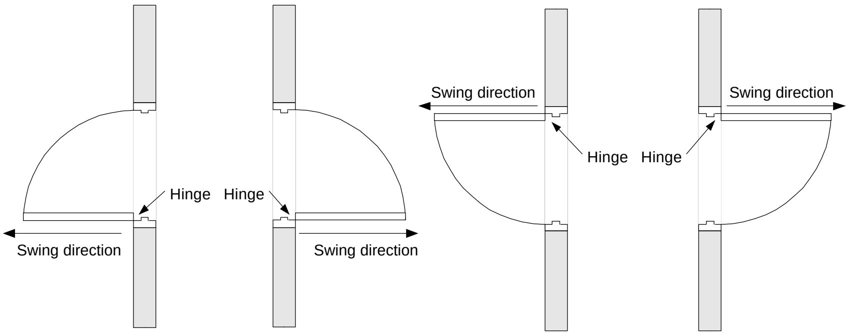
Door swing directions in IFC R2.0

DRAFT 2 – 20.12.1999

Modelling the opening direction of a door is basically an easy task, problems arise because it can be done in so many different ways. This paper tries to make it easier to agree on one single method for IFC R2.0 implementations.

Table of contents

General concept



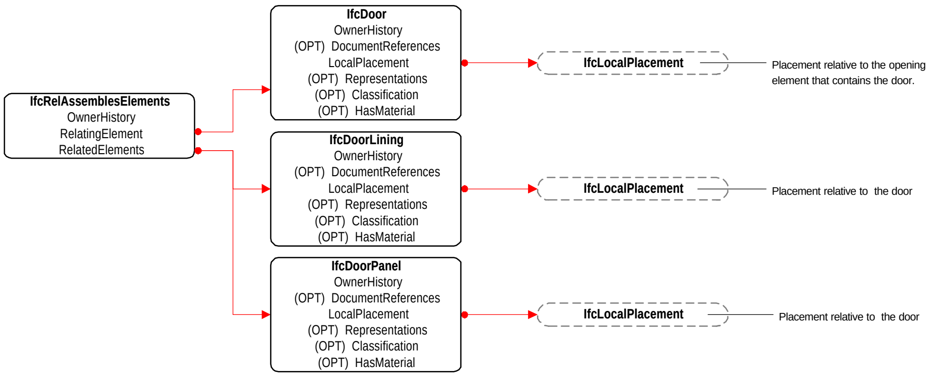
The question of the swing direction boils down to two simple questions.

1. How do we define what is left and what is right?
2. How do we define the location of the hinge?

In IFC R1.5.1 a door was not divided into components, a door was an instance of IfcDoor and PropertySets attached to the IfcDoor instance. The R1.5.1 implementers have not documented how exactly the question of the door swing direction is solved, but the general idea is clear. There are two properties that define the swing direction, one tells if the swing is to the right or to the left, and the other tells on which side of the opening the hinge can be found.

In IFC R2.0 the door was broken up into components, the IfcDoorPanel and the IfcDoorLining. The IfcDoor does no longer have a type, i.e. single or double swing; this information can be found in IfcDoorPanel or in the combination of several IfcDoorPanel instances that make up the door. A double swing door is a door that consist of one IfcDoor instance that references two IfcDoorPanel instances of type Swinging. Consequently the information about the swing direction was moved to IfcDoorPanel (Pset_DoorPanelSwinging.LeftNotRightSwing). The attribute to define the location of the hinge was removed.

Author: Jiri Hietanen (jiri.Hietanen@qpartner.com)

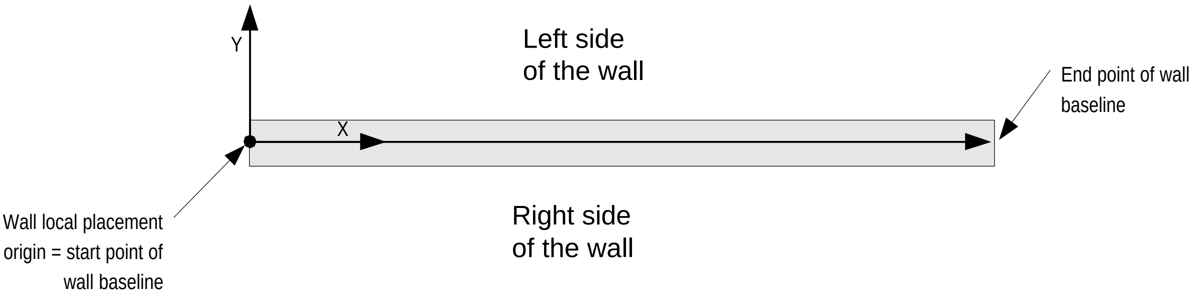


What is left and what is right?

