
Inframodel 4.2.0

Finnish Inframodel application documentation for
LandXML version 1.2

RYTV 22003_1 working group

<https://github.com/buildingSMART-Finland/InfraModel:4.2.0> sha: eb2720b8 at: 2025-01-23

07:24:04



Contents

1	File headers	6
1.1	XML file header	6
1.2	LandXML container	6
1.3	Units	8
1.4	Coordinate and height systems	9
1.4.1	Local coordinate transformation by point pairs	10
1.4.2	Local coordinate transformation by transformation parameters	10
1.5	Project	15
1.6	Type coding systems	16
1.7	Application	17
1.8	Authors	17
1.9	Feature dictionary	18
1.10	Metadata	18
2	Base data	23
2.1	Plan information	24
2.1.1	Current and planned surfaces	25
2.1.2	Type coding	25
2.1.3	Quantity information	26
2.2	Source data	27
2.2.1	Data points	28
2.2.2	Breaklines	29
2.2.3	Boundaries	30
2.3	Triangular mesh surface	34
2.3.1	Vertices	35
2.3.2	Faces	36
2.4	Terrain model	37
2.4.1	Soil properties	38
2.5	Ground layer model	38
3	Route planning	40
3.1	Route description	40
3.2	Naming and Type coding	41
3.3	Composing alignments	41
3.3.1	Plan information	43
3.4	Geometric alignments	43
3.4.1	Horizontal geometry	44
3.4.2	Vertical geometry	48
3.5	Line strings	51
3.5.1	Line string	51
3.6	String line model	52
3.7	Terrain model	55
3.8	Structural model	55

3.9	Cross-section parameters	56
4	Road and street design	57
4.1	Geometry of roads and streets	57
4.2	String line model of roads and streets	58
4.3	Cross-section parameters of roads and streets	59
4.3.1	Cross-section parameters	60
4.3.2	Transitions in superelevation	62
4.4	Terrain model and structural model of road or street	62
4.4.1	Structural layers	64
4.5	Road signs and plan features	64
4.5.1	Plan features	65
4.5.2	Road signs	66
5	Railway design	70
5.1	Track geometry	70
5.2	String line model	70
5.3	KM-posting	71
5.4	Cross-sections and track information	73
5.4.1	Cross-sections	73
5.4.2	Track information	75
5.4.3	Switch information	78
5.5	Terrain model and structural model of the track	78
5.6	Rail plan features	79
5.6.1	Plan features	80
6	Railway signs	80
7	Waterway design	81
7.1	Waterway geometry	81
7.2	Cross section parameters	81
7.3	String line model	83
7.4	Surface and structural model	84
7.5	Waterway plan features	85
7.5.1	Plan features	85
8	Waterway signs	86
9	Area structures	87
9.1	Landscaping	87
9.2	Noise barriers	88
9.3	Geostructures	89
9.4	Surface structures	90
10	Pipenetworks	92
10.1	Pipe networks	93
10.1.1	Plan information	94

10.2	Pipe network	94
10.2.1	Pipe network type extensions	95
10.3	Units	96
10.4	Structures	96
10.4.1	Circular structures	100
10.4.2	Rectangular structures	102
10.4.3	Pipe inlets and outlets	103
10.4.4	Pipe connections	105
10.4.5	Equipment	106
10.5	Pipes	107
10.5.1	Pipe	108
10.6	Center	109
10.7	Details	109
10.7.1	Egg pipes	113
10.7.2	Elliptic pipes	114
10.7.3	Rectangular pipes	115
10.7.4	Channels	116
11	Planimetric features	118
11.1	Plan features	118
11.1.1	Plan feature	119
12	AsBuilt data	130
12.1	Control points	130
12.1.1	Control point	130
12.1.2	Tolerances	131
12.2	As-built survey	135
12.2.1	Survey equipment	136
12.2.2	Survey points	137
13	Deep foundations	140
13.1	Pile groups	140
13.2	Pile group	140
13.3	Pile	143
14	Inframodel <Feature> extensions	145
14.1	Local coordinate transformation definition by point pairs	145
14.2	Type coding systems	149
14.3	Type coding	153
14.4	User defined properties	157
14.5	Plan information	158
14.6	Quantity information	160
14.7	Soil properties	162
14.8	String line model	165
14.9	Cross-section parameters	167
14.10	Structural layer properties	170

14.11 Road and street design - Road signs	176
14.12 Railway design - KM post coordinates	179
14.13 Railway design - switches	180
14.14 Utility networks - network type	181
14.15 Utility networks - structure details	182
14.16 Utility networks - pipe details	187
14.17 Utility networks - cable details	190
14.18 Plan features	192
14.19 As-built survey points	201
14.20 Spatial allocation and spatial avoidance	207

15 Annex 1. Inframodel käyttöohje (Q&A) 209

15.1 Johdanto	209
15.2 Ohjeen sisältö	211
15.3 Käyttöohje + Q&A	211
15.3.1 Yleistä	211
15.3.2 Headers	212
15.3.3 Base data	214
15.3.4 Route planning	215
15.3.5 Roads and street	216
15.3.6 Railways	216
15.3.7 Waterways	217
15.3.8 Areas	217
15.3.9 Water supply and sewerage	217
15.3.10 Planimetric features	218
15.3.11 As-built	221
15.3.12 Deep foundations	222
15.3.13 Extensions	223

1 File headers

1.1 XML file header

Inframodel XML files shall use UTF-8 character encoding, and encoder attribute shall be set on XML header.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

1.2 LandXML container

The namespaces in Inframodel file shall be the following:

- The default namespace to be used without prefix for all LandXML elements specialized for Inframodel in schema inframodel.xsd shall be set in the root element as
“https://github.com/buildingSMART-Finland/InfraModel/releases/tag/4.2.0”
- The namespace for elements in Inframodel extension schema im.xsd (if used in the file) to be used with prefix “im” shall be set in the root element as
“https://github.com/buildingSMART-Finland/InfraModel/releases/tag/4.2.0/im”

Note: The namespace URI is not meant to be used to look up information. Its sole purpose is to give the namespace a unique name.

The schema locations may be set in an Inframodel transfer file, in which case XML Schema Instance namespace shall be declared in the root element: xmlns:xsi=“http://www.w3.org/2001/XMLSchema-instance”, and the following schema locations may be set to access online:

- The schema location (xsi:schemaLocation) for the default namespace
https://github.com/buildingSMART-Finland/InfraModel/releases/tag/4.2.0 as
https://github.com/buildingSMART-Finland/InfraModel/releases/download/4.2.0/inframodel.xsd
- If elements from “im” namespace
https://github.com/buildingSMART-Finland/InfraModel/releases/tag/4.2.0/im are used in the file, the
“im.xsd” schema location (xsi:schemaLocation) as
https://github.com/buildingSMART-Finland/InfraModel/releases/download/4.2.0/im.xsd

The root element (<LandXML>) of the transfer file is used by software to check the validity of the file structure.

A <LandXML> shall have the fields defined in Table 1.1:

Table 1.1: <LandXML> fields.

Field	Type	Use	Description
One <Units> element	<Units>	Required	Units used on file, only certain fixed SI units are allowed by this specification.

Field	Type	Use	Description
One <i><CoordinateSystem></i> element	<i><CoordinateSystem></i>	Required	Coordinate system definition with optional transformation reference points.
One <i><Project></i> element	<i><Project></i>	Required	Defines project base data, ie. classification system definitions
At least one <i><Application></i> element	<i><Application></i>	Required	Identifies the software and author(s) of the file creator and editor(s).
Zero or more <i><Alignments></i> elements	<i><Alignments></i>	Optional	A collection of horizontal Alignments with optional vertical profile(s)
Zero or more <i><CgPoints></i> elements	<i><CgPoints></i>	Optional	A collection of COGO points. (Cg = COGO = Coordinate Geometry)
Zero or more <i><Parcels></i> elements	<i><Parcels></i>	Optional	A collection of Parcels
Zero or more <i><PlanFeatures></i> elements	<i><PlanFeatures></i>	Optional	A collection of planimetric features, ie. footings
Zero or more <i><PipeNetworks></i> elements	<i><PipeNetworks></i>	Optional	Pipe network collection
Zero or more <i><Roadways></i> elements	<i><Roadways></i>	Optional	Roadways collection which contain planimetric features of a route, ie. signage
Zero or more <i><Surfaces></i> elements	<i><Surfaces></i>	Optional	A collection of surface models.
Zero or more <i><Survey></i> elements	<i><Survey></i>	Optional	Survey collection, contains the surveyed as-built data
At least one <i><FeatureDictionary></i> element	<i><FeatureDictionary></i>	Required	Defines the extension sets used by other elements. Each Property element defines one piece of data.
date	<i>date</i>	Required	UTC date of file
time	<i>time</i>	Required	UTC time of file
version	<i>string</i>	Required	XML schema version, fixed value 1.2

XML example of <LandXML>:

```
<LandXML>
  <Units></Units>
  <CoordinateSystem></CoordinateSystem>
  <Project></Project>
  <Application></Application>
  <Alignments></Alignments>
  <CgPoints></CgPoints>
  <Parcels></Parcels>
  <PlanFeatures></PlanFeatures>
  <PipeNetworks></PipeNetworks>
  <Roadways></Roadways>
  <Surfaces></Surfaces>
  <Survey></Survey>
  <FeatureDictionary></FeatureDictionary>
  <im:Metadata></im:Metadata>
  <im:LocalCoordinateTransformation></im:LocalCoordinateTransformation>
  <im:PileGroups></im:PileGroups>
</LandXML>
```

1.3 Units

The units used in the file are defined by the <Units> element. Only certain metric SI system units are allowed, and those are defined under the sub-element <Units><Metric>.

Angular (angularUnit) and direction (directionUnit) units are defined counter-clockwise from the base direction. In angular definitions the base direction is east, and in direction definitions it is north.

An <Units> shall have the fields defined in Table 1.2:

Table 1.2: <Units> fields.

Field	Type	Use	Description
One <Metric> element	<Metric>	Required	File units

A <Metric> shall have the fields defined in Table 1.3:

Table 1.3: <Metric> fields.

Field	Type	Use	Description
areaUnit	metArea	Required	Fixed value, squareMeter
linearUnit	metLinear	Required	Fixed value, meter
volumeUnit	metVolume	Required	Fixed value, cubicMeter

Field	Type	Use	Description
temperatureUnit	<i>metTemperature</i>	Required	Fixed value, celsius
pressureUnit	<i>metPressure</i>	Required	Fixed value, milliBars
diameterUnit	<i>metDiameter</i>	Required	Fixed value, meter
widthUnit	<i>metWidth</i>	Required	Fixed value, meter
heightUnit	<i>metHeight</i>	Required	Fixed value, meter
elevationUnit	<i>elevationType</i>	Required	Fixed value, meter
velocityUnit	<i>metVelocity</i>	Required	Fixed value, kilometersPerHour
angularUnit	<i>angularType</i>	Required	Fixed value, radians.
directionUnit	<i>angularType</i>	Required	Fixed value, radians.
latLongAngularUnit	<i>latLongAngularType</i>	Required	Fixed value, radians.
flowUnit	<i>metFlow</i>	Required	Fixed value, literPerSecond

1.4 Coordinate and height systems

The height and coordinate system information is defined in the element `<CoordinateSystem>`. Exactly one coordinate system shall be defined:

- by defining the `<CoordinateSystem>.epsgCode` using the European Petrol Survey Group (EPSG) naming system without a prefix.
- or by defining local system(s) with `<CoordinateSystem>.horizontalCoordinateSystemName` (and optionally the `<CoordinateSystem>.verticalCoordinateSystemName`).

`<CoordinateSystem>.desc` attribute may provide informal information about the system used.

It is also possible to set a `rotationAngle` for the coordinate system.

A `<CoordinateSystem>` shall have the fields defined in Table 1.4:

Table 1.4: `<CoordinateSystem>` fields.

Field	Type	Use	Description
Zero or one <code><Start></code> element	<code><Start></code>	Optional	Represents the base point of coordinate system
Zero or one <code>IM_coordTransformation</code> <code><Feature></code> element	<code>IM_coordTransformation</code> <code><Feature></code>	Optional	Inframodel localization <code><Feature></code> extension

Field	Type	Use	Description
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
desc	<i>string</i>	Optional	Optional coordinate system description
epsgCode	<i>string</i>	Optional	EPSG code of coordinate system
horizontalCoordinateSystemName	<i>string</i>	Optional	User defined horizontal coordinate system name.
verticalCoordinateSystemName	<i>string</i>	Optional	User defined local vertical coordinate system name
rotationAngle	<i>angle</i>	Optional	Rotation angle of the coordinate system

1.4.1 Local coordinate transformation by point pairs

Local coordinate system may be defined as set of control points sourceCRS-targetCRS point pairs under “IM_coordTransformation” <Feature> extension.

See section **14.1** *Local coordinate transformation definition by point pairs* for detailed information

1.4.2 Local coordinate transformation by transformation parameters

The exact parameters of a local coordinate transformation may be given using <im:LocalCoordinateTransformation> element in im-extension schema as <any> element under <LandXML>.

The im namespace xml schema (im.xsd) for the extension schema elements is available at Inframodel schema page.

An <im:LocalCoordinateTransformation> shall have the fields defined in Table 1.5:

Table 1.5: <im:LocalCoordinateTransformation> fields.

Field	Type	Use	Description
One <im:SourceCRS> element	<im:SourceCRS>	Required	SourceCRS describes the source coordinate reference system

Field	Type	Use	Description
One <i><im:TargetCRS></i> element	<i><im:TargetCRS></i>	Required	TargetCRS describes parameters for the target coordinate reference system
Zero or one <i><im:DatumTransformation></i> element	<i><im:DatumTransformation></i>	Optional	Datum transformation from Source to Target datum.
One <i><im:Projection></i> element	<i><im:Projection></i>	Required	Specifies projection to use for projecting the geographical coordinates to a plane and its parameters.
One <i><im:LocalTransformation></i> element	<i><im:LocalTransformation></i>	Required	Transformation from the projected coordinates to the final local coordinates. This consists of first a 2D transformation for the plane coordinates and then a transformation for the height (using a fitted plane). The plane can be skipped and one can use a real geoid model instead, using the existing markup (<i>verticalCoordinateSystemName</i>).

Coordinate systems use a reference ellipsoid, defined by the semi-major and semi-minor axis, to approximate the shape of the Earth. The datum is then defined by the ellipsoid and its location and orientation, i.e. different datums can use the same ellipsoid but its position varies.

For example, the WGS84 system uses a reference ellipsoid with a semi-major axis of 6 378 137m and a semi-minor axis of 6 356 752m. The ellipsoid center is located at the Earth's center of mass.

1.4.2.1 Source coordinate reference system

An *<im:SourceCRS>* shall have the fields defined in Table 1.6:

Table 1.6: <im:SourceCRS> fields.

Field	Type	Use	Description
Zero or one <im:Ellipsoid> element	<im:Ellipsoid>	Optional	Ellipsoid definition
Zero or one <im:PrimeMeridian> element	<im:PrimeMeridian>	Optional	Prime meridian definition
name	string	Required	Source coordinate reference system name ie. WGS84
epsg	string	Required	EPSG code of the source coordinate reference system

where

An <im:Ellipsoid> shall have the fields defined in Table 1.7:

Table 1.7: <im:Ellipsoid> fields.

Field	Type	Use	Description
name	string	Required	Ellipsoid name
semiMajorAxis	double	Required	Ellipsoid semi-major axis, meters
inverseFlattening	double	Required	Ellipsoid inverse flattening

An <im:PrimeMeridian> shall have the fields defined in Table 1.8:

Table 1.8: <im:PrimeMeridian> fields.

Field	Type	Use	Description
name	string	Required	Prime meridian name
value	double	Required	Prime meridian value, decimal degrees

1.4.2.2 Target coordinate reference system

An <im:TargetCRS> shall have the fields defined in Table 1.9:

Table 1.9: <im:TargetCRS> fields.

Field	Type	Use	Description
Zero or one <im:Ellipsoid> element	<im:Ellipsoid>	Optional	Ellipsoid definition
Zero or one <im:PrimeMeridian> element	<im:PrimeMeridian>	Optional	Prime meridian definition
name	string	Required	Target coordinate reference system name ie. WGS84
epsg	string	Required	EPSG code of the target coordinate reference system

1.4.2.3 Datum transformation

An <im:DatumTransformation> shall have the fields defined in Table 1.10:

Table 1.10: <im:DatumTransformation> fields.

Field	Type	Use	Description
One <im:Helmert3D> element	<im:Helmert3D>	Required	3D helmert transformation

Helmert3D performs a coordinate transformation from one datum to another.

An <im:Helmert3D> shall have the fields defined in Table 1.11:

Table 1.11: <im:Helmert3D> fields.

Field	Type	Use	Description
rotationX	double	Required	Rotation around X-axis
rotationY	double	Required	Rotation around Y-axis
rotationZ	double	Required	Rotation around Z-axis
translationX	double	Required	X-axis translation
translationY	double	Required	Y-axis translation
translationZ	double	Required	Z-axis translation
scale	double	Required	Scale in ppm

1.4.2.4 Projection

An *<im:Projection>* shall have the fields defined in Table 1.12:

Table 1.12: *<im:Projection>* fields.

Field	Type	Use	Description
One <i><im:TransverseMercator></i> element	<i><im:TransverseMercator></i>	Required	Transverse mercator projection

Transverse Mercator Map projection *<im:Projection>.<im:TransverseMercator>* transforms geographical coordinates (latitude, longitude, altitude) to a plane (x, y, z). The grid origin is taken on the central latitude and longitude, and false easting and northing is then applied to prevent negative coordinates west or south of the origin.

An *<im:TransverseMercator>* shall have the fields defined in Table 1.13:

Table 1.13: *<im:TransverseMercator>* fields.

Field	Type	Use	Description
falseEasting	<i>double</i>	Required	False easting value,meters
falseNorthing	<i>double</i>	Required	False northing value,meters
longitude0	<i>double</i>	Required	Projection center longitude,decimal degrees
latitude0	<i>double</i>	Required	Projection center latitude,decimal degrees
scale0	<i>double</i>	Required	Projection scale in ppm

1.4.2.5 Local transformation

In LocalTransformation *<im:LocalTransformation>*, Helmert2D *<im:Helmert2D>* transforms the projected (x,y,z) coordinates to the local coordinate system. FittedPlane *<im:FittedPlane>* corrects height values using a plane as the geoid model. The corrected height at point (northing,easting,elevation) is $elevation_corrected = elevation + (a * northing + b * easting + c)$.

An *<im:Helmert2D>* shall have the fields defined in Table 1.14:

Table 1.14: *<im:Helmert2D>* fields.

Field	Type	Use	Description
rotation	<i>double</i>	Required	Rotation around z-axis
scale	<i>double</i>	Required	Scale in ppm
dn	<i>double</i>	Required	Translation in northing(X) axis
de	<i>double</i>	Required	Translation in easting(Y) axis

An *<im:FittedPlane>* shall have the fields defined in Table 1.15:

Table 1.15: *<im:FittedPlane>* fields.

Field	Type	Use	Description
a	<i>double</i>	Required	Northing slope coefficient in ppm
b	<i>double</i>	Required	Easting slope coefficient in ppm
c	<i>double</i>	Required	Constant adjustment (height offset)

1.5 Project

<Project> element defines base data of the project, including it's name description and classification system definitions.

The description may contain ie. the project long name or code.

The state attribute may be used to describe the state of the project and its content. Sub-elements of the file however may override the state value defined here by setting their own state attribute.

Table 1.16: *<Project>* fields.

Field	Type	Use	Description
One <i>IM_codings <Feature></i> element	<i>IM_codings <Feature></i>	Required	Inframodel type coding system definition <i><Feature></i> extension
Zero or more <i>IM_proprietaryCodings <Feature></i> elements	<i>IM_proprietaryCodings <Feature></i>	Optional	Inframodel proprietary feature coding system definition <i><Feature></i> extension
Zero or more <i>IM_userDefinedProperties <Feature></i> elements	<i>IM_userDefinedProperties <Feature></i>	Optional	Inframodel user defined custom properties <i><Feature></i> extension
name	<i>nameType</i>	Required	Project name
desc	<i>string</i>	Optional	Project description
state	<i>stateType</i>	Optional	Project state

XML example of *<Project>*:

```
<Project name="E66" desc="Road to inframodel" state="proposed">
  <Feature name="12B1BB1" code="IM_codings" source="inframodel">
    <Property label="terrainCoding" value="Infra"></Property>
  </Feature>
</Project>
```

```

<Property label="terrainCodingDesc" value="Finnish Transport Infrastructure Agency terrain codin
<Property label="terrainCodingSourceRef" value="vayla.fi"></Property>
<Property label="surfaceCoding" value="InfraBIM"></Property>
<Property label="surfaceCodingDesc" value="Finnish InfraBIM surface coding"></Property>
<Property label="surfaceCodingSourceRef" value=" buildingsmart.fi"></Property>
<Property label="infraCoding" value="InfraBIM"></Property>
<Property label="infraCodingDesc" value="Finnish InfraBIM coding"></Property>
<Property label="infraCodingSourceRef" value=" buildingsmart.fi"></Property>
</Feature>
<Feature name="674D7" code="IM_proprietaryCodings" source="inframodel">
  <Property label="proprietaryInfraCoding" value="XYCoding"></Property>
  <Property label="proprietaryInfraCodingDesc" value="my own coding system"></Property>
  <Property label="proprietaryInfraCodingSourceRef" value=" buildingsmart.fi"></Property>
</Feature>
<Feature name="88A1C3" code="IM_userDefinedProperties" source="inframodel">
  <Property label="propertyLabel" value="Manufacturer"></Property>
  <Property label="propertyValue" value="CompanyXY"></Property>
  <Property label="propertyDescription" value="Manufacturer name"></Property>
  <Property label="propertySource" value="companyxy.com"></Property>
</Feature>
</Project>

```

where:

A *stateType* is defined per Table 1.17:

Table 1.17: <stateType> definition.

