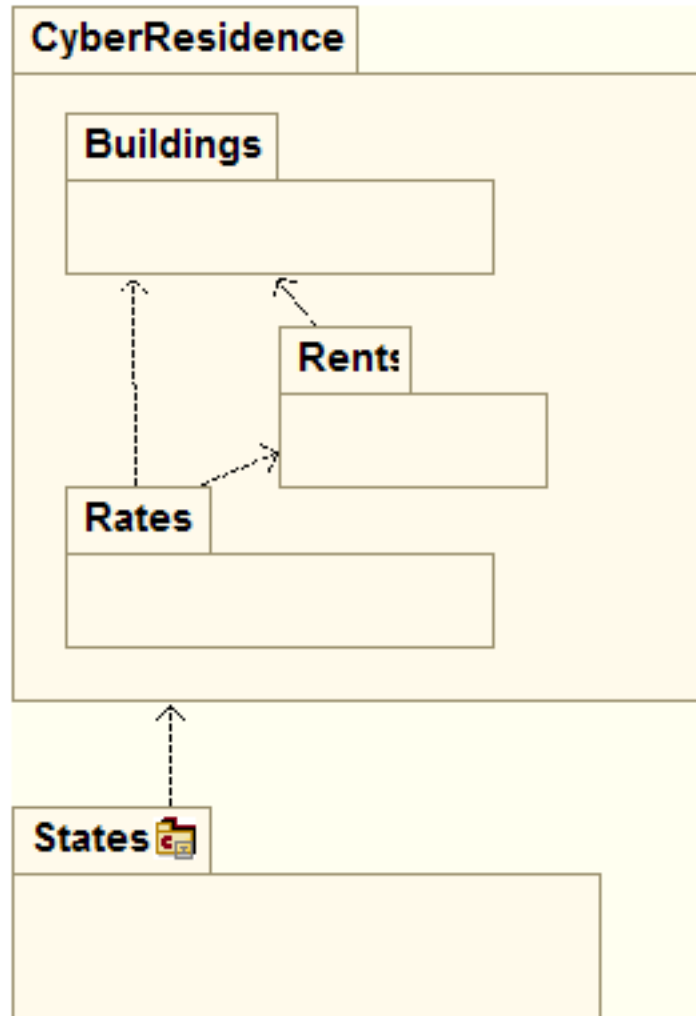
The diagram below presents CyberResidences from a global perspective. This diagram does not show however the constraints. These constraints, or at least their location, are presented below in the context of each package.



In fact, the UML model has been split logically in three different packages: * Buildings package * Rents package * Rates package

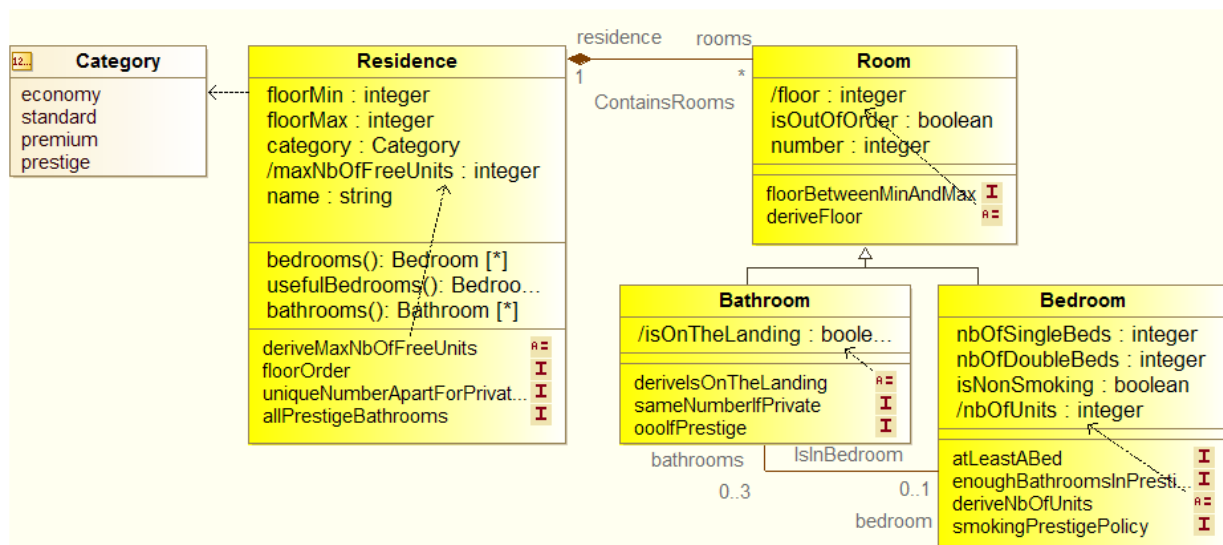
The diagram below shows the decomposition of the system in subsystems. Additionally the package 'States' contains examples of states for illustrating and or testing CyberResidences.