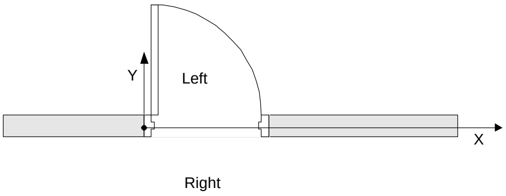
The first question is how to determine what is the left side of the door and what is the right side of the door. There are basically three solutions

1. Let IfcWall decide
2. Let IfcDoor decide
3. Let IfcDoorPanel decide

Because a door is placed relative to a wall in IFCs it could be natural that the wall decides what is left and what is right. Agreeing on this might be helpful also in other cases where the question is about the left and right side of the wall itself. The drawback is that it would not be possible to model a door that exists independent from a wall.



The second solution would be to use the local placement of IfcDoor itself. The IfcOpeningElement containing IfcDoor is placed relative to IfcWall, and IfcDoor is placed relative to IfcOpeningElement. The direction of the local coordinate system of IfcDoor would decide what is left and what is right.



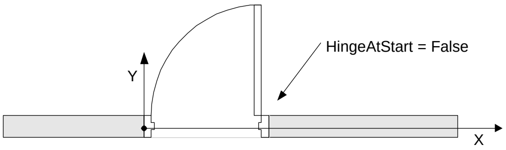
If the door was always modelled down to the component level the decision about left and right could be left to the local coordinate system of IfcDoorPanel

Where is the hinge?

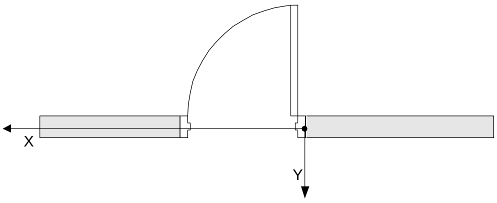
For the location of the hinge there are basically two solutions

1. An attribute telling the position of the hinge
2. The hinge is always at the local coordinate system origin of IfcDoor or IfcDoorPanel.

In the first solution the local coordinate system of IfcDoor or IfcDoorPanel is placed freely on either side of the door, and the HingeAtStart attribute tells if the hinge is on that side or on the opposite side.



The second solution is to always place the local coordinate system origin on the side of the hinge and omit the HingeAtStart attribute.



Considerations

With the considerations above we can make the following table

Left and right	Hinge	Possible use
Wall decides	Attribute decides	Can be used to reach easier mapping with R1.5.1, if R1.5.1 uses wall to determine left and right.
Door decides	Attribute decides	Can be used to reach easier mapping with R1.5.1, if R1.5.1 uses door to determine left and right.
Door panel decides	Attribute decides	Door panel has to be modelled, but uses R1.5.1 hinge concept. This combination is kind of a mismatch between R1.5.1 and R2.0
Wall decides	Local placement decides	Letting the wall decide about left and right is a simplified case. If this is used also the simple attribute for the hinge should be used. I don’t think this is a good combination.
Door decides	Local placement decides	This combination could be used for simple doors that don’t require the definition of their components.
Door panel decides	Local placement decides	This combination seems to match the R2.0 intention best. However, it requires that the IfcDoorPanel be modelled.

Proposal

Swing direction

Since a door can be modelled using a simple IfcDoor or using the door components IfcDoorPanel and IfcDoorLining, I propose that the object deciding what is left and what is right depends on the way the door was modelled.

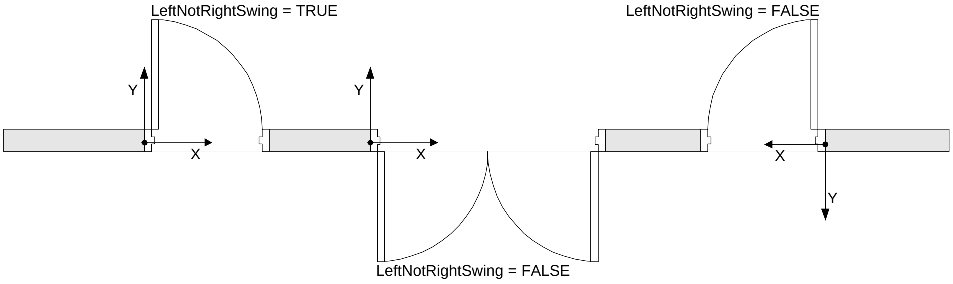
- If IfcDoorPanel is modelled IfcDoorPanel decides about left and right

- If IfcDoorPanel is not modelled IfcDoor decides about left and right

Hinge location

Since the HingeAtStart property was removed in R2.0, and since defining the location of the hinge through the local placement is not complicated, I propose that the origin of the local coordinate system of IfcDoor or IfcDoorPanel is used to determine the location of the hinge.

- If IfcDoorPanel is modelled the hinge is at the origin of the local placement of IfcDoorPanel
- If IfcDoorPanel is not modelled the hinge is at the origin of the local placement of IfcDoor


LeftNotRightSwing = TRUE
LeftNotRightSwing = FALSE
Y
Y
X
X
X
Y
LeftNotRightSwing = FALSE