Type	Basetype	Restriction	Description
stateType	<i>string</i>	One of (abandoned, destroyed, existing, proposed)	State of object

1.6 Type coding systems

The meaning (semantic) of the points, lines and surfaces is defined in the file. The parties of a project agree on type coding systems that are used in the data transfer.

The main coding systems are set in the “IM_codings” extension (*exactly one in each file*) using <Feature> element under <Project>, defining:

1. The terrain description coding system (source data points and breaklines)(*terrainCoding*)
2. The surface/category description coding system (*surfaceCoding*)
3. The coding system for infrastructure objects (including alignments and breaklines, pipe networks, plan features) (*infraCoding*)

The existing terrain description contains source data points and breaklines. The surface description consists of the individual surfaces of the base data (terrain and ground layers) or the planned route or areal structures as TIN surface model or string line model. In addition to surfaces, planned objects may be described as alignment geometry, line strings or points. It is possible to set the same type coding system for more than one of these.

In addition to the main coding systems, it is also possible to define additional or alternative type coding systems (*none, one or more e.g. Company X etc.*), using “IM_proprietaryCodings” extension (one instance per coding system) under <Project>. When a code from a proprietary system is used for an element, each “IM_proprietaryCoding” <Feature> instance placed under the element being coded shall identify the coding system by its property *proprietaryInfraCodingSource*, having the same value as the system name set in “IM_proprietaryCodings” property *proprietaryInfraCoding*.

Detailed information about “IM_codings”, “IM_proprietaryCodings” and “IM_userDefinedProperties” <Feature> extensions can be found from section **14 Inframodel <Feature> extensions**

1.7 Application

The <Application> element describes what software was used to create the file. If the file has been created using several different applications, all are described by their own <Application> element.

An <Application> shall have the fields defined in Table 1.18:

Table 1.18: <Application> fields.

Field	Type	Use	Description
Zero or more <Author> elements	<Author>	Optional	Author information
name	<i>nameType</i>	Required	Application name
desc	<i>string</i>	Optional	Application description
manufacturer	<i>string</i>	Required	Application manufacture
version	<i>string</i>	Required	Application version
manufacturerURL	<i>string</i>	Required	Application webpage URL
timeStamp	<i>datetime</i>	Optional	Time of file writing as UTC time

1.8 Authors

Information of the author of the file is recorded in the sub-element <Application>.<Author>. It is possible to define several authors as separate <Author>-elements.

An <Author> shall have the fields defined in Table 1.19:

Table 1.19: <Author> fields.

Field	Type	Use	Description
createdBy	<i>string</i>	Required	Author name
createdByEmail	<i>string</i>	Optional	Author email address
company	<i>string</i>	Optional	Author company name
companyURL	<i>string</i>	Optional	Author company webpage url

Field	Type	Use	Description
timeStamp	<i>datetime</i>	Optional	Time of writing as UTC time

1.9 Feature dictionary

The `<FeatureDictionary>` identifies the specification source of extensions used in the file, and the point of access to their documentation.

The contents of `<Feature>` elements shall follow the source specification. LandXML-files in general may contain extensions from several different sources. In Inframodel file transfer, proper recognition and interpretation is required only for the extensions documented in this specification (e.g. for the type coding systems used in an Inframodel file).

The dictionary for Inframodel extensions shall be specified using `<FeatureDictionary>` element as shown in the table below.

The name attribute shall be unique, and always “inframodel” for the dictionary of Inframodel extensions, and exactly the same value shall be set in every Inframodel `<Feature>` for attribute source (the `<Feature>` attribute code being labeled with IM_ -prefix).

The `<version>` should match the version number of the Inframodel schema.

`<DocFileRef>` element can be used to provide the URI link to named external documentation where applicable feature code and property type values are described (section **14** *Inframodel <Feature> extensions* in the case of Inframodel feature dictionary).

A `<FeatureDictionary>` shall have the fields defined in Table 1.20:

Table 1.20: `<FeatureDictionary>` fields.

Field	Type	Use	Description
Zero or more <code><DocFileRef></code> elements	<code><DocFileRef></code>	Optional	A reference to external document file containing definitions of extensions
name	<i>nameType</i>	Required	Feature dictionary name
version	<i>string</i>	Required	Feature dictionary version

Proprietary extensions can be included in addition to Inframodel extensions, as “IM_userDefinedProperties” (generic extension specified in Inframodel feature dictionary)

For more information about “IM_userDefinedProperties” see section **14.4** *User defined properties*

1.10 Metadata

Metadata is described with the `<im:Metadata>` element (specified im-extension schema) as `<any>` element under `<LandXML>`. The im namespace xml schema (im.xsd) for the extension schema elements is available

at Inframodel schema page.

Metadata is optional and enables the following features shown below.

An *<im:Metadata>* shall have the fields defined in Table 1.21:

Table 1.21: *<im:Metadata>* fields.

Field	Type	Use	Description
One <i><im:CreatedBy></i> element	<i><im:CreatedBy></i>	Required	Person who has created the data
Zero or one <i><im:DataOwner></i> element	<i><im:DataOwner></i>	Optional	Organisation who owns the data
Zero or one <i><im:History></i> element	<i><im:History></i>	Optional	Data creation and modification information
Zero or one <i><im:DataQuality></i> element	<i><im:DataQuality></i>	Optional	Group of attributes which describe the quality from data collection point of view
Zero or one <i><im:ModelStatus></i> element	<i><im:ModelStatus></i>	Optional	This field identifies the status of process phase for this document. This is used to determine processing of the record.
Description	<i>string</i>	Required	Description of the data set
Status	<i>one of (final, draft, other, unknown)</i>	Optional	Data status
Language	<i>string</i>	Optional	Metadata language
MetadataURL	<i>string</i>	Optional	URL pointing to location of the metadata http://www.paikkatietoikkuna.fi/web/fi/aUIID=924a68ba-665f-4ea0-a830-26e80112b5dc
InformationProductURL	<i>string</i>	Optional	URL pointing to location which has the details about information product
PlanningStatus	<i>string</i>	Optional	Planning status

Where:

An *<im:CreatedBy>* shall have the fields defined in Table 1.22:

Table 1.22: *<im:CreatedBy>* fields.

Field	Type	Use	Description
One <i><im:ContactInformation></i> element	<i><im:ContactInformation></i>	Required	Contact information within company
Company	<i>string</i>	Required	Company name
CompanyURL	<i>string</i>	Optional	Company URL

An *<im:ContactInformation>* shall have the fields defined in Table 1.23:

Table 1.23: *<im:ContactInformation>* fields.

Field	Type	Use	Description
Firstname	<i>string</i>	Required	First name
Lastname	<i>string</i>	Required	Last name
Phone	<i>string</i>	Optional	Phone number
Email	<i>string</i>	Optional	Email address

An *<im:DataOwner>* shall have the fields defined in Table 1.24:

Table 1.24: *<im:DataOwner>* fields.

Field	Type	Use	Description
Zero or one <i><im:ContactInformation></i> element	<i><im:ContactInformation></i>	Optional	Contact information within company
Company	<i>string</i>	Required	Company name
CompanyURL	<i>string</i>	Optional	Company URL

An *<im:History>* shall have the fields defined in Table 1.25:

Table 1.25: *<im:History>* fields.

Field	Type	Use	Description
One <i><im:CreateTimestamp></i> element	<i><im:CreateTimestamp></i>	Required	Creation timestamp

Field	Type	Use	Description
Zero or more <im:ModifiedTimestamp> elements	<im:ModifiedTimestamp>	Optional	Modification timestamp(s)

An <im:ModifiedTimestamp> shall have the fields defined in Table 1.26:

Table 1.26: <im:ModifiedTimestamp> fields.

Field	Type	Use	Description
Zero or one <im:ContactInformation> element	<im:ContactInformation>	Optional	Contact information within company
TimeStamp	datetime	Required	Modification time as UTC
Company	string	Optional	Company name
CompanyURL	string	Optional	Company WWW URL
Description	string	Optional	Description

An <im:DataQuality> shall have the fields defined in Table 1.27:

Table 1.27: <im:DataQuality> fields.

Field	Type	Use	Description
One <im:Source> element	<im:Source>	Required	When and where is data originated from
CreateMethod	string	Required	Free text describing how data was created from the Source
Accuracy	string	Optional	Free text describing the data accuracy

An <im:Source> shall have the fields defined in Table 1.28:

Table 1.28: <im:Source> fields.

Field	Type	Use	Description
Date	date	Required	Date
Datasource	string	Required	Source of this data

An *<im:ModelStatus>* shall have the fields defined in Table 1.29:

Table 1.29: *<im:ModelStatus>* fields.

Field	Type	Use	Description
label	<i>one of (preliminary plan, general plan, plan, construction plan, construction, maintenance)</i>	Optional	Status
desc	<i>string</i>	Optional	Description

2 Base data

The base data contains the data points and breaklines of the source data, as well as the triangulated representation of mesh surfaces.

The surface description contains the points used to form the surface (as vertices of the triangles in TIN)).

Breaklines are not used to form surfaces, but if transferred under <Surface> as source data they shall coincide with the triangulation defining the surface, i.e. each pair of consecutive breakline points given in 3D coordinates, must match the coordinates of two vertices of a triangle (exactly, within the numeric precision of the exchange file).

Surface meshes are used for terrain or ground layer models (top surface of each layer) as described in this section.

They are also used for terrain (visible surface) or structural models of roads, streets, railways, waterways and area structures, as described in following sections.

Surfaces and source data is described as surface groups <Surfaces>, which are made of individual <Surface>-elements.

A <Surfaces> shall have the fields defined in Table 2.1:

Table 2.1: <Surfaces> fields.

Field	Type	Use	Description
At least one <Surface> element	<Surface>	Required	Surface mesh
Zero or one <i>IM_plan</i> <Feature> element	<i>IM_plan</i> <Feature>	Optional	Inframodel plan information <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
desc	<i>string</i>	Optional	Description
name	<i>nameType</i>	Optional	Unique name
state	<i>stateType</i>	Optional	State

A <Surface> shall have the fields defined in Table 2.2:

Table 2.2: <Surface> fields.

Field	Type	Use	Description
Zero or one <SourceData> element	<SourceData>	Optional	The collection of data that was used to triangulate the surface.
Zero or one <Definition> element	<Definition>	Optional	The collection of faces and points that defined the surface.
Zero or one <i>IM_soil</i> <Feature> element	<i>IM_soil</i> <Feature>	Optional	Inframodel soil properties <Feature> extension
Zero or one <i>IM_structLayer</i> <Feature> element	<i>IM_structLayer</i> <Feature>	Optional	Inframodel structural layer properties <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one <i>IM_quantity</i> <Feature> element	<i>IM_quantity</i> <Feature>	Optional	Inframodel quantities <Feature> extension
name	<i>nameType</i>	Required	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

2.1 Plan information

For the surface description the project is divided into surface groups <Surfaces> with optional “IM_plan” <Feature> extension.

If the project consists of sub-projects that have different rates of progress, the plan contents of the file are divided into sub-projects according to the same division.

Note: The plan state is described according to a scheme agreed on by the parties of the project.

An *IM_plan* <Feature> shall have the fields defined in Table 2.3:

Table 2.3: IM_plan <Feature> fields.

Field	Type	Use	Description
One <i>planName</i> <Property> element	<i>planName</i> <Property>	Required	Plan name
Zero or one <i>planCode</i> <Property> element	<i>planCode</i> <Property>	Optional	Plan code
Zero or one <i>planDesc</i> <Property> element	<i>planDesc</i> <Property>	Optional	Plan description
Zero or one <i>planState</i> <Property> element	<i>planState</i> <Property>	Optional	Plan state
code	<i>string</i>	Required	Fixed value, IM_plan
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

2.1.1 Current and planned surfaces

An existing surface is defined by setting the state of the <Surfaces> or <Surface> element to “existing”.

If all the surfaces within a surface group have the same state, it is possible to set the state on a higher level in the surface group <Surfaces>.

However, <Surface> element (or its sourcedata element(s)) may override the parent state by defining its own state.

2.1.2 Type coding

The type coding systems used in Inframodel file transfers are set in the header information. Type codes can be set for individual plan elements:

- In the element for source data points
- In the element for breaklines
- In the element for part of surface
- In the element for surface
- In the element for surface group

Type coding is set by “IM_Coding” <Feature> extension under a parent element.

An *IM_coding* <Feature> shall have the fields defined in Table 2.4:

Table 2.4: IM_coding <Feature> fields.

Field	Type	Use	Description
Zero or one <i>terrainCoding</i> <Property> element	<i>terrainCoding</i> <Property>	Optional	Terrain code
Zero or one <i>terrainCodingDesc</i> <Property> element	<i>terrainCodingDesc</i> <Property>	Optional	Terrain code description
Zero or one <i>surfaceCoding</i> <Property> element	<i>surfaceCoding</i> <Property>	Optional	Surface code
Zero or one <i>surfaceCodingDesc</i> <Property> element	<i>surfaceCodingDesc</i> <Property>	Optional	Surface code description
Zero or one <i>infraCoding</i> <Property> element	<i>infraCoding</i> <Property>	Optional	Feature code
Zero or one <i>infraCodingDesc</i> <Property> element	<i>infraCodingDesc</i> <Property>	Optional	Feature code description
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, IM_coding
source	<i>string</i>	Required	Fixed value, inframodel

Individual type codes are set primarily in parent elements, from which the child elements will inherit the values.

Terrain points and breaklines are type coded using the **terrainCoding** and a coding description **terrainCodingDesc**.

It is optional to set a **surfaceCoding** and a surface coding description **surfaceCodingDesc** for terrain points and breaklines.

Surfaces are given a **surfaceCoding** and a surface coding description **surfaceCodingDesc**, and optionally **terrainCoding** and a coding description **terrainCodingDesc**.

These both may be given an **infraCoding** and its description **infraCodingDesc**.

Alternative type codings can be given using “IM_proprietaryCoding” with **proprietaryInfraCoding** and their descriptions **proprietaryInfraCodingDesc**, and with **proprietaryInfraCodingSource** where they both have prefix per proprietary coding systems named under <Project> element.

Type coding set by the parent element is also inherited by the child elements, ie. <Surfaces> element may also set the type coding of its child elements.

2.1.3 Quantity information

Calculated area or volume quantities may be assigned to entire <Surface>, or part of it in source data <Boundary>, using “IM_quantity” extension:

An *IM_quantity* <Feature> shall have the fields defined in Table 2.5:

Table 2.5: IM_quantity <Feature> fields.

Field	Type	Use	Description
Zero or one <i>areaQuantity</i> <Property> element	<i>areaQuantity</i> <Property>	Optional	Area
Zero or one <i>volumeQuantity</i> <Property> element	<i>volumeQuantity</i> <Property>	Optional	Volume
Zero or one <i>quantityCoding</i> <Property> element	<i>quantityCoding</i> <Property>	Optional	Quantity code
Zero or one <i>quantityCodingDescription</i> <Property> element	<i>quantityCodingDescription</i> <Property>	Optional	Quantity code description
Zero or one <i>quantityCodingSource</i> <Property> element	<i>quantityCodingSource</i> <Property>	Optional	Quantity coding system name
Zero or one <i>description</i> <Property> element	<i>description</i> <Property>	Optional	Description or note
Zero or one <i>surfaceRef</i> <Property> element	<i>surfaceRef</i> <Property>	Optional	Reference to the name of <Surface> element (applicable if volume calculation is performed between two surface elements)
code	<i>string</i>	Required	Fixed value, IM_quantity
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

2.2 Source data

The source data is described by the element <SourceData>. This element has no attributes.

Source data consists of:

- Source data points as <DataPoints> and
- Source breaklines as <BreakLines>
- Source boundaries as <Boundaries>

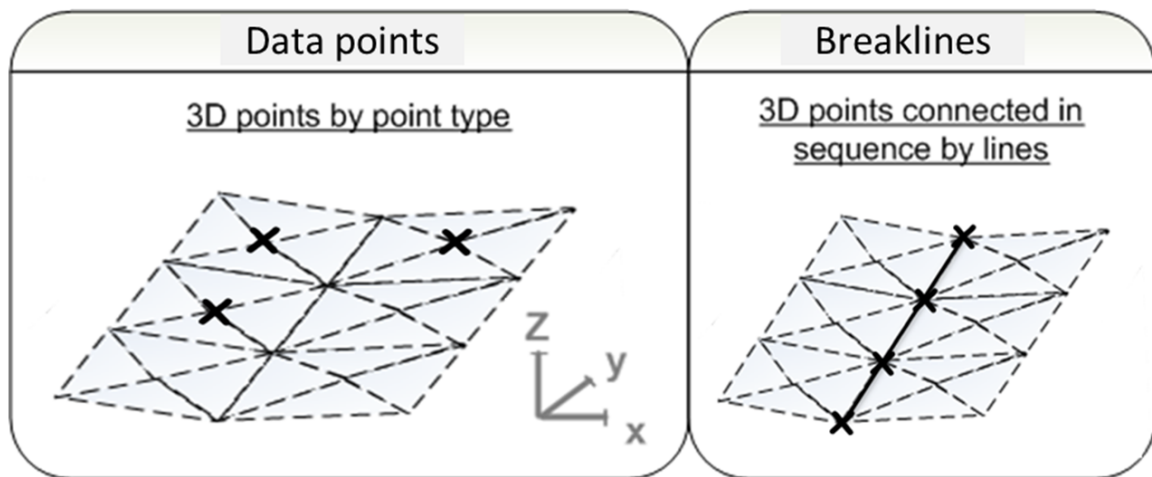


Figure 2.1: Source data

A *<SourceData>* shall have the fields defined in Table 2.6:

Table 2.6: *<SourceData>* fields.

Field	Type	Use	Description
At least one <i><Boundaries></i> element	<i><Boundaries></i>	Required	Boundaries that was used to define the surface.
At least one <i><Breaklines></i> element	<i><Breaklines></i>	Required	Breaklines that was used to define the surface.
At least one <i><DataPoints></i> element	<i><DataPoints></i>	Required	Data points that was used to define the surface.

2.2.1 Data points

Source *data points* are described by the element *<DataPoints>*, sorting every point group into individual elements.

A *<DataPoints>* shall have the fields defined in Table 2.7:

Table 2.7: *<DataPoints>* fields.

Field	Type	Use	Description
Zero or more <i><PntList3D></i> elements	<i><PntList3D></i>	Optional	A sequential space delimited list of 3D coordinates.

Field	Type	Use	Description
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

2.2.2 Breaklines

Source *breakline group* is described by the element <BreakLines>, where each <BreakLine> presents single continous line.

A <Breaklines> shall have the fields defined in Table 2.8:

Table 2.8: <Breaklines> fields.

Field	Type	Use	Description
Zero or more <Breakline> elements	<Breakline>	Optional	Individual surface breakline

A <Breakline> shall have the fields defined in Table 2.9:

Table 2.9: <Breakline> fields.

Field	Type	Use	Description
One <PntList3D> element	<PntList3D>	Required	A sequential space delimited list of 3D coordinates with a minimum of 2 points.
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension

Field	Type	Use	Description
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
brkType	<i>breakLineType</i>	Optional	Breakline type
desc	<i>string</i>	Optional	Description
name	<i>nameType</i>	Optional	Unique name
state	<i>stateType</i>	Optional	State

2.2.3 Boundaries

Additionally, it is also possible to define boundaries of the source data in the boundary group <Boundaries>, where each <Boundary> is presented in its own element. Each boundary may have properties of the area as “IM_surfaceStructure”, “IM_structLayer” or “IM_soil”, as well as calculated area or volume quantities as “IM_quantity”.

A <Boundaries> shall have the fields defined in Table 2.10:

Table 2.10: <Boundaries> fields.

Field	Type	Use	Description
At least one <Boundary> element	<Boundary>	Required	The boundary region contains a 2D north/east or 3D north/east/elev list of points that define the geometry.

A <Boundary> shall have the fields defined in Table 2.11:

Table 2.11: <Boundary> fields.

Field	Type	Use	Description
One <PntList2D> or <PntList3D> element	<PntList2D> or <PntList3D>	Required	Boundary may be represented as PntList2D or PntList3D element
Zero or one <i>IM_soil</i> <Feature> element	<i>IM_soil</i> <Feature>	Optional	Inframodel soil properties <Feature> extension

Field	Type	Use	Description
Zero or one <i>IM_structLayer</i> <Feature> element	<i>IM_structLayer</i> <Feature>	Optional	Inframodel structural layer properties <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one <i>IM_quantity</i> <Feature> element	<i>IM_quantity</i> <Feature>	Optional	Inframodel quantities <Feature> extension
Zero or one <i>IM_surfaceStructure</i> <Feature> element	<i>IM_surfaceStructure</i> <Feature>	Optional	Inframodel surface structure <Feature> extension
bndType	<i>surfBndType</i>	Required	Boundary type
edgeTrim	<i>boolean</i>	Required	Edge trimming flag
area	<i>double</i>	Optional	Boundary area
desc	<i>string</i>	Optional	Description
name	<i>nameType</i>	Optional	Unique name
state	<i>stateType</i>	Optional	State

Where:

A *surfBndType* is defined per Table 2.12:

Table 2.12: <surfBndType> definition.

Type	Basetype	Restriction	Description
surfBndType	<i>string</i>	One of (outer, void, island)	Surface boundaries can be one of three types: outer, void, island

An *IM_surfaceStructure* <Feature> shall have the fields defined in Table 2.13:

Table 2.13: IM_surfaceStructure <Feature> fields.

Field	Type	Use	Description
One <i>material</i> <Property> element	<i>material</i> <Property>	Required	Surface material name
Zero or one <i>thickness</i> <Property> element	<i>thickness</i> <Property>	Optional	Surface material thickness
Zero or one <i>surfaceRef</i> <Property> element	<i>surfaceRef</i> <Property>	Optional	Reference to the name of <Surface> element
code	<i>string</i>	Required	Fixed value, IM_surfaceStructure
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

An IM_structLayer <Feature> shall have the fields defined in Table 2.14:

Table 2.14: IM_structLayer <Feature> fields.

