# LandXML Station equation in IFC Alignment

* **According to the LandXML specification at Trafikverket**

## Background and purpose

The Swedish Transport Administration (STA) uses LandXML today to exchange data regarding rail geometry. The scope of this exchange is:

* Horizontal alignment
* Vertical alignment
* Cant
* Station equations (broken chainage)
* Coordinates for significant points for switches

STA wishes to gradually move from LandXML to IFC and in the process widen the scope beyond the above listed data types.

This document elaborates on station equations and how that may be represented using IFC.

## Example case (Trafikverket)



Figure 1 - STA broken chainage example

Figure 1 above is used as an example.

* There are referents for each kilometer starting with KM32+
* The alignment starts 164.197 meters from KM32+ in positive direction
* The actual distance between each km referent varies and is never exactly 1 km.

## Representation in LandXML according to STA specification

The above case is represented in LandXML according to the XML listed below:

<StaEquation staInternal="32164.197" staBack="32164.197" staAhead="164.19700000000012">

<Feature code="StaEquation">

<Property value="" label="bckEqn"/>

<Property value="KM32+" label="ahdEqn"/>

</Feature>

</StaEquation>

<StaEquation staInternal="32999.858" staBack="999.858" staAhead="0.0">

<Feature code="StaEquation">

<Property value="KM32+" label="bckEqn"/>

<Property value="KM33+" label="ahdEqn"/>

</Feature>

</StaEquation>

<StaEquation staInternal="34000.499" staBack="1000.641" staAhead="0.0">

<Feature code="StaEquation">

<Property value="KM33+" label="bckEqn"/>

<Property value="KM34+" label="ahdEqn"/>

</Feature>

</StaEquation>

<StaEquation staInternal="35000.139" staBack="999.64" staAhead="0.0">

<Feature code="StaEquation">

<Property value="KM34+" label="bckEqn"/>

<Property value="KM35+" label="ahdEqn"/>

</Feature>

</StaEquation>

## Possible representation in IFC

One possible IFC representation of the example above, including KM32+ which is actually not placed along the alignment being exchanged, is illustrated below:



Figure 2 - IFC representation

The assumption is that IfcReferent should be used to represent each km marker and that linear placement vs alignments may be used to indicate the actual distances.

Below is a simplified IFC representation of the example (no STEP file at this stage).

The alignment may use StartDistAlong to indicate where the alignment starts with regards to the “global” km system.

**Alignment: IfcAlignment**

Alignment.Horizontal.StartDistAlong = 32164.197

For each km marker, an IfcReferent instance may be used to indicate the marker value (the km) and its cartesian and/or linear location (vs an alignment).

The first referent in the example occurs before the actual alignment starts. f this referent shall be represented in the same manner as in the LandXML example (with the first StaEquation element and “staAhead” value), then one option is to set a “virtual” negative distance. Another option would be to use the ObjectPlacement attribute for IfcAlignment and reference the LinearPlacement of the KM32+ referent.

The rest of the referents may be defined using linear placement vs the alignment geometry as follows.

**Referent3 : IfcReferent**

Referent3.Name = “KM34+”

Referent3.ObjectPlacement.PlacementMeasuredAlong = Alignment.Axis

Referent3.ObjectPlacement.Distance.DistanceAlong = 1836,302 (835,661+1000,641)

Referent3.ObjectPlacement.Distance.AlongHorizontal = True

Referent3.ObjectPlacement.CartesianPosition = X3,Y3

Referent2.ObjectPlacement.PredefinedType = KILOPOINT

Referent3.RestartDistance = 34000 (?)

**Referent2 : IfcReferent**

Referent2.Name = “KM33+”

Referent2.ObjectPlacement.PlacementMeasuredAlong = Alignment.Axis

Referent2.ObjectPlacement.Distance.DistanceAlong = 835,661 (999,858-164,197)

Referent2.ObjectPlacement.Distance.AlongHorizontal = True

Referent2.ObjectPlacement.CartesianPosition = X2,Y2

Referent2.ObjectPlacement.PredefinedType = KILOPOINT

Referent2.RestartDistance = 33000 (?)

**Referent1 : IfcReferent**

Referent1.Name = “KM32+”

Referent1.ObjectPlacement.PlacementMeasuredAlong = Alignment.Axis

Referent1.ObjectPlacement.Distance.DistanceAlong = -164.197

Referent1.ObjectPlacement.Distance.AlongHorizontal = True

Referent1.ObjectPlacement.CartesianPosition = X1,Y1

Referent1.ObjectPlacement.PredefinedType = KILOPOINT

Referent1.RestartDistance = 32000 (?)

**Referent4 : IfcReferent**

Referent4.Name = “KM35+”

Referent4.ObjectPlacement.PlacementMeasuredAlong = Alignment.Axis

Referent4.ObjectPlacement.Distance.DistanceAlong = 2835,906 (1836,302+999,604)

Referent4.ObjectPlacement.Distance.AlongHorizontal = True

Referent4.ObjectPlacement.CartesianPosition = X4,Y4

Referent2.ObjectPlacement.PredefinedType = KILOPOINT

Referent4.RestartDistance = 35000 (?)