Each package is described below separately.



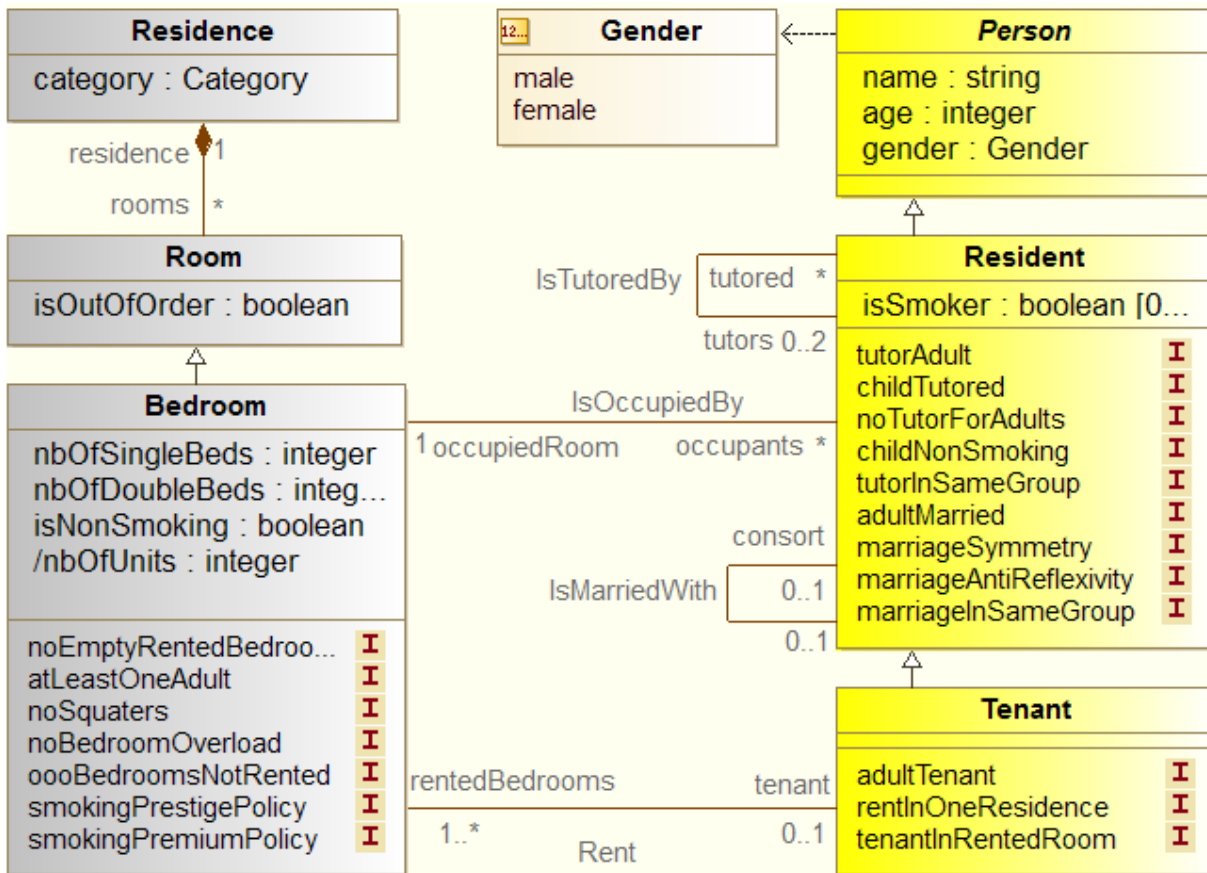
“Buildings” package

The “Buildings” package describes residences from a rather physical perspective. Residences are made of rooms that are either bedrooms or bathrooms. This diagram shows not only these concepts but also lists the constraints associated with each class.