Field	Type	Use	Description
Zero or one <i>material</i> <Property> element	<i>material</i> <Property>	Optional	Material name of the structural layer
Zero or one <i>grainSizeMin</i> <Property> element	<i>grainSizeMin</i> <Property>	Optional	Minimum grain size of the material
Zero or one <i>grainSizeMax</i> <Property> element	<i>grainSizeMax</i> <Property>	Optional	Maximum grain size of the material
Zero or one <i>nominalThickness</i> <Property> element	<i>nominalThickness</i> <Property>	Optional	Nominal thickness of the product
Zero or one <i>strength</i> <Property> element	<i>strength</i> <Property>	Optional	Material strength
Zero or one <i>E-module</i> <Property> element	<i>E-module</i> <Property>	Optional	E-module used in dimensioning road structure
Zero or one <i>loadCapacity</i> <Property> element	<i>loadCapacity</i> <Property>	Optional	Load-bearing capacity or the structure
Zero or one <i>pavementType</i> <Property> element	<i>pavementType</i> <Property>	Optional	Pavement type
Zero or one <i>surfacingMethod</i> <Property> element	<i>surfacingMethod</i> <Property>	Optional	Surfacing method
Zero or one <i>stoneStructureType</i> <Property> element	<i>stoneStructureType</i> <Property>	Optional	Stone structure type

Field	Type	Use	Description
Zero or one <i>stoneType</i> <Property> element	<i>stoneType</i> <Property>	Optional	Stone type
Zero or one <i>stabilizationMethod</i> <Property> element	<i>stabilizationMethod</i> <Property>	Optional	Stabilization method
Zero or one <i>lightweightFillMaterial</i> <Property> element	<i>lightweightFillMaterial</i> <Property>	Optional	Lightweight fill material
code	<i>string</i>	Required	Fixed value, IM_structLayer
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

An *IM_soil* <Feature> shall have the fields defined in Table 2.15:

Table 2.15: *IM_soil* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>soiltypeGEO</i> <Property> element	<i>soiltypeGEO</i> <Property>	Optional	Soil type classification according to Geotekninen maaluokitus (VTT 1974)
Zero or one <i>soiltypeISO</i> <Property> element	<i>soiltypeISO</i> <Property>	Optional	Soil type classification according to SFS-EN ISO 14688-1 or SFS-EN ISO 14688-2
Zero or one <i>qualificationClass</i> <Property> element	<i>qualificationClass</i> <Property>	Optional	Soil type qualification according to table 10 in TIEH 2100029-04
Zero or one <i>frostSwellingFactorWet</i> <Property> element	<i>frostSwellingFactorWet</i> <Property>	Optional	Soil type frost swelling factor in wet conditions
Zero or one <i>frostSwellingFactorDry</i> <Property> element	<i>frostSwellingFactorDry</i> <Property>	Optional	Soil type frost swelling factor in dry conditions
Zero or one <i>elasticModulusWet</i> <Property> element	<i>elasticModulusWet</i> <Property>	Optional	Soil type modulus of elasticity in wet conditions (MPa)

Field	Type	Use	Description
Zero or one <i>elasticModulusDry</i> <Property> element	<i>elasticModulusDry</i> <Property>	Optional	Soil type modulus of elasticity in dry conditions (MPa)
code	<i>string</i>	Required	Fixed value, IM_soil
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Type *IM_quantity* <Feature> is defined per Table 2.5 above.

2.3 Triangular mesh surface

Surface geometry is described as triangulated meshes.

Each surface is defined under the <Definition> in terms of boundaries, exterior features and holes.

A triangular mesh is defined in two steps:

- First by defining the vertices of the triangular faces as surface points,
- Then each individual face by three vertices.

The surface points used as vertices are assigned unique identifiers *id* within the same surface definition <Surface>.<Definition> element.

The face definitions are done by referring to the *id* numbers *id* of the vertex points.

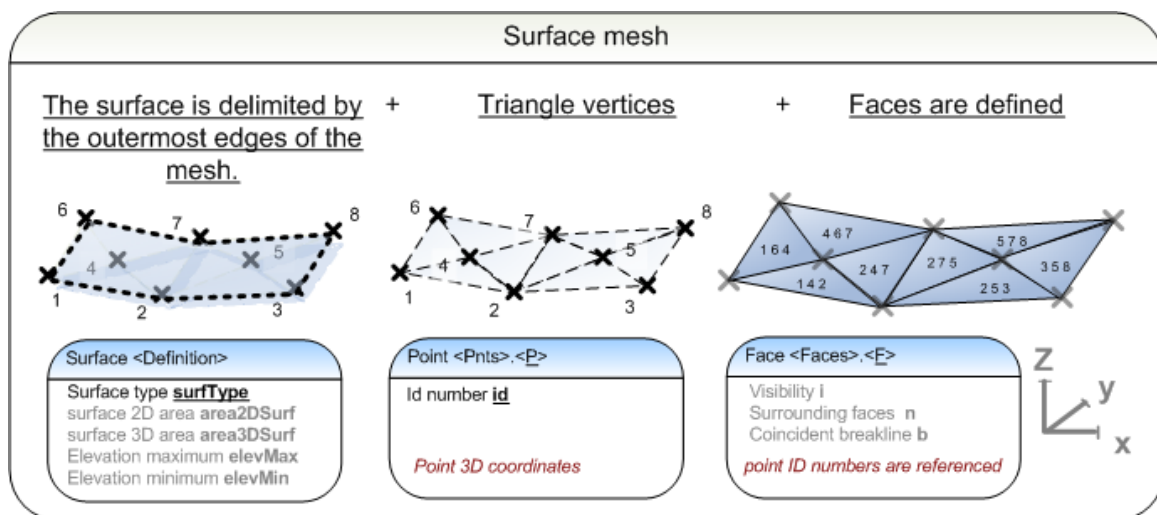


Figure 2.2: Triangular mesh

A <Definition> shall have the fields defined in Table 2.16:

Table 2.16: <Definition> fields.

Field	Type	Use	Description
One <Pnts> element	<Pnts>	Required	The collection of points that define the surface.
At least one <Faces> element	<Faces>	Required	The collection of faces that define the surface.
surfType	surfTypeEnum	Required	Surface definition type, fixed value “TIN”
area2DSurf	double	Optional	2D surface area
area3DSurf	double	Optional	3D surface area

The surface type surfType is fixed to “TIN” when describing a triangular mesh.
The precision of the mesh model depends on the available software and data.

2.3.1 Vertices

The *vertex point group* <Pnts> contains a listing of individual vertices <P>, which are each assigned an individual id number id.

A <Pnts> shall have the fields defined in Table 2.17:

Table 2.17: <Pnts> fields.

Field	Type	Use	Description
At least 3 <P> elements	<P>	Required	A surface point. it contains an id attribute and a space delimited “northing easting elevation” text value. The id values are referenced by the surface faces for the coordinate values.

A <P> shall have the fields defined in Table 2.18:

Table 2.18: <P> fields.

Field	Type	Use	Description
content	3 of type double	Required	3d point
name	nameType	Optional	Unique name
desc	string	Optional	Description

Field	Type	Use	Description
code	<i>string</i>	Optional	Feature code
state	<i>stateType</i>	Optional	State
id	<i>unsigned integer</i>	Required	Id of the point “P”. Id shall be unique in scope of single surface

2.3.2 Faces

The *triangulation* is defined by the <Faces> collection. It consists of consecutive list of faces <F>. The order of the faces implicitly defines the index number of each triangle (1,2,..). Each face is defined by referencing three vertex id numbers.

A <Faces> shall have the fields defined in Table 2.19:

Table 2.19: <Faces> fields.

Field	Type	Use	Description
At least one <F> element	<F>	Required	Individual surface face. It contains a space delimited list of “id” references for 3 surface “P” points.
desc	<i>string</i>	Optional	Description
name	<i>nameType</i>	Optional	Unique name
state	<i>stateType</i>	Optional	State

A <F> shall have the fields defined in Table 2.20:

Table 2.20: <F> fields.

Field	Type	Use	Description
<i>content</i>	<i>3 of type integer</i>	Required	Face definition
i	<i>integer</i>	Optional	Invalid triangle flag. If i=1, the face should be hidden
n	<i>integer</i>	Optional	Surrounding faces defined as integer numbers. if there is a face adjacent to the face described.

Field	Type	Use	Description
b	unsigned integer	Optional	Coincidence with breakline. Sum value indicating which edges touch breakline(value 0-7): edge 1=1,edge 2 = 2,edge 3 = 4.E.g. value 4 = only edge 3 touches breakline

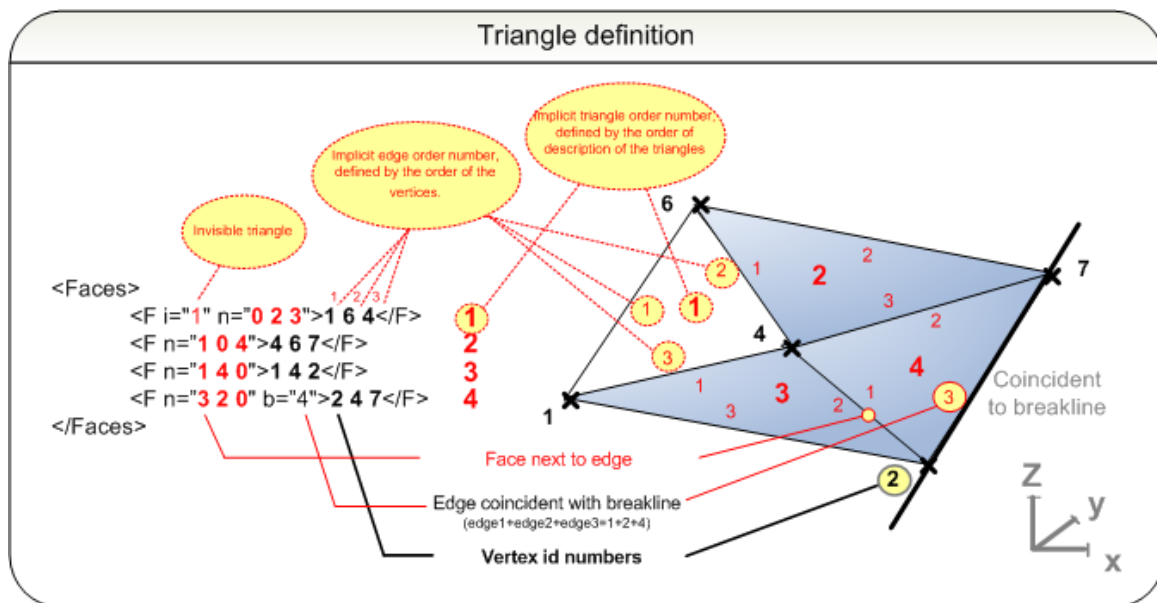


Figure 2.3: Triangle face definition

2.4 Terrain model

The *terrain model* contains the description of the topmost terrain surface (also Relief or Digital Elevation Model) as one or more <Surface> under <Surfaces> *surface group*.

It consists of the vertices of the component faces and the faces as explained above.

In inframodel file transfers it is also possible to assign source data points and breaklines to the surface.

An “IM_coding” <Feature> extension enables surface classifications, and “IM_soil” <Feature> extension allows to add the technical properties.

Terrain model may be part of a optional plan described in “IM_plan” <Feature> defined in section 14.5 *Plan information*.

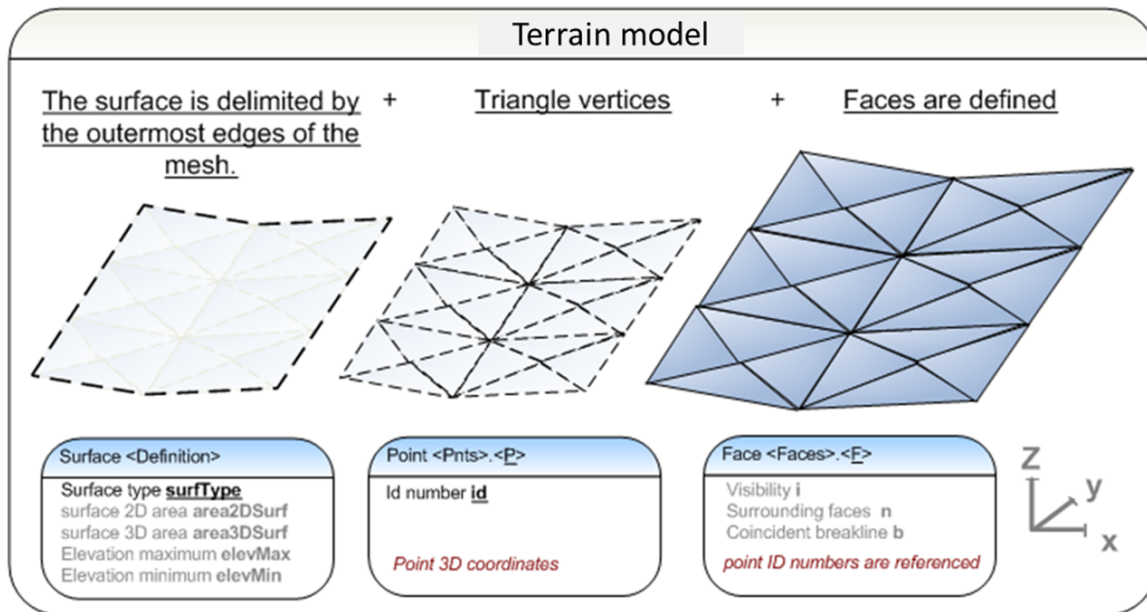


Figure 2.4: Terrain model

2.4.1 Soil properties

When no information of individual ground layers is available, surface model may define the soil properties below the topmost surface by using "IM_soil" <Feature> extension.

Type *IM_soil* <Feature> is defined per Table 2.15 above.

2.5 Ground layer model

The *ground layer model* contains a description of all the surfaces between different ground layers (and the topmost surface) in the plan.

It is recommended that surfaces are described top-down. Individual layer surfaces are constructed as explained above.

A surface may be part of a plan described in "IM_plan" <Feature>.

A "IM_coding" <Feature> extension provides surface classifications.

The technical properties of each soil layer between two surfaces may be given in "IM_soil" <Feature> defined in section **14.7 Soil properties**.

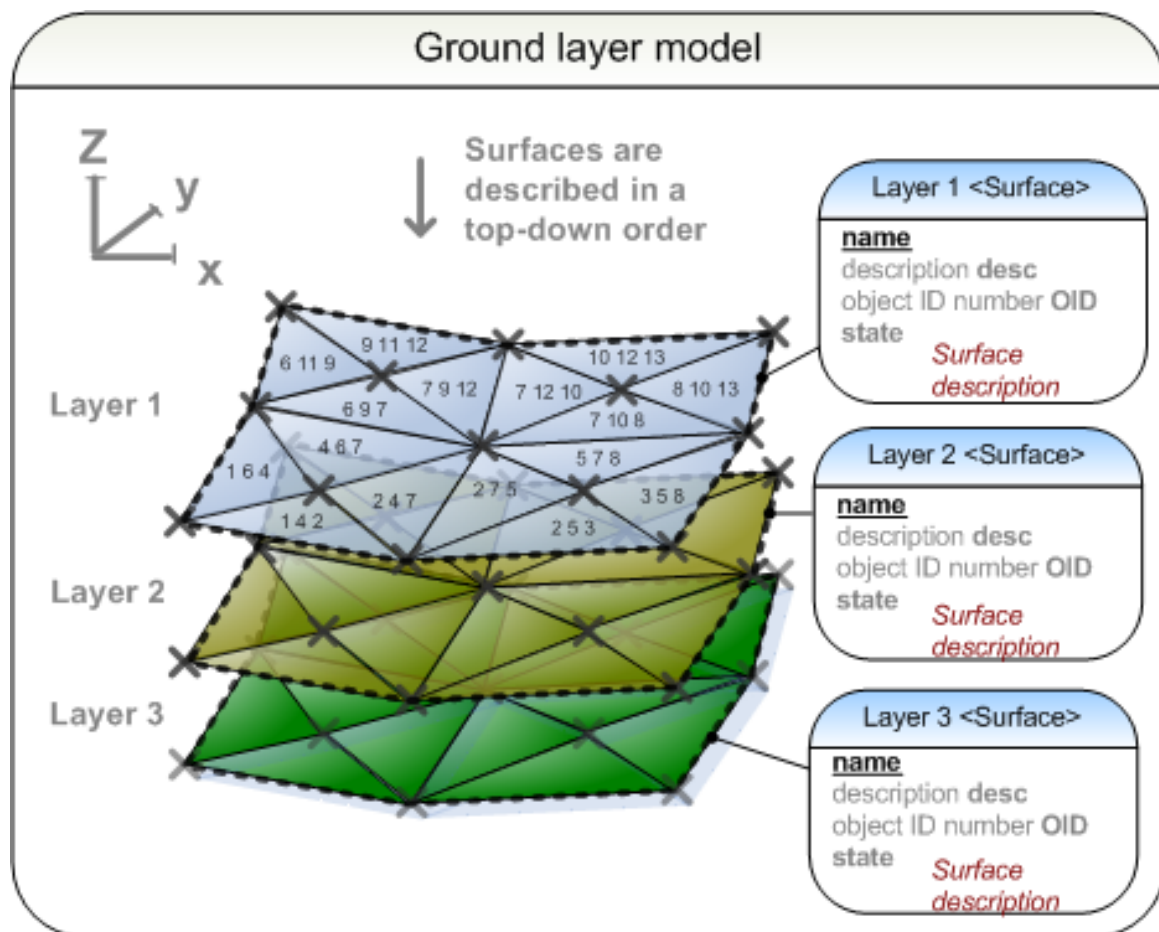


Figure 2.5: Ground layer model

3 Route planning

Routes encompass highways, local roads and private roads, waterways and railways. Each route has one continuous stationing reference alignment and a vertical alignment. In inframodel file transfer, a route plan may consist of route geometric alignments, their stringline models and surface or structural models as triangulated meshes.

Geometric alignments describe parameters of the horizontal and optional vertical elements of an alignment. A line string is a description where consecutive points are connected by line segments (in 2D or 3D). Geometric alignments are typically used to describe the stationing reference line of a road as well as other important geometric descriptions such as road edges. Other route components are usually described as line strings.

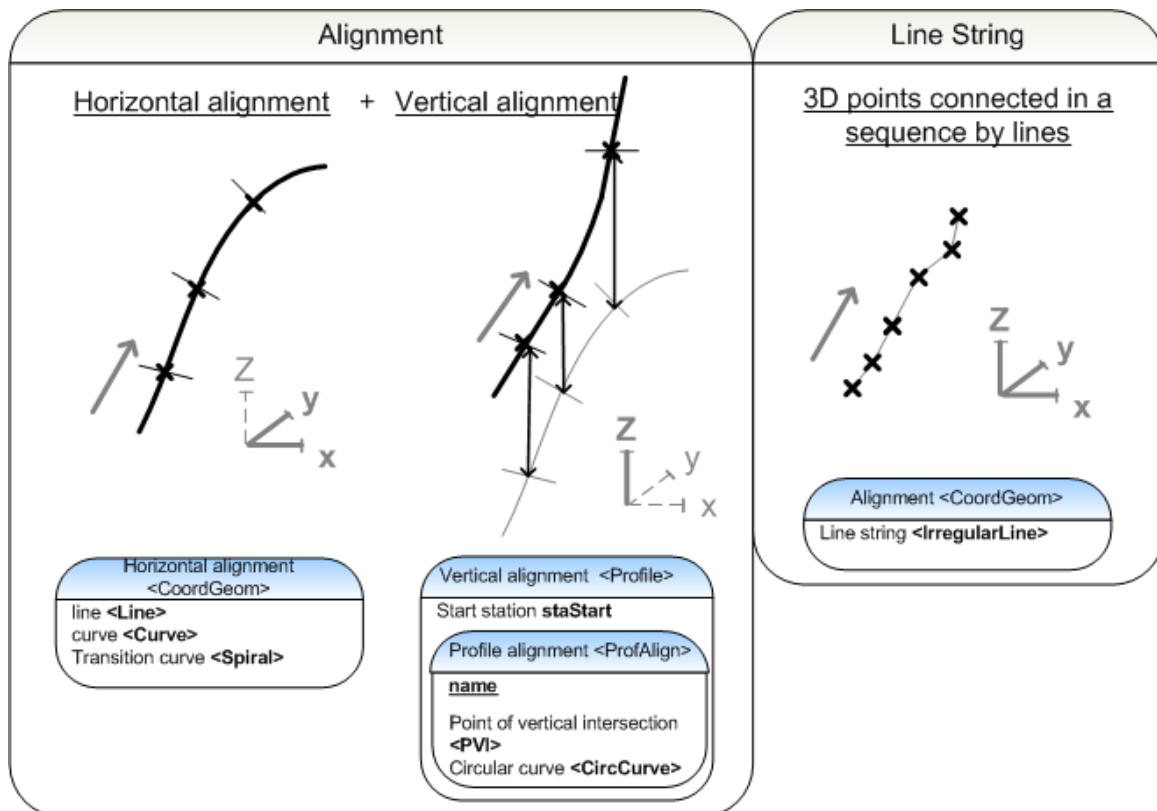


Figure 3.1: Route geometry

Once the alignments have been described, it is possible to assign them to a string line model, that contains a description of the layers of the route structure, as described in section 3.6 *String line model*. Alternatively, a triangulated surface mesh model can be used to represent the top surface of a route, or its structural model can be composed by describing all its layers as triangulated mesh surfaces.

Cross-section parameters, which are described in further detail in the sections covering each route type, complement the route description with design parameter information of the cross-sections (without actual cross section geometry).

