# Inheritance

In this post we will look at how different types of inheritance can be translated to OWL. We consider the case where <code>Person</code> is specialized by <code>Employee</code> and <code>Client</code>. In a UML class diagram if inheritance is not annotated the default annotation <code>{incomplete, disjoint}</code> is assumed. <code>incomplete</code> means there are instances of <code>Person</code> which are neither of type <code>Employee</code> nor <code>Client</code>. <code>disjoint</code> means there is no instance of <code>Person</code> that is both of type <code>Employee</code> and of type <code>Client</code>.

The annotation <code>{complete, disjoint}</code> means every instance of <code>Person</code> is either a instance of <code>Employee</code> or an instance of <code>Client</code>.

When <code>overlapping</code> is used rather than <code>disjoint</code> it mean an instance of <code>Person</code> may be of both of type <code>Employee</code> and of type <code>Client</code>.