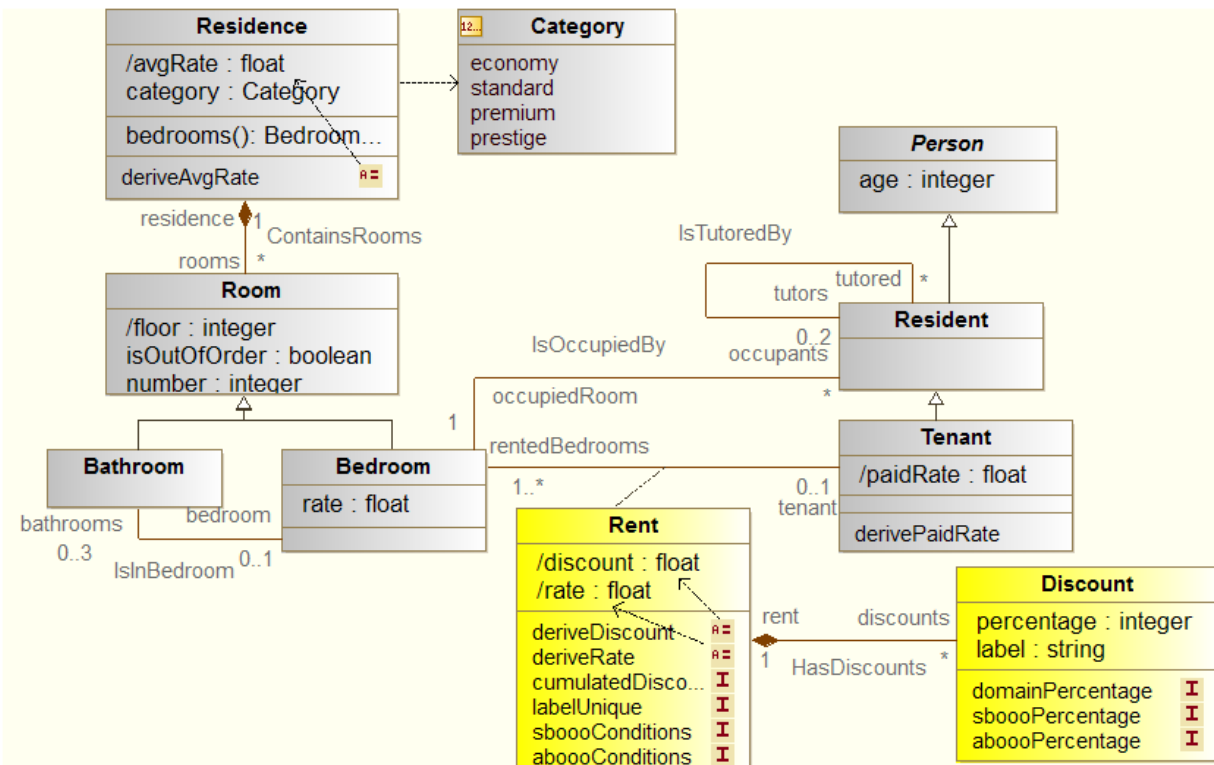
“Rents” package

The “Rents” package is basically answering “who is in the residence”. A tenant pays for one or more rooms, but he or she may be responsible of a much larger group of residents. These residents all occupy a room rented from that tenant. This diagram shows not only these concepts but also lists the constraints associated with each class. The “Rents” package heavily depends on the [Buildings](#).



“Rates” package

The “Rates” package is basically answering “how much are rents?”. It is all about money, rates and discounts. The rate for a rent depends on various factors but in particular on the category of the residence. This package depends on the [Rents](#) and on the [Buildings](#).



Constraints

The constraints on the model are written in natural language. Each constraint has a given name. There are about 40 constraints in the model.

Table 5.1: Examples of constraints

Name	Description
adultMarried	To be registered as married, resident must be adult.
childTutored	Resident that are less than 18 years old must have at least one tutor, apart in prestige residences where this rule does not apply.

The goal is to convert to express these constraints in a an operation model, namely using the OCL language.

USE OCL Specification

The [USE OCL](#) tool allows to express a full specification. Such a textual specification contains in a same file:

- **the structural specification.** This is the equivalent of class diagrams but expressed using a simple textual notation. Instead of writing *A* in a box as in a UML class diagram, in USE OCL we write `class A`. You can see below the class `Residence` translated in USE OCL:

```
class Residence
attributes
  name      : String
  floorMin  : Integer
  floorMax  : Integer
  category  : Category
  maxNbOfFreeUnits : Integer -- @derived
  avgRate   : Real           -- @derived
end
```

- **the constraint specifications.** All the constraints can then be written in a dialect of standard OCL. Only a few keywords change with respect to OCL.

Note: The details of the USE OCL language can be found in the [USE OCL documentation](#) and in the particular in the reference manual.

Questions?

Attention: If you have some **questions**, or if you find some **errors** please do not hesitate to [post an issue](#). You should also have a look there to see if some questions are already answered.