3.1 Route description

Route description is driven by stationing reference line (principal alignment). Other geometry lines are given in the same *alignment collection* <Alignments> each as separate *alignment* <Alignment>. Geometry lines and

string lines are given in separate *alignment collections* <Alignments>.

Different routes, alignment options and stationing reference line discontinuities are placed in *separate* <Alignments>.

3.2 Naming and Type coding

Alignment groups and each individual *alignment* within a group must be assigned unique name. It is advisable to use different naming convention for *geometric alignments* and line strings in string line models.

Assigning *type codes* to alignments is optional in Inframodel file transfers. The type coding systems to be used for an <Alignment> shall be defined in the project information. The type code can then be set for each <Alignment> element (whose children inherit the applied coding) in the extension “IM_coding” with the **infraCoding** and its description **infraCodingDesc**. Alternative or additional type coding systems can be used (if defined in the project information) and each such type code is set for <Alignment> element using “IM_proprietaryCoding” with **proprietaryInfraCoding**, reference to the coding system (defined in the project information) by **proprietaryInfraCodingSource** and description **proprietaryInfraCodingDesc**.

Layers of a string line model, described in optional “IM_stringLineLayers” extension may also have type codes, as well as the triangulated meshes surfaces in surface or structural models, using “IM_coding” with **surfaceCoding** and its description **surfaceCodingDesc**, or also as “IM_proprietaryCoding”.

3.3 Composing alignments

The names *name* of *alignment groups* <Alignments> are unique. The **state** attribute is optional. The *description* attribute **desc** is also optional and may be used to describe the *alignment group* <Alignments> in further detail.

An <Alignments> shall have the fields defined in Table 3.1:

Table 3.1: <Alignments> fields.

Field	Type	Use	Description
At least one <Alignment> element	<Alignment>	Required	Geometric alignment
Zero or one IM_plan <Feature> element	IM_plan <Feature>	Optional	Inframodel plan information <Feature> extension
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one IM_stringlineLayers <Feature> element	IM_stringlineLayers <Feature>	Optional	Inframodel strigline layer collection<Feature> extension

Field	Type	Use	Description
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

An **<Alignment>** is an element that describes

1. a geometric alignment or
2. a line string

The alignments within a file do not have to be presented in any particular order. It is, however, advisable to first describe geometric alignments and then line strings. The **<Alignment>** definition describes a **name**, **length**, *the stationing start* **staStart** and the **state** of the **<Alignment>**. It is recommended that lines are named in an intuitive fashion. If the **state** is set for the entire alignment group **<Alignments>** the **<Alignment>** elements will inherit the **state** attribute from the parent element, hence it should not be set. When alternative alignments are being described by different *alignment groups* the differences between elements can be described briefly in the attribute **desc**. The optional *object identifying number* **oid** makes object management easier in applications.

An **<Alignment>** shall have the fields defined in Table 3.2:

Table 3.2: <Alignment> fields.

Field	Type	Use	Description
One <CoordGeom> element	<CoordGeom>	Required	Geometry
Zero or one <Cant> element	<Cant>	Optional	The “Cant” element represents a proposed railway cant / superelevation alignment definitions and railway design speed change stations.
Zero or more <StaEquation> elements	<StaEquation>	Optional	Station equation (KM-posting)
Zero or one <Profile> element	<Profile>	Optional	A profile (long section)
Zero or one <CrossSects> element	<CrossSects>	Optional	Cross section definitions
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension

Field	Type	Use	Description
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Required	Unique name
length	<i>double</i>	Required	Alignment length
staStart	<i>double</i>	Required	Station offset
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

A *geometric alignment* contains a horizontal geometry in a **<CoordGeom>** element and the corresponding *vertical alignment* in a **<Profile>.<ProfAlign>** element. Line strings are described as a chain of 3D points in the **<CoordGeom>** element.

3.3.1 Plan information

The *plan information* of an *alignment group* is described under the **<Alignments>** element in the optional extension “IM_plan”. If the plan consists of subsets that progress at a different rate or there is some other reason to partition the project into smaller entities, these subsets should be sorted into separate *alignment groups*. The *plan information* is also set when describing the surfaces of a route. These are set in the “IM_plan” extension of the **<Surfaces>** element.

Type *IM_plan* <Feature> is defined per Table 2.3 above.

3.4 Geometric alignments

The geometric alignment contains the horizontal and vertical alignment information. The *horizontal alignment* information is described in the **<CoordGeom>** and the corresponding (0 or 1) *vertical geometry* in the element **<Profile>.<ProfAlign>**. For the connection between horizontal and vertical geometry it is crucial that the geometric description is continuous from the beginning of the first element to the end of the last element. The *horizontal geometry* is described using a 2D coordinate representation, and the final elevation values along the element can only be produced once the vertical geometry is finished. The illustration below shows the horizontal and vertical geometry definition and their connection principal, the optional **staStart** attribute in **<Line>**, **<Curve>**, **<Spiral>** and **<Profile>** **SHALL NOT** be used for calculating horizontal or vertical geometry.

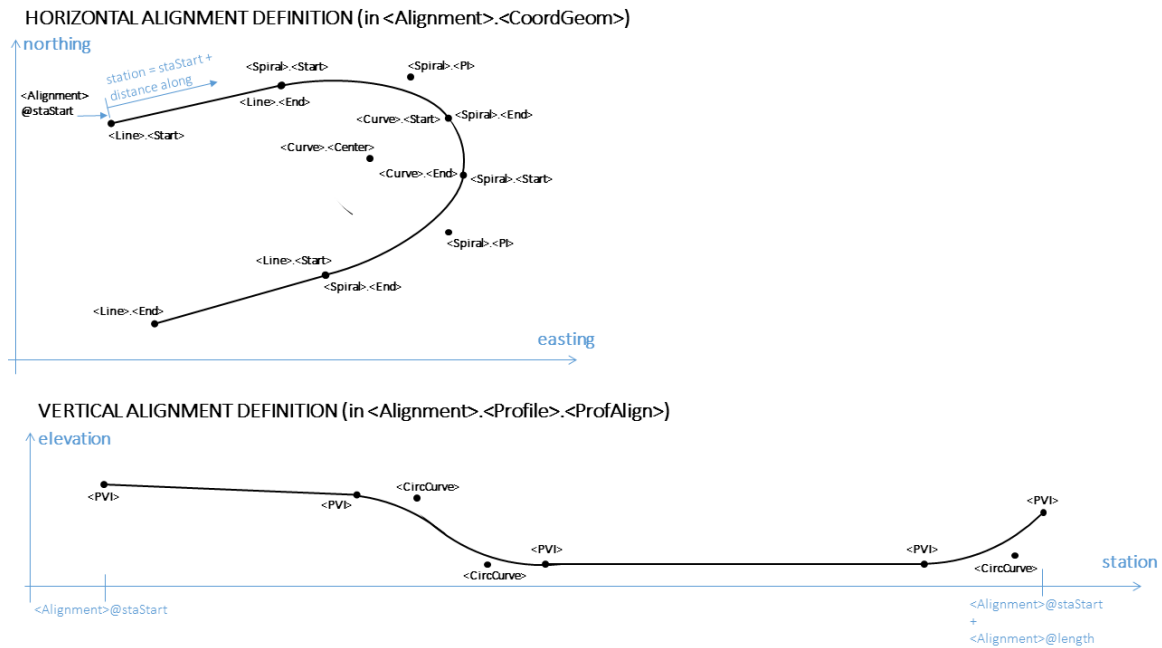


Figure 3.2: Alignment

3.4.1 Horizontal geometry

The dimensioning components of horizontal alignments:

- `<Line>`
- `<Curve>`
- `<Spiral>`

The horizontal alignment is a listing of consecutive dimensioning components, starting at the **staStart**. The precise location of the elements is defined in terms of 2D coordinates.

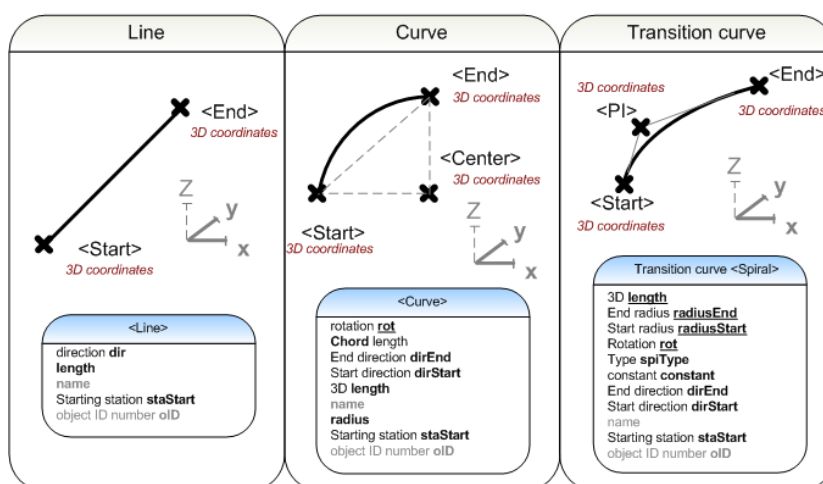


Figure 3.3: Horizontal geometry

inframodel does not use attributes for the `<CoordGeom>` element.

A `<CoordGeom>` shall have the fields defined in Table 3.3:

Table 3.3: <CoordGeom> fields.

Field	Type	Use	Description
Zero or more <Line> elements	<Line>	Optional	Line element
Zero or more <IrregularLine> elements	<IrregularLine>	Optional	Used to record lines that are irregular such as river boudaries etc. It has Start and End point elements and a list of intermediate points. Point list should also include the start and end points.
Zero or more <Curve> elements	<Curve>	Optional	The distance from the Start to the Center provides the radius value.The rotation attribute “rot” defines whether the arc travels clockwise or counter-clockwise from the Start to End point.
Zero or more <Spiral> elements	<Spiral>	Optional	An “infinite” spiral radius is denoted by the value “INF”. This conforms to XML Schema which defines infinity as “INF” or “-INF” for all numeric datatypes
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	nameType	Optional	Unique name
desc	string	Optional	Description
state	stateType	Optional	State

3.4.1.1 Line

A <Line> is defined by <Start> and <End> 2D coordinates. In addition, attributes *direction* **dir** and **length** are mandatory, but shall be used as additional information only.

A **<Line>** shall have the fields defined in Table 3.4:

Table 3.4: **<Line>** fields.

Field	Type	Use	Description
One <Start> element	<Start>	Required	Represents a 2D or 3D Starting or beginning Point
One <End> element	<End>	Required	Represents a 2D or 3D ending Point
Zero or one <i>IM_switch</i> <Feature> element	<i>IM_switch</i> <Feature>	Optional	Inframodel railway switch <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
staStart	<i>double</i>	Optional	Station offset
length	<i>double</i>	Optional	Line length
dir	<i>direction</i>	Optional	Line direction
state	<i>stateType</i>	Optional	State

3.4.1.2 Curve

A circular arc **<Curve>** is defined by **<Start>** **<Center>** and **<End>** 2D coordinates. In addition, attributes *direction of rotation* **rot**, **chord**, *end direction* **dirEnd**, *start direction* **dirStart**, **length** and **radius** are mandatory, but shall be used as additional information only.

A **<Curve>** shall have the fields defined in Table 3.5:

Table 3.5: **<Curve>** fields.

Field	Type	Use	Description
One <Start> element	<Start>	Required	Represents a 2D or 3D Starting or beginning Point
One <Center> element	<Center>	Required	Represents a 2D or 3D center Point
One <End> element	<End>	Required	Represents a 2D or 3D ending Point
Zero or one <PI> element	<PI>	Optional	Represents a 2D or 3D point of Intersection
rot	<i>clockwise</i>	Required	Rotation direction
chord	<i>double</i>	Required	Chord length
length	<i>double</i>	Required	Curve length

Field	Type	Use	Description
radius	<i>double</i>	Required	Curve radius
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
crvType	<i>curveType</i>	Optional	Curve type
dirStart	<i>direction</i>	Optional	Direction at start of the curve
dirEnd	<i>direction</i>	Optional	Direction at end of the curve
staStart	<i>double</i>	Optional	Station offset
state	<i>stateType</i>	Optional	State
tangent	<i>double</i>	Optional	Tangent length

3.4.1.3 Transition curve

A **<Spiral>** is defined by **<Start>**, point of intersection of the end tangents **<PI>** and **<End>** 2D coordinates, together with mandatory attribute *transition curve type* **spiType**. In addition, attributes **length**, end radius **radiusEnd**, start radius **radiusStart**, direction of rotation **rot**, the transition curve parameter **constant**, end direction **dirEnd** and start direction **dirStart** are mandatory. In Finnish route design the default *transition curve type* is an Euler spiral “clothoid”; bi-quadratic parabola “biquadraticParabola”, or third-degree spiral “cubic” may be used under special circumstances e.g. in railway design.

NOTE: since attribute **spiType** is mandatory, but has no meaning for “biquadraticParabola” or “cubic”, it shall have value set to “NaN” in these cases.

Type **<Curve>** is defined per Table 3.5 above.

A **<Spiral>** shall have the fields defined in Table 3.6:

Table 3.6: **<Spiral>** fields.

Field	Type	Use	Description
One <Start> element	<Start>	Required	Represents a 2D or 3D Starting or beginning Point
One <PI> element	<PI>	Required	Represents a 2D or 3D point of Intersection
One <End> element	<End>	Required	Represents a 2D or 3D ending Point
length	<i>double</i>	Required	Spiral length
radiusStart	<i>double</i>	Required	Radius at start of the spiral, use “INF” for infinite
radiusEnd	<i>double</i>	Required	Radius at end of the spiral, use “INF” for infinite

Field	Type	Use	Description
rot	<i>clockwise</i>	Required	Rotation direction
spiType	<i>spiralType</i>	Required	Spiral type
constant	<i>double</i>	Required	Spiral constant
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
chord	<i>double</i>	Optional	Chord length
dirStart	<i>direction</i>	Optional	Direction at start of the spiral
dirEnd	<i>direction</i>	Optional	Direction at end of the spiral
staStart	<i>double</i>	Optional	Station offset
state	<i>stateType</i>	Optional	State

Where:

A *spiralType* is defined per Table 3.7:

Table 3.7: <spiralType> definition.

Type	Basetype	Restriction	Description
spiralType	<i>string</i>	One of (clothoid, cubic, biquadraticParabola)	

The **<Start>**, point on intersection of start and end tangents **<PI>** and **<End>** are defined as 2D coordinates separated by spaces.

3.4.2 Vertical geometry

The vertical geometry is described in the **<Profile>**.**<ProfAlign>** element in concert with the horizontal geometry. In Inframodel, each horizontal geometry can have only one (or 0) vertical geometry. The dimensioning components of the vertical geometry are:

- Point of Vertical Intersection **<PVI>**
- Vertical circular arc **<CircCurve>**

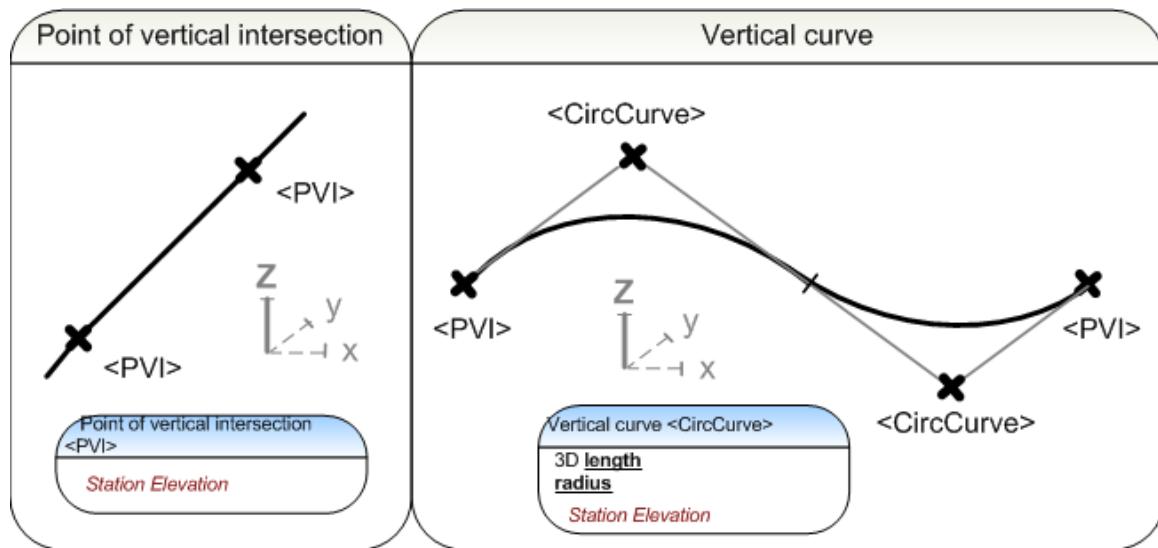


Figure 3.4: Vertical geometry

A *<Profile>* shall have the fields defined in Table 3.8:

Table 3.8: *<Profile>* fields.

Field	Type	Use	Description
Zero or more <i><ProfAlign></i> elements	<i><ProfAlign></i>	Optional	Represents a vertical alignment for a profile.
Zero or more <i>IM_userDefinedProperties</i> <i><Feature></i> elements	<i>IM_userDefinedProperties</i> <i><Feature></i>	Optional	Inframodel user defined custom properties <i><Feature></i> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
staStart	<i>double</i>	Optional	Station offset
state	<i>stateType</i>	Optional	State

A *<ProfAlign>* shall have the fields defined in Table 3.9:

Table 3.9: *<ProfAlign>* fields.

Field	Type	Use	Description
Zero or more <i><PVI></i> elements	<i><PVI></i>	Optional	Represents a Point of Vertical Intersection with a space delimited “station elevation” text value

Field	Type	Use	Description
Zero or more <i><ParaCurve></i> elements	<i><ParaCurve></i>	Optional	A Point of Vertical Intersection with a space delimited “station elevation” text value and a parabolic vertical curve defined by the “length” attribute.
Zero or more <i><UnsymParaCurve></i> elements	<i><UnsymParaCurve></i>	Optional	A Point of Vertical Intersection with a space delimited “station elevation” text value and a parabolic vertical curve defined by the “lengthIn” and “lengthOut” attributes.
Zero or more <i><CircCurve></i> elements	<i><CircCurve></i>	Optional	A Point of Vertical Intersection with a space delimited “station elevation” text value and curvature definition
name	<i>nameType</i>	Required	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

3.4.2.1 Point of vertical intersection

The first and last element of the *vertical profile* is always a *Point of Vertical Intersection* **<PVI>**.

A *Point of Vertical Intersection* **<PVI>** marks the ends of the line segments of a vertical geometry. A *Point of Vertical Intersection* is described by a **station** and an **elevation**. These are separated by a space.

A *<PVI>* shall have the fields defined in Table 3.10:

Table 3.10: *<PVI>* fields.

Field	Type	Use	Description
<i>content</i>	<i>2 of type Point</i>	Required	2d point
<i>desc</i>	<i>string</i>	Optional	Description

3.4.2.2 Vertical curve

Vertical circular arcs may be combined into S-curves or compound curves.

The location of the **<CircCurve>** is defined by the *station* and *elevation*, separated by spaces.

A **<CircCurve>** shall have the fields defined in Table 3.11:

Table 3.11: **<CircCurve>** fields.

Field	Type	Use	Description
content	2 of type <i>Point</i>	Required	2d point
length	<i>double</i>	Required	Curve length
radius	<i>double</i>	Required	Curve radius
desc	<i>string</i>	Optional	Description

3.5 Line strings

Line strings are defined in concert with the *horizontal geometry* **<CoordGeom>**. *Line strings* are defined as a series of 2D points or 3D points, hence it does not need a vertical **<Profile>** element for 3D representation. The dimensioning element of a line string is **<IrregularLine>**

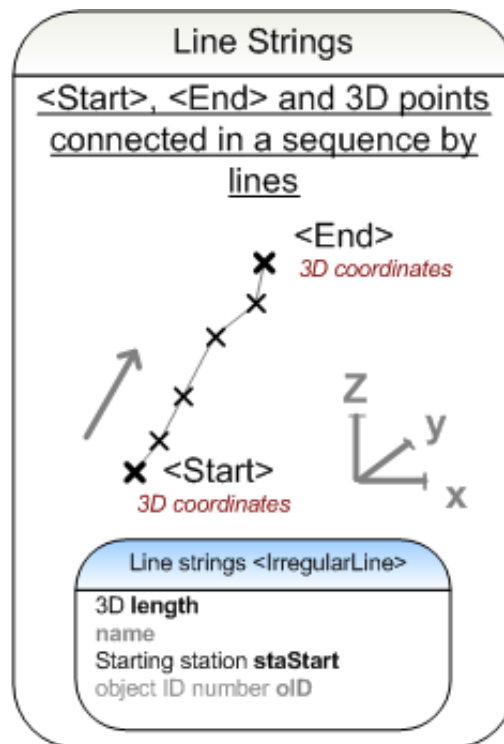


Figure 3.5: Line Strings

3.5.1 Line string

A *line string* has optional attributes and sub-elements to define its **<Start>**, **<End>** and the *intermediate points* either as **<PntList2D>** or **<PntList3D>**.

An *<IrregularLine>* shall have the fields defined in Table 3.12:

Table 3.12: *<IrregularLine>* fields.

Field	Type	Use	Description
One <i><Start></i> element	<i><Start></i>	Required	Represents a 2D or 3D Starting or beginning Point
One <i><End></i> element	<i><End></i>	Required	Represents a 2D or 3D ending Point
One <i><PntList2D></i> or <i><PntList3D></i> element	<i><PntList2D></i> or <i><PntList3D></i>	Required	Point list
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
staStart	<i>double</i>	Optional	Station offset
length	<i>double</i>	Optional	Line length
dir	<i>direction</i>	Optional	Line direction
state	<i>stateType</i>	Optional	State

3.6 String line model

An *alignment group* **<Alignments>** is a collection of geometric alignments and line strings. The string line model of a route is composed of their descriptions in the file, ordered into layers. The string line model used in Inframodel is based on the Leica RoadRunner software.

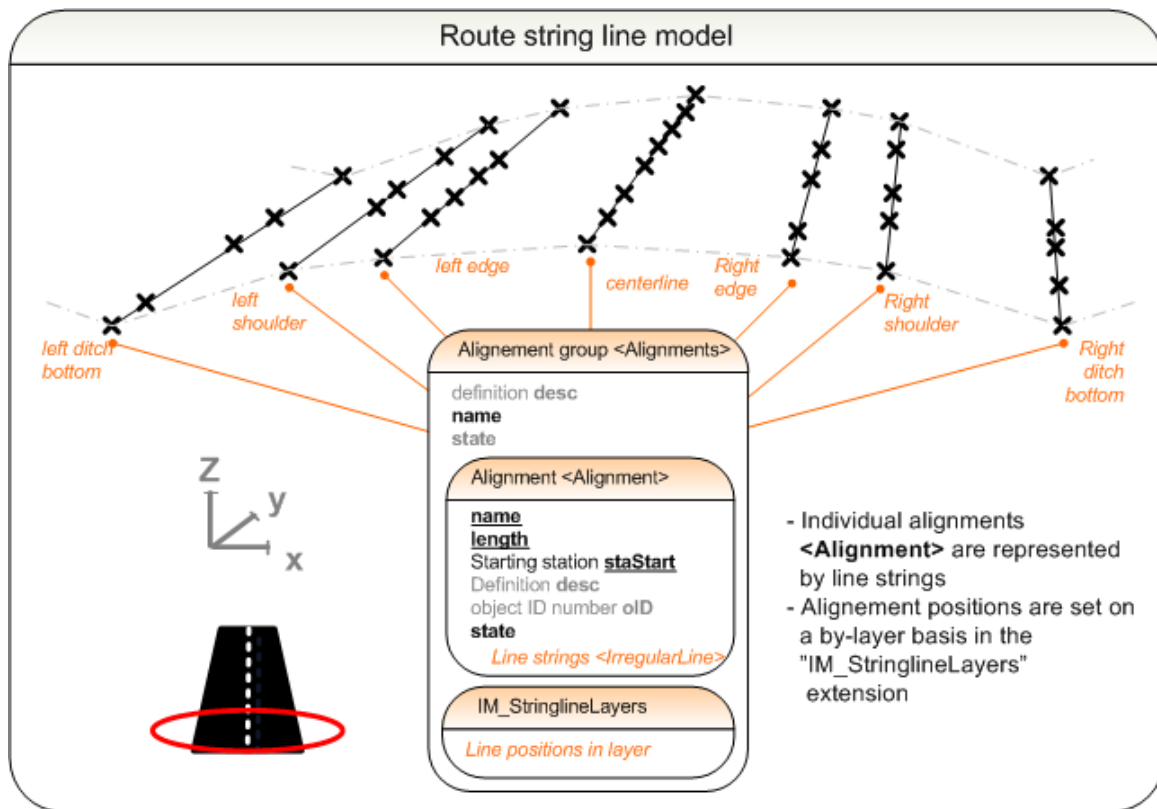


Figure 3.6: String line model

The string line model of a **<Alignment>** is defined by the “IM_stringlineLayers” extension. The string line model consists of individual line strings, whose locations are described layer by layer in the “IM_stringlineLayer” child element. The order of the **<Alignment>** elements is irrelevant, because their unique names are used as alignment identifiers **<Alignment>.name**. Each layer of the string line model is assigned a **unique name** and the **alignments** it contains. It is optional to define a **centerline** and set the *surface codes* **surfaceCoding**.

The procedure for constructing a new layer in the string line model in the “IM_stringlineLayers” extension goes as follows:

1. The layer is assigned a unique **name**.
2. The constituent line string alignments are selected by addressing the line strings by their name **<Alignment>.name** going from the left to the right. The line names are separated by commas.
3. The **centreline** may optionally be set.
4. The **surfaceCoding** and **surfaceCodingDesc** may optionally be set.

A line string may belong to several different layers. It is recommended to describe the layers in order beginning from the topmost layer. The string line model sample below utilizes the general surface coding.

An *IM_stringlineLayers* **<Feature>** shall have the fields defined in Table 3.13:

Table 3.13: IM_stringlineLayers <Feature> fields.

Field	Type	Use	Description
At least one <i>IM_stringlineLayer</i> <Feature> element	<i>IM_stringlineLayer</i> <Feature>	Required	Inframodel stringline layer <Feature> extension
code	<i>string</i>	Required	Fixed value, IM_stringlineLayers
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

An *IM_stringlineLayer* <Feature> shall have the fields defined in Table 3.14:

Table 3.14: IM_stringlineLayer <Feature> fields.

Field	Type	Use	Description
One <i>name</i> <Property> element	<i>name</i> <Property>	Required	Unique name
One <i>alignments</i> <Property> element	<i>alignments</i> <Property>	Required	List of alignments which are used to form the surface (sorted from left to right)
One <i>centreline</i> <Property> element	<i>centreline</i> <Property>	Required	Centre line of stringline layer
Zero or one <i>desc</i> <Property> element	<i>desc</i> <Property>	Optional	Description
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
code	<i>string</i>	Required	Fixed value, IM_stringlineLayer
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

3.7 Terrain model

The *route terrain model* (<Surfaces>) contains a description of the topmost surface (one or more <Surface>) of the route (Digital Elevation Model). It consists of the vertices of the component faces <Pnts> and the faces <Faces> as explained in section 2.2 *Source data*. Also, random points and breaklines of the surface can be described as explained in section 2.2 *Source data*. The route terrain model shall have the same name as the route alignments group, i.e. <Surfaces>.name shall match the corresponding <Alignments>.name

The route terrain model consists of:

- Triangle vertices and
- Faces,
- Random points and
- Breaklines,
- Inframodel type coding.

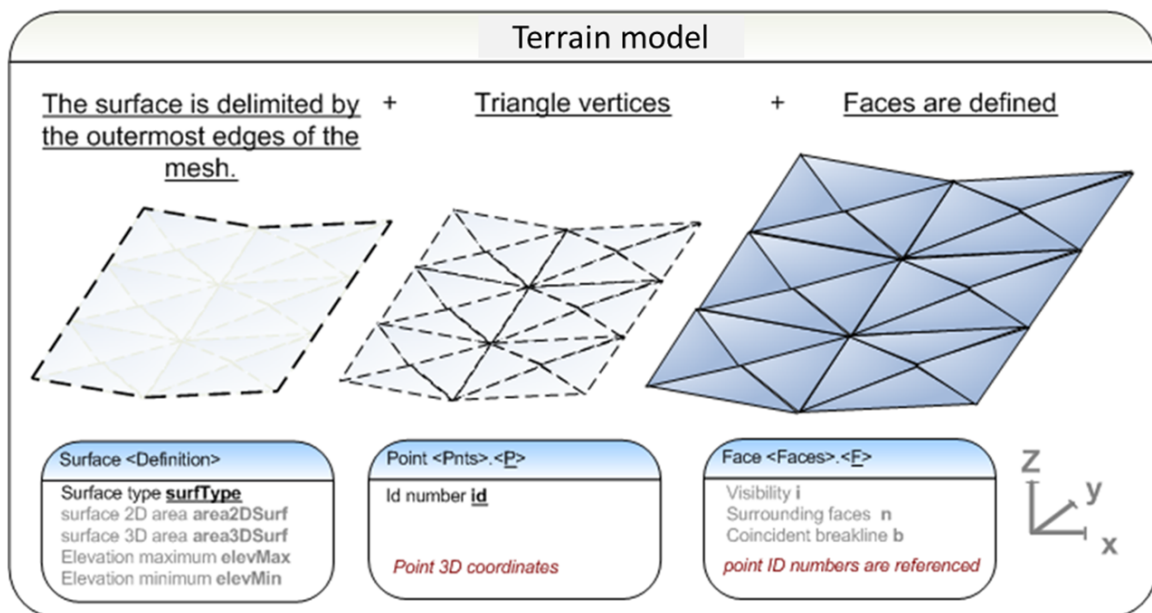


Figure 3.7: Terrain model

Type <Surfaces> is defined per Table 2.1 above.

3.8 Structural model

The structural model of a route contains the surface meshes of all structural layers. When several layers are transferred in the same file, they shall be described in order from top to down, as explained in section 2.5 *Ground layer model*.

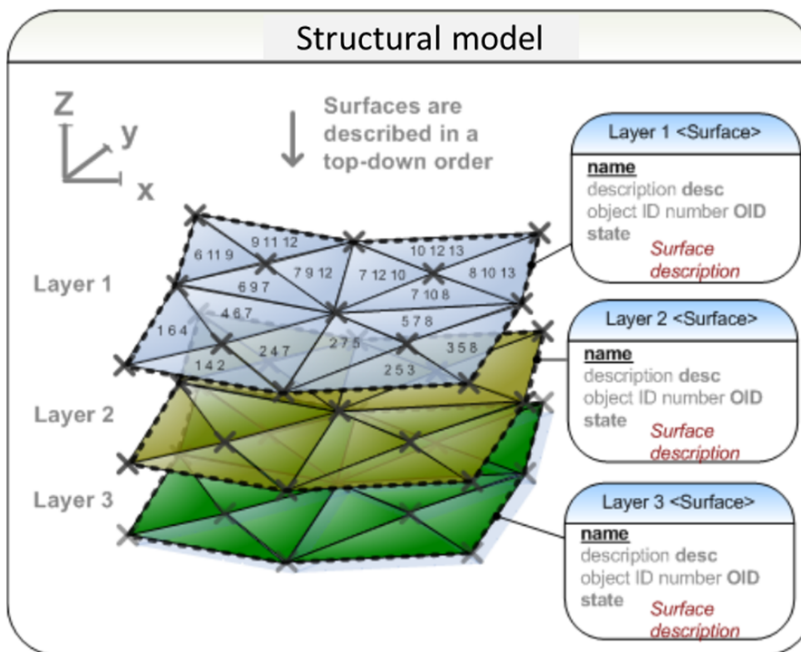


Figure 3.8: Structural model

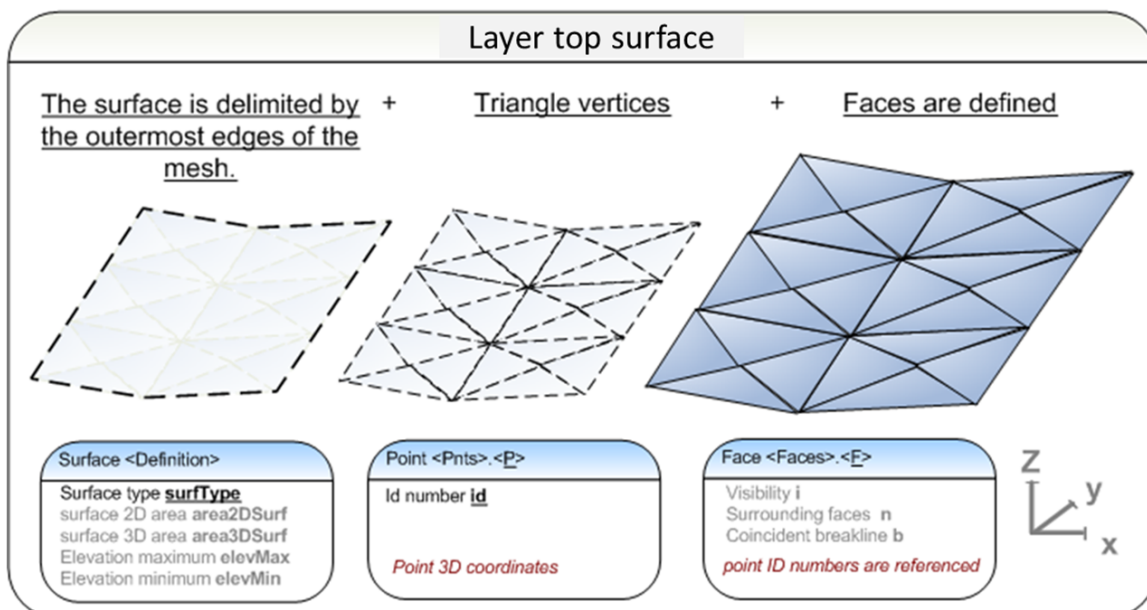


Figure 3.9: Triangulated model

Type <Surfaces> is defined per Table 2.1 above.

3.9 Cross-section parameters

Cross-section parameters refer to parametric information complementing the model represented as surface or stringline models. These include design parameters such as the widths and superelevations of roads.

The relevant cross-section parameters for each route type are described in further detail in the corresponding section of each route type.

4 Road and street design

The file describing a road or street plan contains the header information described in the previous section. In inframodel-compliant file transfers a road or street alignment is defined in according to the process defined in section 3 *Route planning*. A LandXML file may contain plans describing several topics, i.e. the same file may contain e.g. road, street, and railway plans.

The content of the design file is defined in metric units, using an adequate number of decimals for accuracy. For example, when the length unit in use is a meter, the values must be defined to at least six decimal places. Street designs often interface with water supply and sewerage systems, which are described in the same file. The constituent surfaces of the water supply and sewerage design are defined as triangular meshes and the design information in the <PipeNetworks> element.

A road or street design in inframodel may contain:

- individual alignments and their purpose, defined by type coding (geometric alignments and line strings)
- line string model (line string alignments)
- a terrain model (the visible surface)
- a structural model (the visible surface and the structural layers of the road or street)
- cross-section parameters
- road signs and planimetric features

The plan information may optionally contain the following:

- source data (random points and breaklines)
- drainage plan
- railway plan
- waterway plan

The superelevation of the road can be described alongside the cross-section parameters. Cross-section parameters are described at the transition points, when the transition of the value of a parameter begins or ends. Superelevation encompasses the cross slope of the roadway and in case of a street also the sidewalk.

4.1 Geometry of roads and streets

An *alignment group* of road or street **<Alignments>** contains a *geometric description* of a continuous stationing reference **<Alignment>**. The *geometric description* is composed of horizontal and vertical geometric elements. The central *alignments* such as the centerline and left and right edges of a road or the centerline and edges of a sidewalk for a street are described as *geometric alignments* while the rest are described in terms of *line strings*. The alignments described as *geometric descriptions* are also described as *line strings* for use in the *line string model*.

The *alignments* **<Alignment>** of an *alignment group* **<Alignments>** are described before the *line string model* or the *plan information* contained by the extensions “IM_stringlinelayers” and “IM_plan”. The particular order of the alignments **<Alignment>** within the alignment group **<Alignments>** does not matter. The alignment description process is described in further detail in section 3.4 *Geometric alignments*.

The **infraCode** is set in the extension “IM_coding” - this *type coding* for an individual **<Alignment>** describes the purpose of the alignment.

Selected alignments are included in the *line string model* defined by the extension “IM_stringlineLayers”. It is defined in the file after the **<Alignment>** elements and arranges line strings into surfaces using **surfaceCoding**.

4.2 String line model of roads and streets

Once the *alignments* **<Alignment>** of an *alignment group* **<Alignments>** are defined, the string line model of the *alignment group* is defined in the extension “IM_stringlineLayers”. The presentation method resembles a cross-section, a *layer* of the string line model are referred to by their name **<Alignment>.name** and their location is presented as surfaces. It is not always possible to present all line strings contained by the layer in order from left to right, although this is recommended.

The detailed description of the construction process of line string model can be found in section 3.6 *String line model*. The string line model employs the same **infraCoding** system for *line strings* as the **<Alignment>**. The surface codes are set using the **surfaceCoding**.

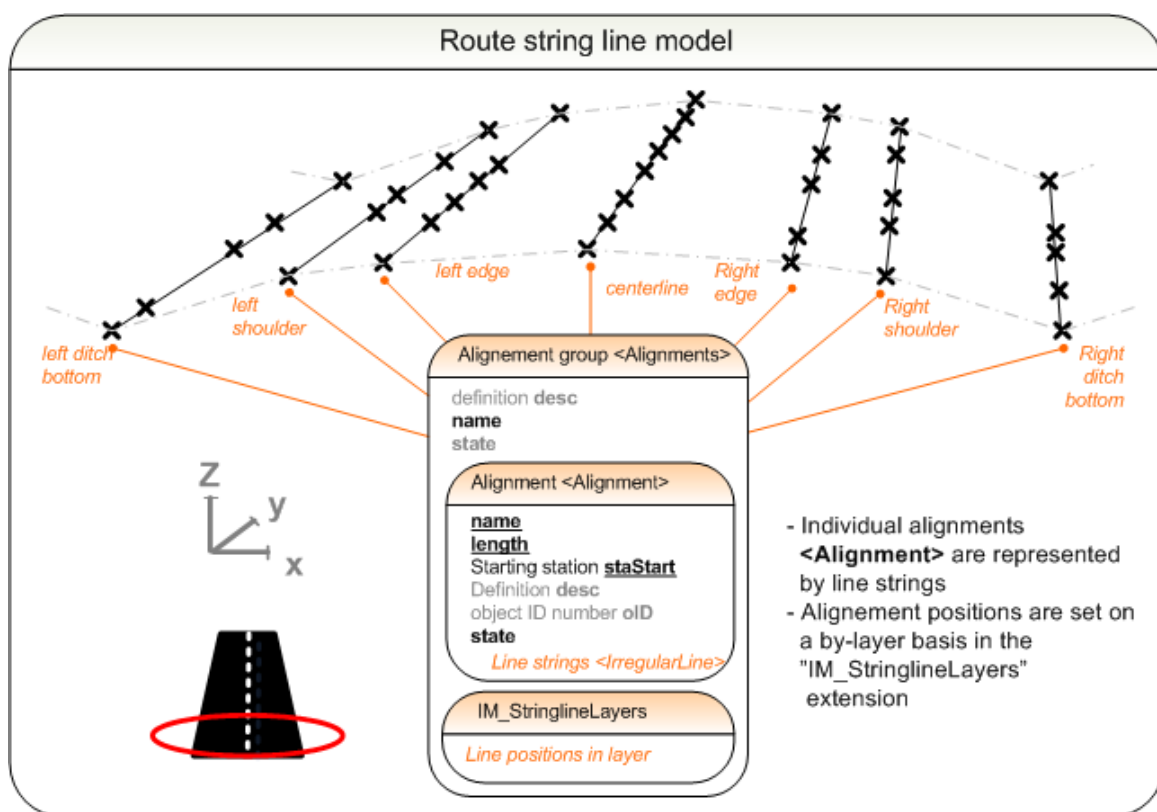


Figure 4.1: Stringline representation of road

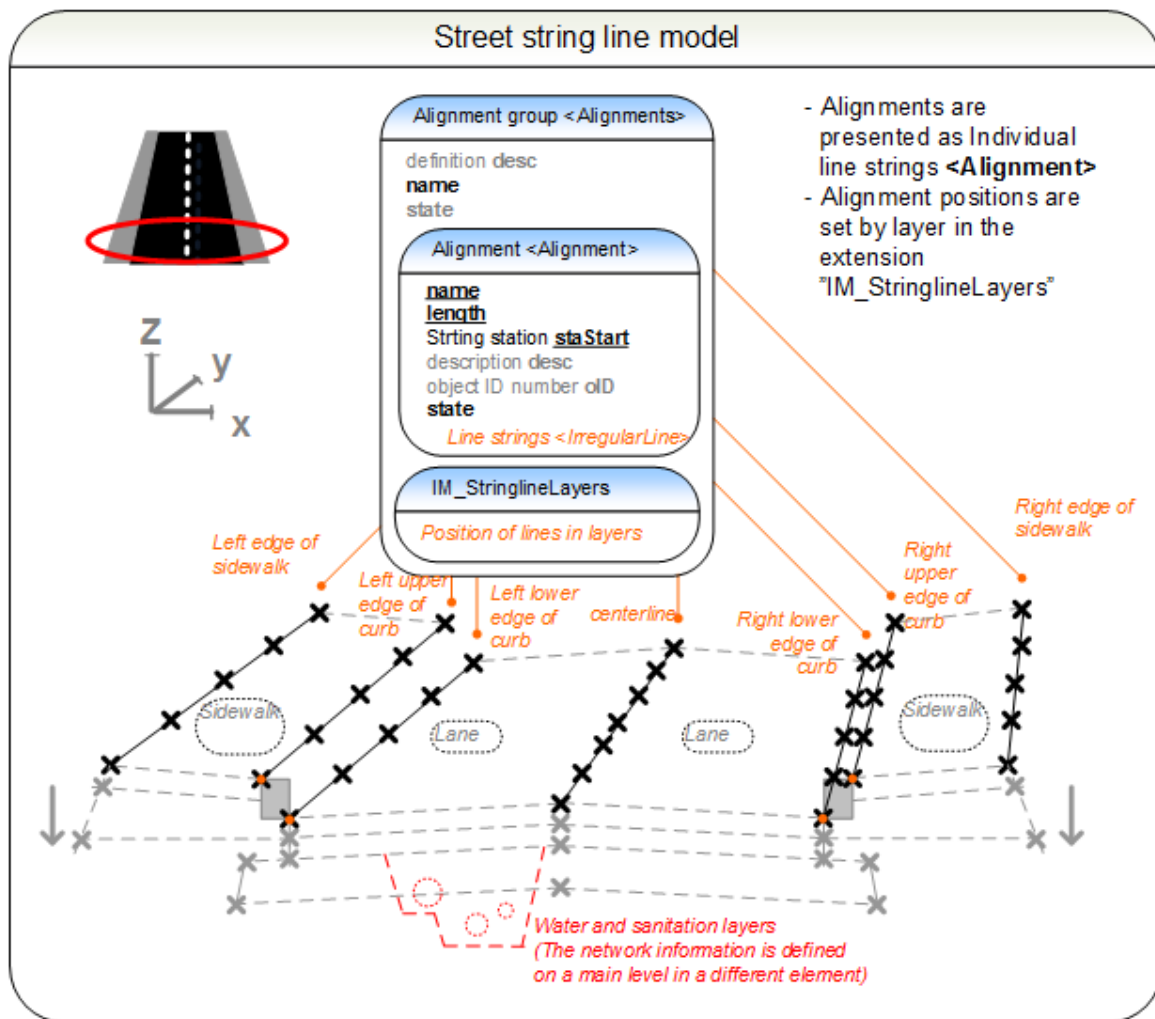


Figure 4.2: Stringline representation of street

4.3 Cross-section parameters of roads and streets

The cross section parameters of an alignment group <Alignments> expand on the geometric description and line string model by defining the cross sections by station intervals in a parametric form. The cross-section parameters are set in inframodel file transfers in the "IM_crossSect" extension along with the cross-section elements. The implementation of the extension is similar to that for waterways.

The cross-section parameters describe the situation at a given station, including the *cross-slopes* of the roadways, streets and sidewalks. The following elaborates on the process of description: the cross-section parameters are presented at a station where a value begins or stops changing. Cross-slopes are set for each lane starting from the left to the right. A positive superelevation indicates a superelevation where the edge located further from the centerline of two is above the inner one. Accordingly, a negative one indicates that the outer edge is below the inner one.

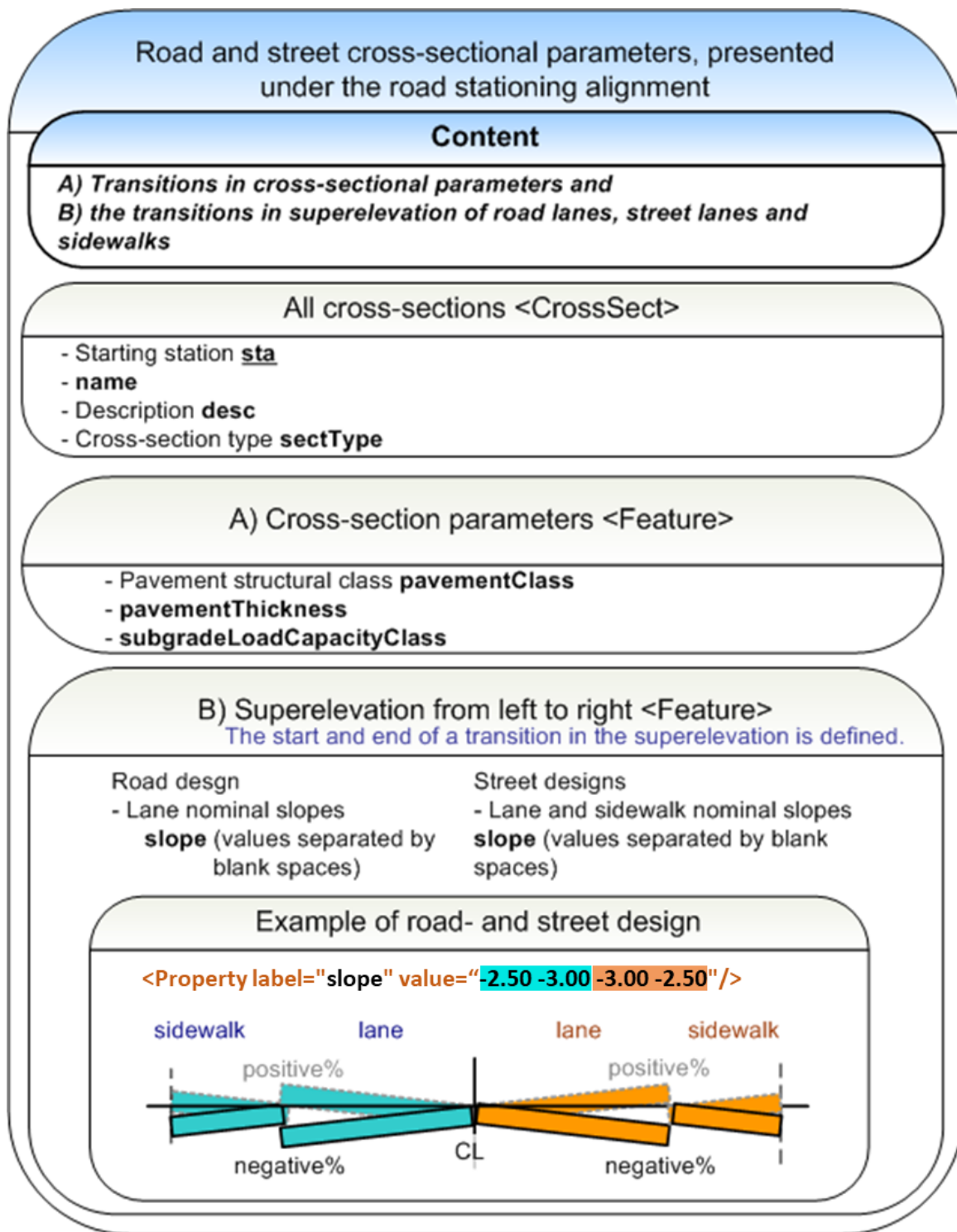


Figure 4.3: Road cross section

4.3.1 Cross-section parameters

The cross-section parameters are set under **<Alignment>.<CrossSects>.<CrossSect>** in the extension “IM_crossSect”.

It is recommended that all parameters are described along with the cross-section. When transitions from one parameter value to another occur, the start and end of the transition are defined. Details on the Finnish road design parameters are provided by Finnish Transport Infrastructure Agency (FTIA).

A *<CrossSect>* shall have the fields defined in Table 4.1:

Table 4.1: *<CrossSect>* fields.

Field	Type	Use	Description
Zero or more <i><DesignCrossSectSurf></i> elements	<i><DesignCrossSectSurf></i>	Optional	Design cross section definition
Zero or one <i>IM_crossSect</i> <i><Feature></i> element	<i>IM_crossSect <Feature></i>	Optional	Inframodel cross section details <i><Feature></i> extension
Zero or one <i>IM_coding</i> <i><Feature></i> element	<i>IM_coding <Feature></i>	Optional	Inframodel type coding <i><Feature></i> extension
Zero or one <i>IM_proprietaryCoding</i> <i><Feature></i> element	<i>IM_proprietaryCoding <Feature></i>	Optional	Inframodel proprietary feature coding <i><Feature></i> extension
Zero or more <i>IM_userDefinedProperties</i> <i><Feature></i> elements	<i>IM_userDefinedProperties <Feature></i>	Optional	Inframodel user defined custom properties <i><Feature></i> extension
sta	<i>double</i>	Required	Cross section station
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description

An *IM_crossSect <Feature>* shall have the fields defined in Table 4.2:

Table 4.2: *IM_crossSect <Feature>* fields.

Field	Type	Use	Description
Zero or one <i>pavementClass</i> <i><Property></i> element	<i>pavementClass <Property></i>	Optional	Pavement class
Zero or one <i>pavementThickness</i> <i><Property></i> element	<i>pavementThickness <Property></i>	Optional	Pavement thickness
Zero or one <i>subgradeLoadCapacityClass</i> <i><Property></i> element	<i>subgradeLoadCapacityClass <Property></i>	Optional	Subgrade load capacity class
Zero or one <i>slope <Property></i> element	<i>slope <Property></i>	Optional	Slope
code	<i>string</i>	Required	Fixed value, <i>IM_crossSect</i>

Field	Type	Use	Description
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

4.3.2 Transitions in superelevation

The superelevation is defined at the transition points, when a transition in the superelevation either begins or end. The cross-slopes are defined along with the cross-section parameters. The following picture illustrates the process in a road design environment.

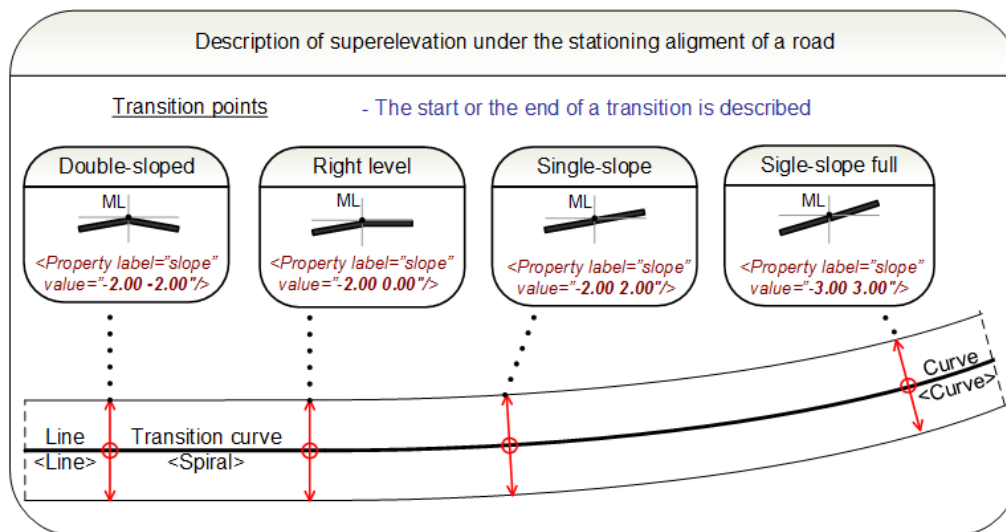


Figure 4.4: Superelevation

4.4 Terrain model and structural model of road or street

The process of constructing a terrain model or structural model is described in detail in section 2.4 *Terrain model* and section 3.8 *Structural model* of a route. The terrain model only contains a triangle mesh of the visible surfaces. The structural model contains all the structure boundaries. All layers in the terrain model and the structural model may be assigned a *type code* (**surfaceCoding**).

It is also possible to attach source data point or breakline information to the meshes, with assigned a *type code* (**terrainCoding**). The process is described in further detail in the section 2.2 *Source data*.

The example illustrations below demonstrate the composition of structural models in road and street design.

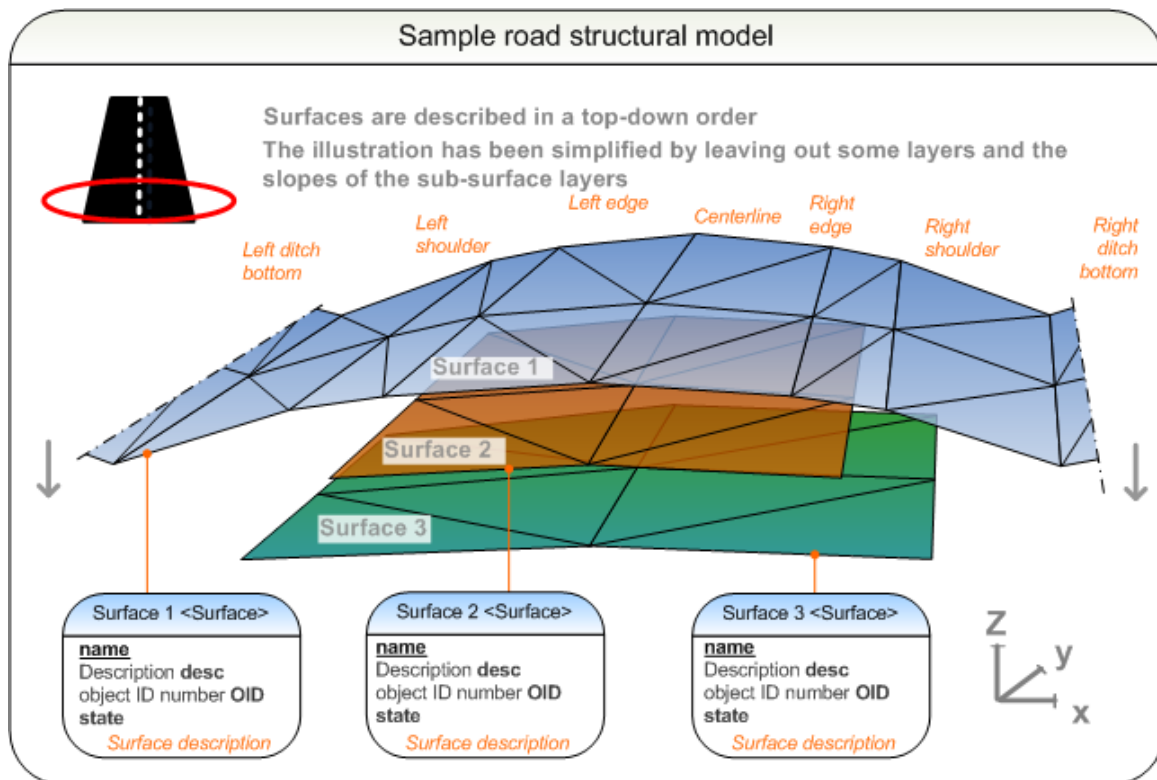


Figure 4.5: Structural model of a road

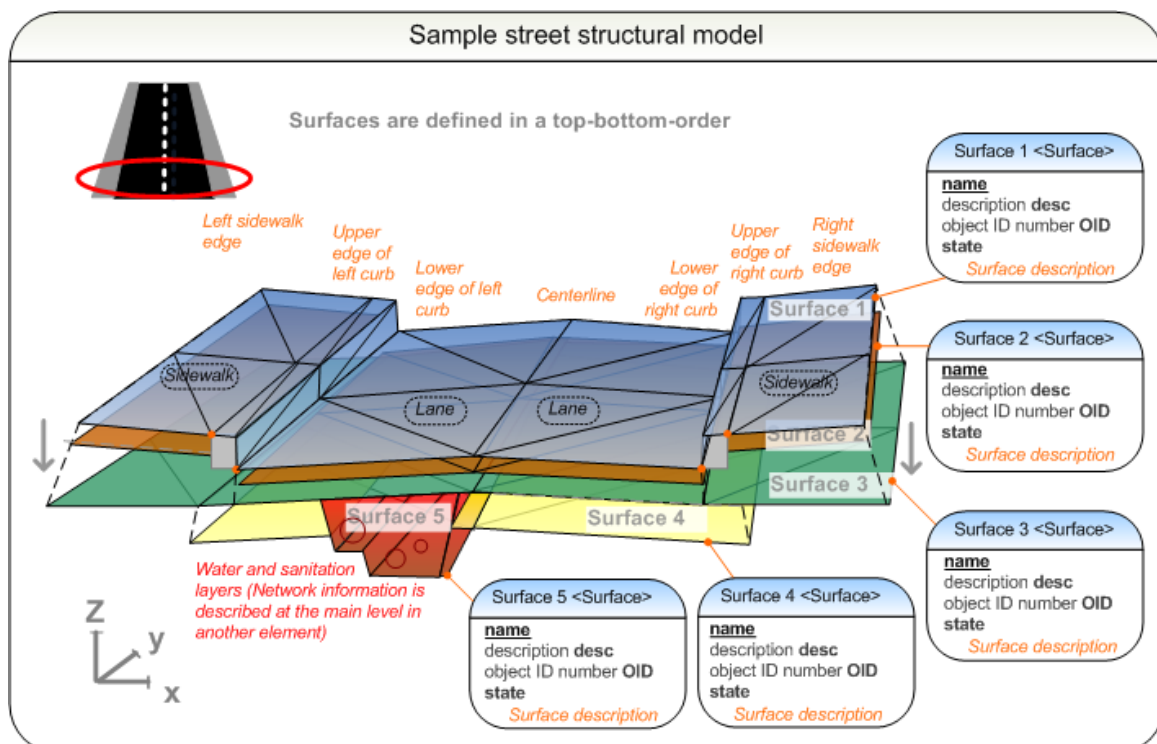


Figure 4.6: Structural model of a street

4.4.1 Structural layers

The material properties of a structural layer between two surfaces are assigned to its top surface, i.e. in a *structural model* of road or street an “IM_structLayer” **<Feature>** extension describes the soil properties below the **<Surface>**.

Details of **<Surface>** in “IM_structLayer” **<Feature>**

Type *IM_structLayer* **<Feature>** is defined per Table 2.14 above.

4.5 Road signs and plan features

The road signs and planimetric features such as fences, guard rails, lightpole or signage footings that are assigned to a particular road or street are described under *roadways*. A *roadways collection* **<Roadways>** may consist of several *roadway* **<Roadway>** elements. Each *roadway* has a reference to its *stationing reference line* **<Alignment>**, and it can hold a number of **<PlanFeature>** elements, as well as **<Roadside>** elements with **<RoadSign>** elements.

Attributes of the *roadways collection* **<Roadways>** are not used in inframodel.

A **<Roadway>** shall have the fields defined in Table 4.3:

Table 4.3: **<Roadway>** fields.

Field	Type	Use	Description
Zero or more <Roadside> elements	<Roadside>	Optional	Roadside, contains road signs
Zero or more <PlanFeature> elements	<PlanFeature>	Optional	A planimetric feature, such as building footprints, guard rails, tree lines, lightpoles or signage.
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Required	Unique name
desc	<i>string</i>	Optional	Description

Field	Type	Use	Description
alignmentRefs	<i>alignmentNameRefs</i>	Required	Reference to <code><Alignment>.name</code> defining the geometry of road,railway or waterway
surfaceRefs	<i>surfaceNameRefs</i>	Optional	Reference to <code><Surface>.name(s)</code> defining the structural model of road,railway or waterway
staStart	<i>station</i>	Optional	Station start
staEnd	<i>station</i>	Optional	Station end
state	<i>stateType</i>	Optional	State

4.5.1 Plan features

The individual *plan features* are each described under **<PlanFeature>**, having a mandatory and unique **name** and optionally **<Location>** and *geometry* as **<CoordGeom>**.

Attributes of **<PlanFeature>**:

A *<PlanFeature>* shall have the fields defined in Table 4.4:

Table 4.4: *<PlanFeature>* fields.

Field	Type	Use	Description
One <i><CoordGeom></i> element	<i><CoordGeom></i>	Required	Geometry
Zero or more <i><Location></i> elements	<i><Location></i>	Optional	Represents a 2D or 3D point location for a PlanFeature.
Zero or one <i>IM_cable <Feature></i> element	<i>IM_cable <Feature></i>	Optional	Inframodel cable properties <i><Feature></i> extension
Zero or one <i>IM_footing <Feature></i> element	<i>IM_footing <Feature></i>	Optional	Inframodel footing <i><Feature></i> extension
Zero or one <i>IM_railing <Feature></i> element	<i>IM_railing <Feature></i>	Optional	Inframodel railing <i><Feature></i> extension
Zero or one <i>IM_fence <Feature></i> element	<i>IM_fence <Feature></i>	Optional	Inframodel fence <i><Feature></i> extension

Field	Type	Use	Description
Zero or one <i>IM_surfaceStructure</i> <Feature> element	<i>IM_surfaceStructure</i> <Feature>	Optional	Inframodel surface structure <Feature> extension
Zero or one <i>IM_planfeature</i> <Feature> element	<i>IM_planfeature</i> <Feature>	Optional	Inframodel generic planimetric feature <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one <i>IM_spatialZone</i> <Feature> element	<i>IM_spatialZone</i> <Feature>	Optional	Inframodel spatial zone <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

<PlanFeature> *geometry* is described in **<CoordGeom>** using **line strings** for linear features, e.g. *cables*, *railings* and *fences*. For point features, such as *footings*, location is given in **<Location>** element as a two or three dimensional point:

Details of **<PlanFeature>** are described as **<Feature>** extension, defined for each type as follows:

1. subsection **11.1.1.1** *Cable* in “IM_cable” extension
2. subsection **11.1.1.2** *Footing* in “IM_footing” extension
3. subsection **11.1.1.3** *Railing* in “IM_railing” extension
4. subsection **11.1.1.4** *Fence* in “IM_fence” extension
5. subsection **11.1.1.5** *Surface structure* in “IM_surfaceStructure” extension
6. subsection **11.1.1.6** *Generic feature* in “IM_planfeature” extension

Additionally, all plan features may be type coded in **<Feature>** using section **14.3** *Type coding* in “IM_coding” extension.

4.5.2 Road signs

The individual *road signs* are each described under **<RoadSign>** (placed under **<RoadSide>**, having no attributes). Mandatory attributes of **<RoadSign>** in Inframodel are **width** and **height**, defining the bound-

ing rectangle around the *road sign*, the actual shape and other properties can be specified “IM_roadSign” extension.

Attributes of **<RoadSign>**:

A *<RoadSign>* shall have the fields defined in Table 4.5:

Table 4.5: *<RoadSign>* fields.

Field	Type	Use	Description
Zero or one <i>IM_roadSign</i> <Feature> element	<i>IM_roadSign</i> <Feature>	Optional	Inframodel generic planimetric feature <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
station	<i>station</i>	Optional	Station number
offset	<i>offsetDistance</i>	Optional	Distance from main alignment
sideofRoad	<i>sideofRoadType</i>	Optional	Side of the road
width	<i>double</i>	Required	Sign width
height	<i>double</i>	Required	Sign height
mountHeight	<i>double</i>	Optional	Mounting height, from edge of roadway surface to sign lower edge

An *IM_roadSign* <Feature> shall have the fields defined in Table 4.6:

Table 4.6: *IM_roadSign* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>signName</i> <Property> element	<i>signName</i> <Property>	Optional	Sign name
Zero or one <i>description</i> <Property> element	<i>description</i> <Property>	Optional	Description

Field	Type	Use	Description
One <i>signCode</i> <Property> element	<i>signCode</i> <Property>	Required	Sign code
Zero or one <i>sizeDesignation</i> <Property> element	<i>sizeDesignation</i> <Property>	Optional	Sign size designation
Zero or one <i>location</i> <Property> element	<i>location</i> <Property>	Optional	Sing location, space delimited 3d coordinate
Zero or one <i>mountType</i> <Property> element	<i>mountType</i> <Property>	Optional	Mounting type
Zero or one <i>material</i> <Property> element	<i>material</i> <Property>	Optional	Sign material
Zero or one <i>reflection</i> <Property> element	<i>reflection</i> <Property>	Optional	Reflection type
Zero or one <i>shape</i> <Property> element	<i>shape</i> <Property>	Optional	Sign shape
Zero or one <i>background</i> <Property> element	<i>background</i> <Property>	Optional	Sign background
Zero or one <i>state</i> <Property> element	<i>state</i> <Property>	Optional	Sign state
Zero or one <i>note</i> <Property> element	<i>note</i> <Property>	Optional	Note
code	<i>string</i>	Required	Fixed value, IM_roadSign
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

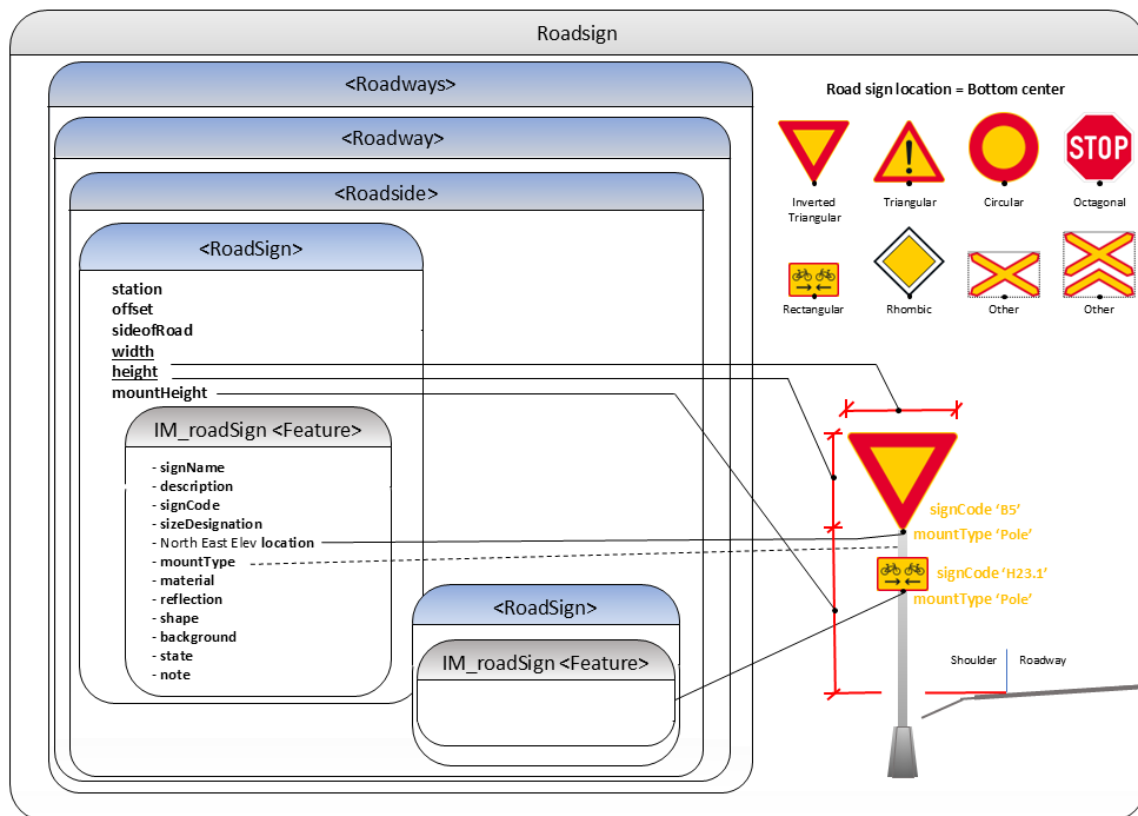


Figure 4.7: Road sign

5 Railway design

The methods used to describe a route in inframodel file transfer are described in detail in the section route planning. It is possible to describe road, street and railway plans and information about water supply and sewerage. A railway plan typically consists of one continuous track for KM-posting and other tracks. The centerlines and the bottoms of the rails of each track are described as geometric alignments. The transition points for cant and design speeds are also described in conjunction with the stationing of the centerlines of the KM-posting track and of other tracks. The current specification does not include description of switches and crossings. The alignments with string line representations can be collected into a string line model of the railway. The model can also include a surface model and structural model, and it is possible to attach additional breakline and random point information.

5.1 Track geometry

One **<Alignments>** collection containing several **<Alignment>** elements is used for describing a railway geometry. Only one of those **<Alignment>** elements can be the centerline of reference track for KM-posting. Description a track centerline is always a geometric **<Alignment>**.

The geometric alignment description is composed of elements in horizontal and vertical geometry, respectively (see section 3.4 *Geometric alignments*). The KM-posting reference track and additional track center lines have their own type codes in inframodel rail planning. Other lines than centerlines and the bottoms of the rails are described as line strings. Since the *string line model* of the track only uses line strings, the *line information* contained by the *geometry lines* is also described as an approximate *line string*. These may be presented in any order. The type coding of individual lines determines their purposes; the type code (terrainCoding) for the terrain model is set in a structural extension. The desired lines can be included in a *string line model*, according to the structural extension “IM_stringlineLayers”. In the stringline model the surfaces can be assigned surface codes (**surfaceCoding**). The *surface and structural models of a railway* can also be described as triangular meshes (TIN surfaces). The railway *plan information* is defined in the optional “IM_plan” structural extension.

5.2 String line model

After a particular **<Alignment>** out of a group of **<Alignments>** has been defined, the stringline model may be defined in the structural extension “IM_stringlineLayers”. The mode of presentation is akin to cross-sections, the stringline layers of the string line model are referred to by their name **<Alignment>.name** and the location is described as surfaces. It is not always possible to present all line strings contained by the layer in order from left to right, although this is recommended.

The detailed description of the construction process of line string model can be found in section 3.6 *String line model*. The lines of the string line model employ the same terrain point coding as alignments. Surfaces are defined in the string line model with surface codes (**surfaceCoding**).

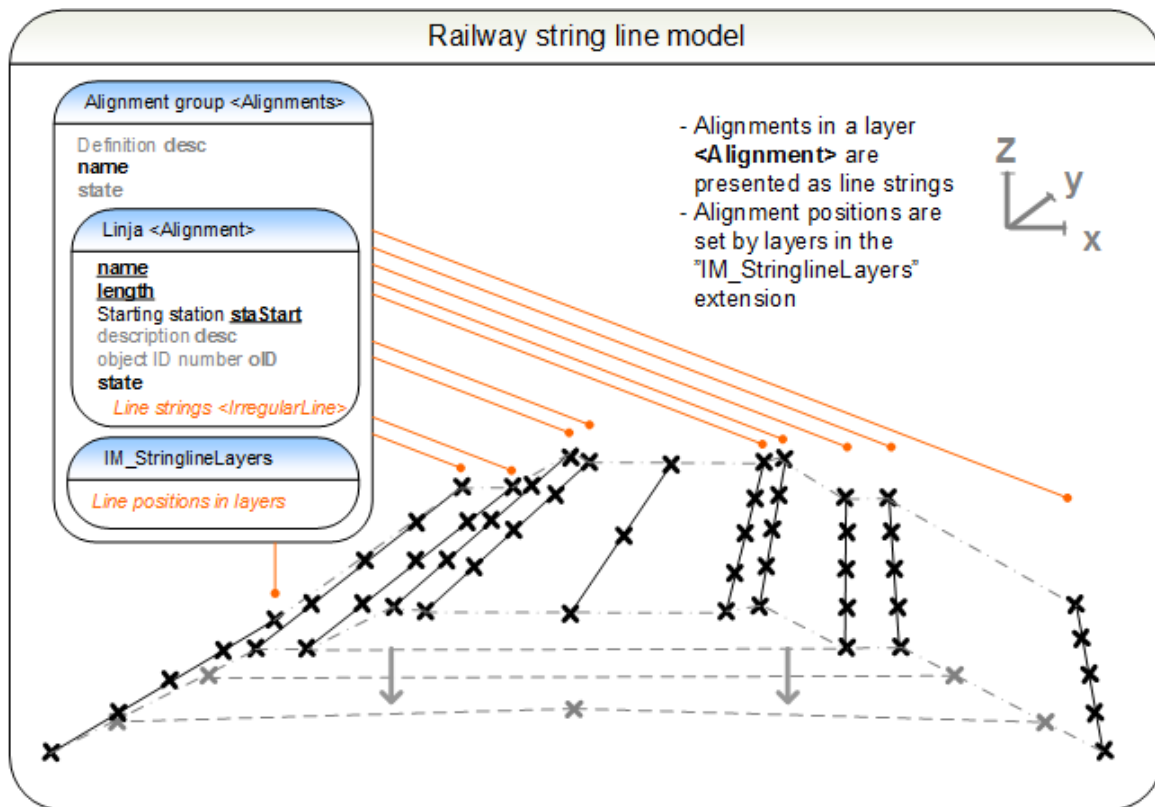


Figure 5.1: String line representation of railway

5.3 KM-posting

The KM-posting method in inframodel uses **<StaEquation>** in a way that somewhat differs from its typical use in LandXML. The individual km-post station distances from the previous post station are defined by their back stations **staBack** (for the first KM-post on a given alignment, the **staBack** can be set to "NaN", when it is not known or relevant). This use of **<StaEquation>** does not imply a re-start in the alignment stationing, hence in inframodel the **staAhead** attribute is always set to the same value as **staInternal** attribute. Individual locations according to the KM-posting system on all tracks are presented in relation to the KM-posting reference alignment. The name of the KM-post station is defined by the **desc** attribute.

Although the Finnish KM-posting system is nominally kilometre based, it cannot be used to define distances between points. The actual length of a track is therefore calculated along the centerline **<Alignment>** of that track, resulting in continuous internal stationing in **staInternal** attribute values.

A **<StaEquation>** shall have the fields defined in Table 5.1:

Table 5.1: **<StaEquation>** fields.

Field	Type	Use	Description
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension

Field	Type	Use	Description
Zero or one <i>IM_kmPostCoords</i> <Feature> element	<i>IM_kmPostCoords</i> <Feature>	Optional	Inframodel KM-post coordinates <Feature> extension
staAhead	double	Required	The “new” station value
staBack	double	Required	Distance to previous km-post station
staInternal	double	Required	Internal stationing
desc	string	Required	Description

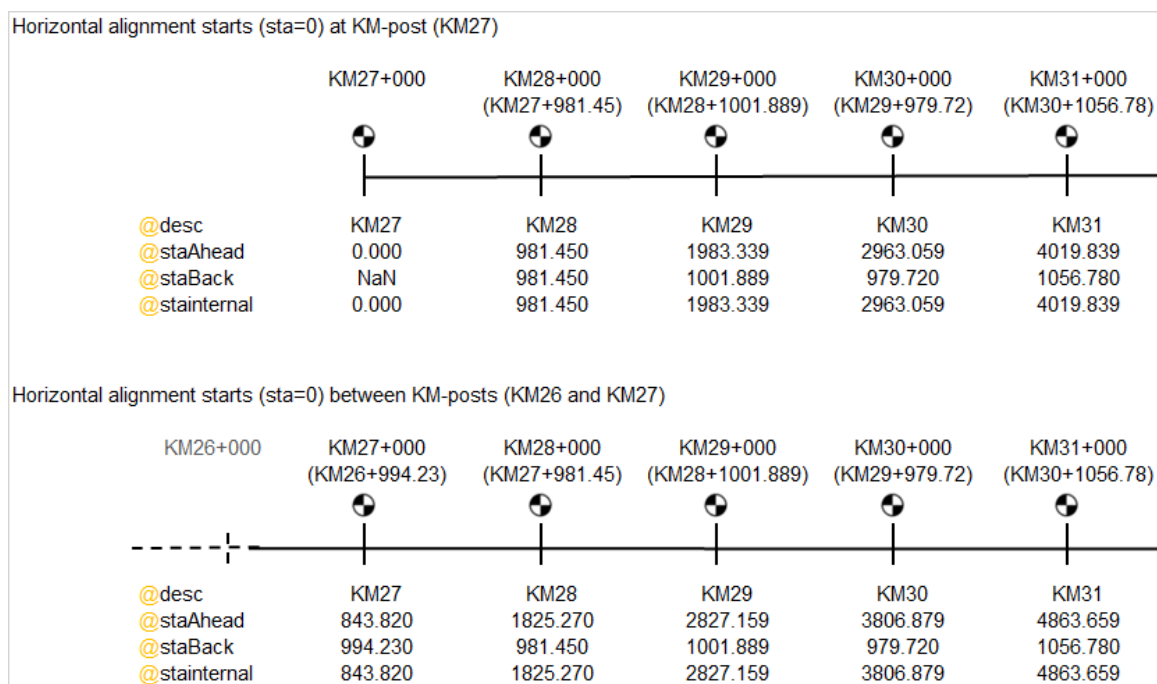


Figure 5.2: KM-posting

An *IM_kmPostCoords* <Feature> shall have the fields defined in Table 5.2:

Table 5.2: *IM_kmPostCoords* <Feature> fields.

Field	Type	Use	Description
One <i>northing</i> <Property> element	<i>northing</i> <Property>	Required	Northing coordinate value in target system in meters
One <i>easting</i> <Property> element	<i>easting</i> <Property>	Required	Easting coordinate value in target system in meters
code	string	Required	Fixed value, <i>IM_kmPostCoords</i>
source	string	Required	Fixed value, <i>inframodel</i>

Field	Type	Use	Description
name	<i>nameType</i>	Optional	Optional unique name

5.4 Cross-sections and track information

Cross-section plan contains information that fleshes out the geometry and string line model descriptions of the railway. The information contained by the cross-section element is valid from the given station forward, either to the next transition cross-section element or the end of the line. The cross-section parameters are described under the KM-posting reference track in the **<Alignment>.<CrossSects>.<CrossSect>** element and its extension “IM_crossSect” (**<Feature>**).

It is recommended that all parameters are described along with the cross-section. Details on the Finnish railway design parameters are provided by Finnish Transport Infrastructure Agency (FTIA).

5.4.1 Cross-sections

The cross-sections for KM-posting reference track are defined by the element **<Alignment>.<CrossSects>.<CrossSect>**. For individual cross-sections, the cross-section parameters are presented in the extension “IM_crossSect”. Transitions are defined in the points where parameters change start and where they have reached their final values after the transition. Triple and higher multiple track railways are composed of double and single track standard cross-sections.

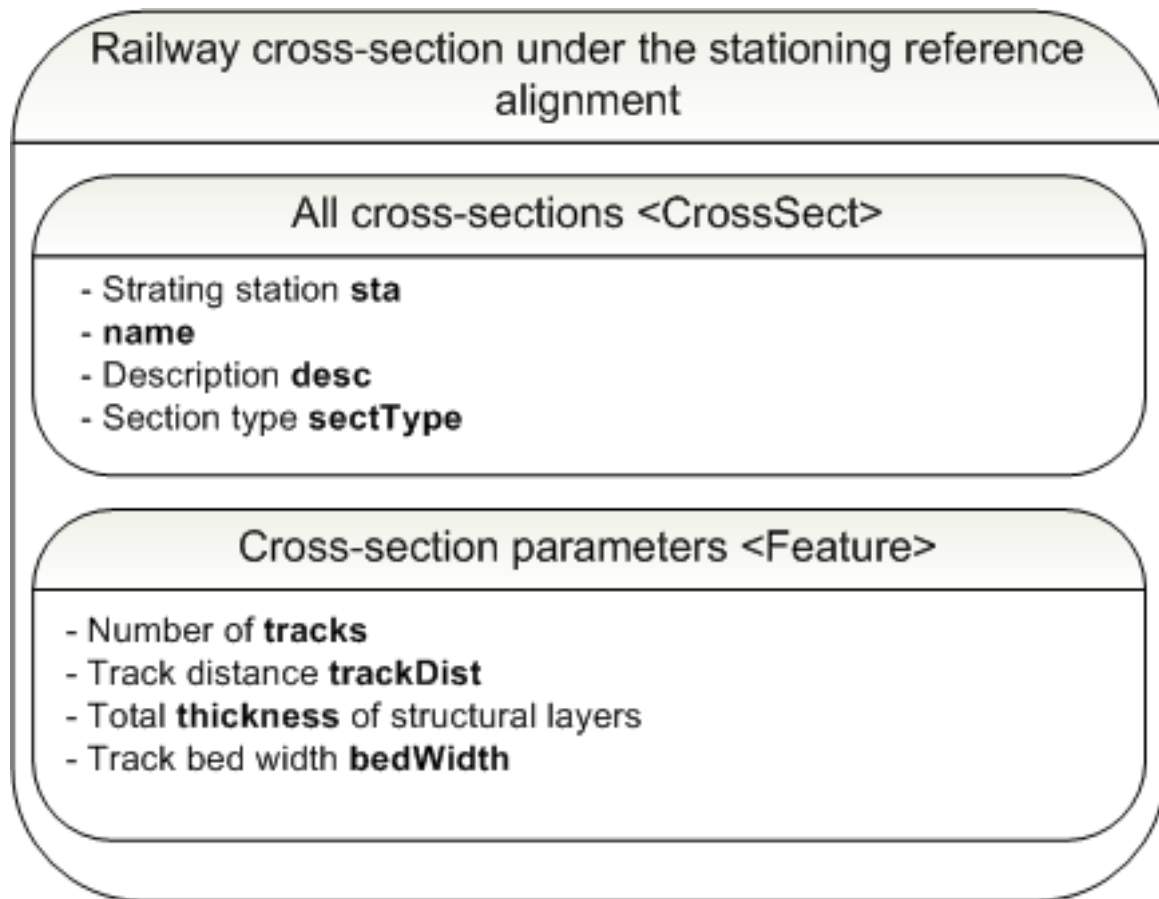


Figure 5.3: Cross section

A <CrossSect> shall have the fields defined in Table 5.3:

Table 5.3: <CrossSect> fields.

Field	Type	Use	Description
Zero or more <DesignCrossSectSurf> elements	<DesignCrossSectSurf>	Optional	Design cross section definition
Zero or one IM_crossSect <Feature> element	IM_crossSect <Feature>	Optional	Inframodel cross section details <Feature> extension
Zero or one IM_coding <Feature> element	IM_coding <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one IM_proprietaryCoding <Feature> element	IM_proprietaryCoding <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension

Field	Type	Use	Description
sta	<i>double</i>	Required	Cross section station
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description

The following information must be defined in the extension “IM_crossSect” for an individual cross-section:

- the number of tracks **tracks**
- the distance between track centerlines **trackDist**
- the total thickness of track bed layers **thickness**
- track bed or cut width **bedWidth**

An *IM_crossSect* <Feature> shall have the fields defined in Table 5.4:

Table 5.4: *IM_crossSect* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>tracks</i> <Property> element	<i>tracks</i> <Property>	Optional	Number of tracks
Zero or one <i>trackDist</i> <Property> element	<i>trackDist</i> <Property>	Optional	Track distance
Zero or one <i>thickness</i> <Property> element	<i>thickness</i> <Property>	Optional	Total thickness
Zero or one <i>bedWidth</i> <Property> element	<i>bedWidth</i> <Property>	Optional	Bed width
code	<i>string</i>	Required	Fixed value, IM_crossSect
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

5.4.2 Track information

The transitions in cant and design speed for each track are described under the track centerline <Alignment> in <Cant> sub-elements. The *track information* <Cant> defines the **name**, the track **gauge** and the *cant rotation point* **rotationPoint**.

A <Cant> shall have the fields defined in Table 5.5:

Table 5.5: <Cant> fields.

Field	Type	Use	Description
Zero or more <CantStation> elements	<CantStation>	Optional	A cant change station.
Zero or more <SpeedStation> elements	<SpeedStation>	Optional	A cant speed-only station.
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Required	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State
equilibriumConstant	<i>double</i>	Optional	Equilibrium constant
appliedCantConstant	<i>double</i>	Optional	Cant constant
gauge	<i>double</i>	Required	Track gauge
rotationPoint	<i>one of (left, center, right)</i>	Optional	Rotation point

The following transitions are described by the track information sub-element:

- **<CantStation>** is used at cant events, typically: 0-value at the start and end of straight track segments, same value at the start and end of circular curve segments, and different values at start and end of transition curve segments (interpolated according to the horizontal spiral curvature change)
- **<SpeedStation>** is used when the design speed changes

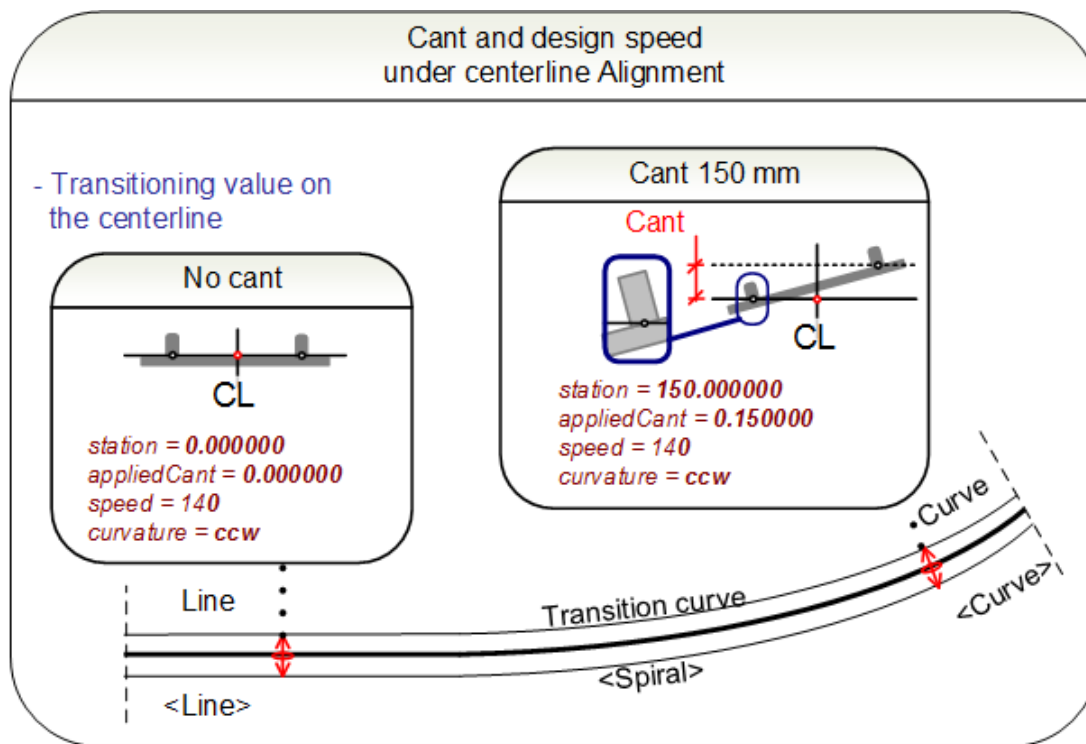


Figure 5.4: Railway cant

1. When the *cant* changes in the **<CantStation>**:

A **<CantStation>** shall have the fields defined in Table 5.6:

Table 5.6: **<CantStation>** fields.

Field	Type	Use	Description
station	double	Required	Cant change station
appliedCant	double	Required	Cant value
speed	double	Optional	Design speed at cant station
curvature	clockwise	Required	Direction of the cant

2. When only the *design speed* changes in the **<SpeedStation>**

A **<SpeedStation>** shall have the fields defined in Table 5.7:

Table 5.7: **<SpeedStation>** fields.

Field	Type	Use	Description
station	double	Required	Design speed change station
speed	double	Optional	Design speed

The **station** attribute of **<CantStation>** and **<SpeedStation>** shall have an stationing value where **<Align-**

ment>.staStart is taken in account.

5.4.3 Switch information

The information on switches of tracks is given under track centerlines in the **<Alignment>.<CoordGeom>.<Line>** using “IM_switch” **<Feature>**

An *IM_switch* **<Feature>** shall have the fields defined in Table 5.8:

Table 5.8: *IM_switch* **<Feature>** fields.

Field	Type	Use	Description
Zero or one <i>switchType</i> <Property> element	<i>switchType</i> <Property>	Optional	Switch type
Zero or one <i>switchHand</i> <Property> element	<i>switchHand</i> <Property>	Optional	Switch hand
Zero or one <i>switchJoint</i> <Property> element	<i>switchJoint</i> <Property>	Optional	Switch joint
code	<i>string</i>	Required	Fixed value, IM_switch
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

5.5 Terrain model and structural model of the track

The presentation method of the *terrain model* is described in further detail in section **2.4 Terrain model**. The terrain model of the track only contains the triangular mesh surface of the visible track structures. The structural model contains all surfaces as described in section **3.8 Structural model**. The goal is to assign all surfaces a type code in accordance to the type coding system.

It is also possible to add source data point and breakline information to surfaces. This is described in further detail in the section section **2.2 Source data**.

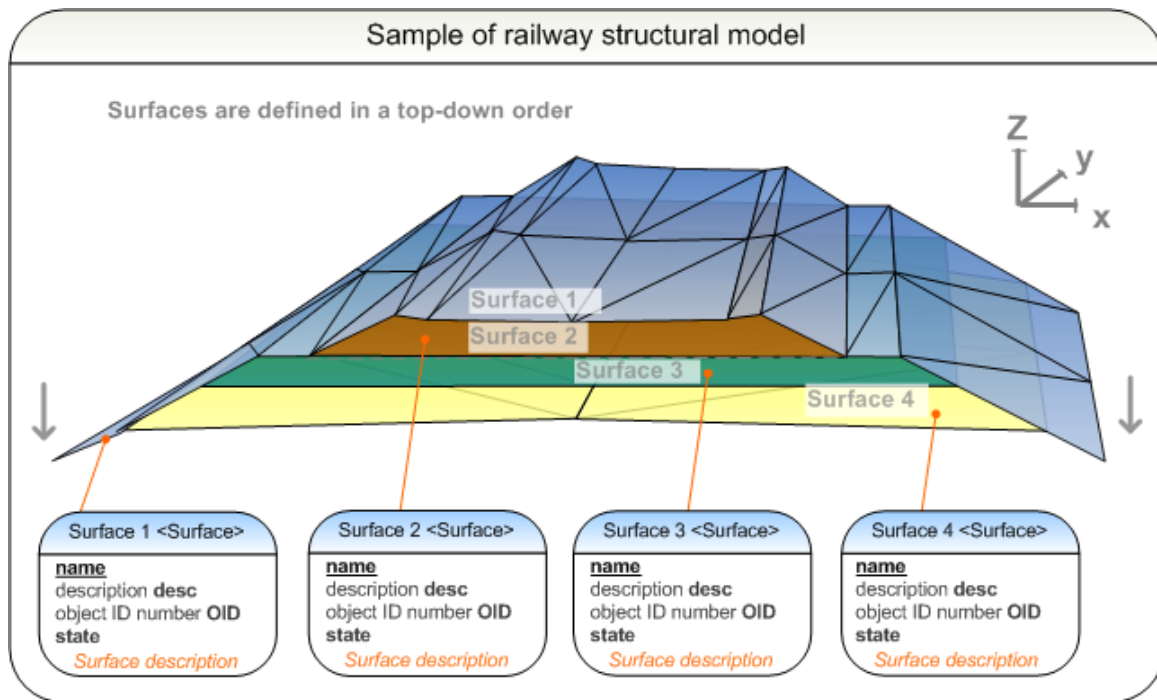


Figure 5.5: Railway structural model

5.6 Rail plan features

The rail planimetric features such as fences, lightpole or signage footings that are assigned to a particular railway are described under *roadways*. A *roadways* collection **<Roadways>** may consist of several *roadway* **<Roadway>** elements. Each *roadway* has a reference to its *centerline* **<Alignment>**, and it can hold a number of **<PlanFeature>** elements.

Attributes of the *roadways* collection **<Roadways>** are not used in inframodel.

A **<Roadways>** shall have the fields defined in Table 5.9:

Table 5.9: **<Roadways>** fields.

Field	Type	Use	Description
At least one <Roadway> element	<Roadway>	Required	(Inframodel)
Zero or one <i>IM_plan</i> <Feature> element	<i>IM_plan</i> <Feature>	Optional	Inframodel plan information <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description

Field	Type	Use	Description
state	<i>stateType</i>	Optional	State

5.6.1 Plan features

The individual *plan features* are each described under **<PlanFeature>**, having a mandatory and unique **name** and optionally **<Location>** and *geometry* as **<CoordGeom>**.

Type **<PlanFeature>** is defined per Table 4.4 above.

<PlanFeature> *geometry* is described in **<CoordGeom>** using line strings for linear features, e.g. *cables*, *railings* and *fences*. For point features, such as *footings*, location is given in **<Location>** element as a two or three dimensional point:

Details of **<PlanFeature>** are described as **<Feature>** extension, defined for each type as follows:

1. subsection **11.1.1.1** *Cable* in “IM_cable” extension
2. subsection **11.1.1.2** *Footing* in “IM_footing” extension
3. subsection **11.1.1.3** *Railing* in “IM_railing” extension
4. subsection **11.1.1.4** *Fence* in “IM_fence” extension
5. subsection **11.1.1.5** *Surface structure* in “IM_surfaceStructure” extension
6. subsection **11.1.1.6** *Generic feature* in “IM_planfeature” extension

Additionally, all plan features may be type coded in **<Feature>** using section **14.3** *Type coding* in “IM_coding” extension.

6 Railway signs

The individual *railway signs* can be described as **<RoadSign>** elements.

See subsection **4.5.2** *Road signs* for details.

7 Waterway design

An inframodel compliant file transfer utilizes the header information defined in section **1** *File headers*. The plan is described as appropriate according to the guidelines laid out in section **3** *Route planning*. All plan information contained in the file utilize a common coordinate, elevation and unit system. The utilized units are metric units appropriate for the file content.

The geometry of a waterway is not described by tangent points in this context. An waterway design contains at least one *alignment group* **<Alignments>** composed of *alignments* **<Alignment>**, a string line model or a surface model and a structural model. Surfaces can have source data point or breakline information attached to them. In addition, the cross-section, minimum depth, minimum width and dimensioning water level height is described.

7.1 Waterway geometry

Inframodel waterways **<Alignments>** must have a continuous stationing reference alignment. This alignment and other central alignments are presented as geometric alignments. In waterway designs, the stationing reference line and the left and right edge alignments are usually described as geometric alignments. A geometric alignment contains information about the horizontal and vertical dimensioning elements of an alignment. Other alignments are described in terms of line strings. The geometric alignments are also given a line string representation for use in the line string model. The process is described in further detail in section **3** *Route planning*.

The alignment group must contain at least one continuous geometric alignment as the stationing reference line. The reference line may be something other than the alignment centerline and its position relative to the centerline may vary. The tangent points are not described in the alignment definition. They can, however, be described as separate *line strings* **<Alignment>**. The elevation of the centerline is usually set to the average lowest height of the sailing season in inland bodies of water.

The order the individual **<Alignment>** elements are described in under the **<Alignments>** element does not matter. The *type coding* of individual *alignments* defines the purpose of the *alignment*. A **terrainCoding**-compliant *terrain code* is set in the extension “IM_coding”. Selected alignments are included in the *line string model*, defined in the extension “IM_stringLineLayers”. The *line string model* consists of alignments structured into surfaces using a **surfaceCoding**. The *surface model* and *structural model* of the design are defined as triangle meshes. The plan information is optionally set in the “IM_plan” extension.

7.2 Cross section parameters

The cross-section parameters of an *alignment group* describe the values of cross-sectional parameters from a the given station onwards. The cross-section parameters are set in the extension “IM_crossSect”. The extension is implemented in a fashion similar to road and street design. The parameters are *minimum depth* of the route, the *minimum width* and the dimensioning **waterLevel**. The dimensioning water level is situation-specific and might be e.g. average daily low or average daily mean water level. The described Cross-section parameters are valid from the set station onwards.

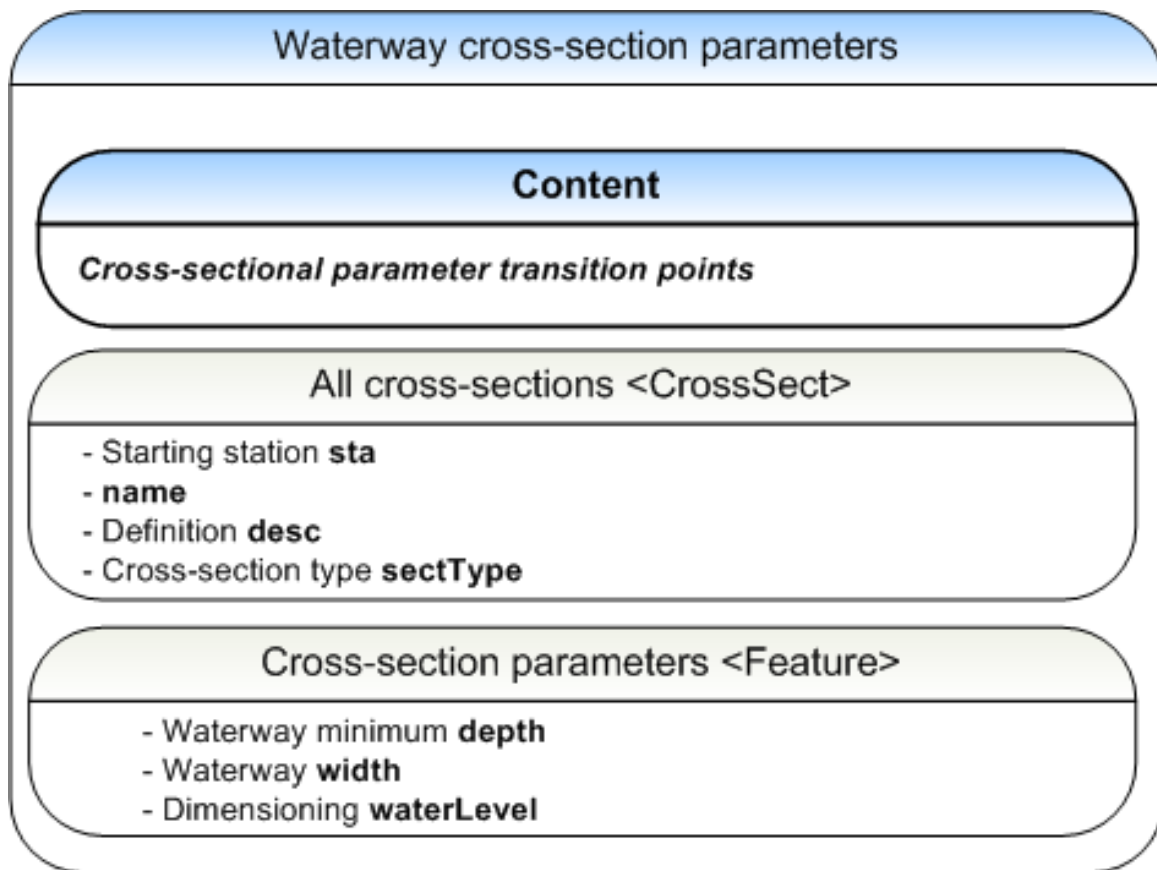


Figure 7.1: Waterway cross sections

The *cross-section parameters* are set for the *stationing reference alignment cross-sections* **<Alignment>.<CrossSects>.<CrossSect>** in the “IM_crossSects” extension **<Feature>**. The first cross-section of the alignment is defined by describing all parameters of the cross-section. The parameters of the following cross-sections are only described if a value begins or stops changing.

A **<CrossSect>** shall have the fields defined in Table 7.1:

Table 7.1: **<CrossSect>** fields.

Field	Type	Use	Description
Zero or more <DesignCrossSectSurf> elements	<DesignCrossSectSurf>	Optional	Design cross section definition
Zero or one <i>IM_crossSect</i> <Feature> element	<i>IM_crossSect</i> <Feature>	Optional	Inframodel cross section details <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension

Field	Type	Use	Description
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
sta	<i>double</i>	Required	Cross section station
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description

An *IM_crossSect* <Feature> shall have the fields defined in Table 7.2:

Table 7.2: *IM_crossSect* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>depth</i> <Property> element	<i>depth</i> <Property>	Optional	Waterway minimum depth
Zero or one <i>width</i> <Property> element	<i>width</i> <Property>	Optional	Waterway width
Zero or one <i>waterLevel</i> <Property> element	<i>waterLevel</i> <Property>	Optional	Dimensioning water level
code	<i>string</i>	Required	Fixed value, <i>IM_crossSect</i>
source	<i>string</i>	Required	Fixed value, <i>inframodel</i>
name	<i>nameType</i>	Optional	Optional unique name

7.3 String line model

The string line model of an *alignment group* <Alignments> is defined in the extension “*IM_stringlineLayers*” after the *alignments* <Alignment>. A line string layer is presented by referring to the names of its constituent *alignments* <Alignment>.name.

The *line string model* of routes is described in further detail in section 3.6 *String line model* The constituent *line strings* <Alignment> of the *string line model* adhere to the *terrain codes* (**terrainCoding**) defined for the line strings. The layers of the line string model are defined by assigning them a *surface code*.

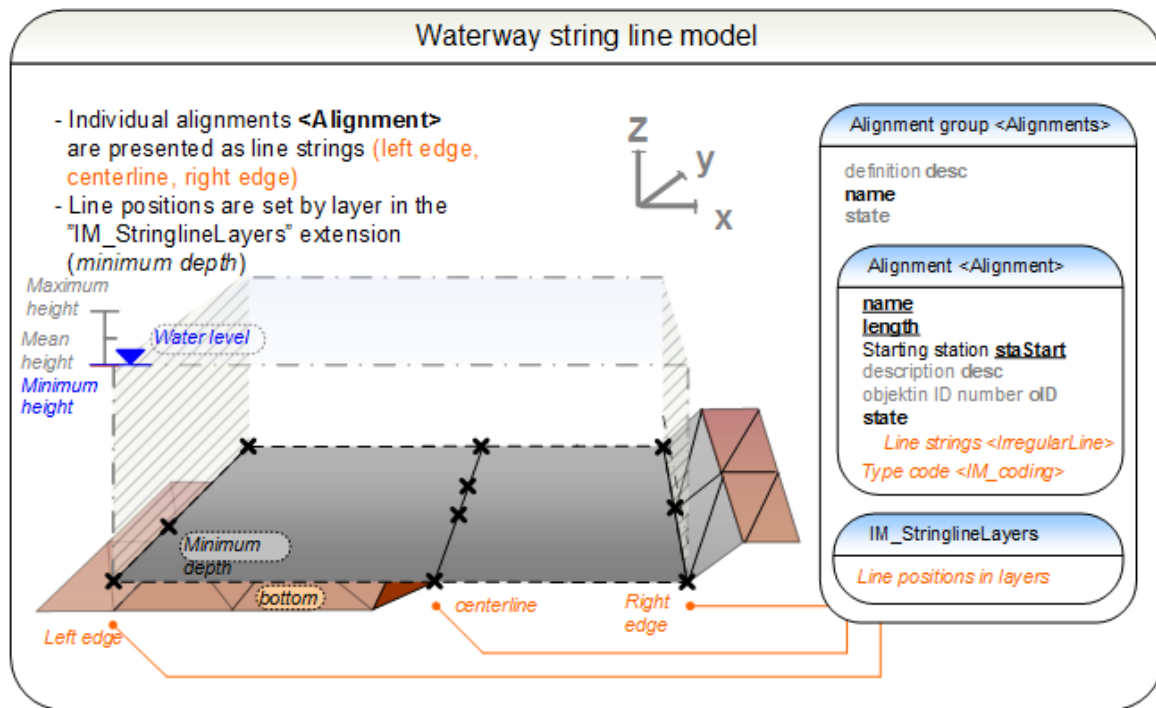


Figure 7.2: Waterway string line representation

7.4 Surface and structural model

The structural model of a waterway is described in detail in section **3.8 Structural model**. The surface model contains at least one triangular mesh that describes the bottom of the waterway. The structural model contain at least the water level in addition to this. It is possible to define *surface codes* (**surfaceCoding**) for all surfaces.

It is possible to attach source data point and breakline information to surfaces. The process is described in further detail in section **2.2 Source data**.

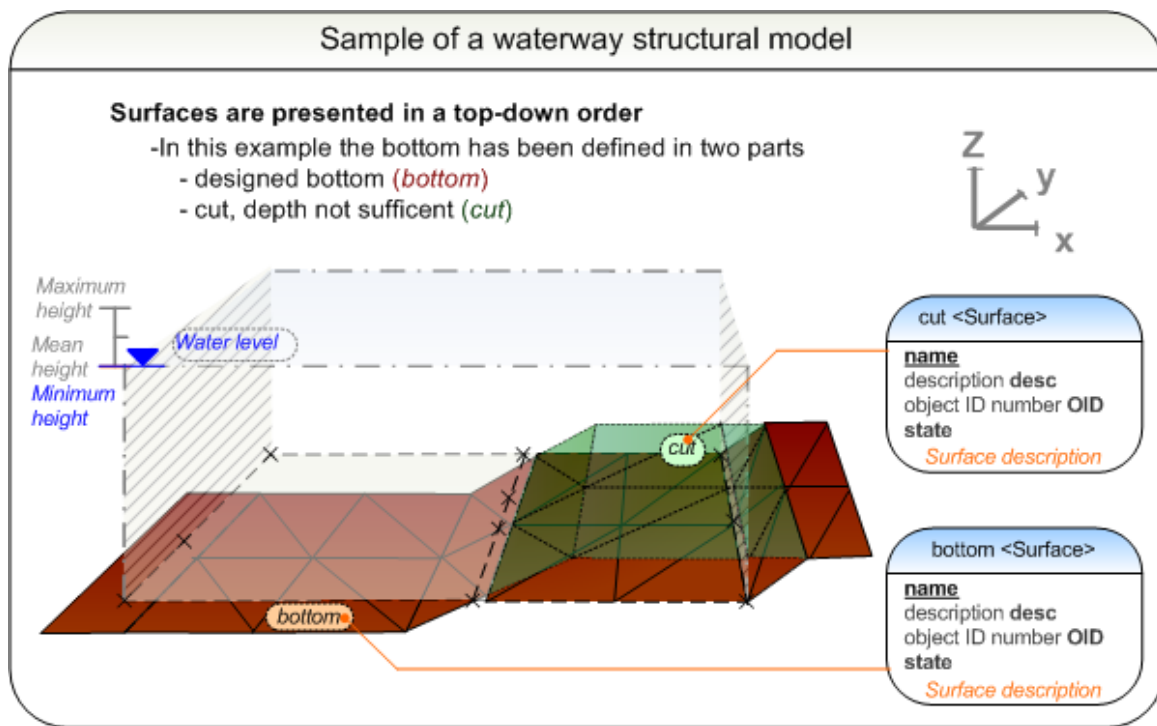


Figure 7.3: Waterway structural model

7.5 Waterway plan features

The waterway planimetric features such as guard rails, lightpole or signage footings that are assigned to a particular waterway are described under *roadways*. A *roadways collection* **<Roadways>** may consist of several *roadway* **<Roadway>** elements. Each *roadway* has a reference to its *stationing reference line* **<Alignment>**, and it can hold a number of **<PlanFeatures>**.

Type **<Roadway>** is defined per Table 4.3 above.

7.5.1 Plan features

The individual *plan features* are each described under **<PlanFeature>**, having a mandatory and unique **name** and optionally **<Location>** and *geometry* as **<CoordGeom>**.

Type **<PlanFeature>** is defined per Table 4.4 above.

<PlanFeature> *geometry* is described in **<CoordGeom>** using **line strings** for linear features, e.g. *cables*, *railings* and *fences*. For point features, such as *footings*, location is given in **<Location>** element as a two or three dimensional point:

Details of **<PlanFeature>** are described as **<Feature>** extension, defined for each type as follows:

1. subsection **11.1.1.1** *Cable* in “IM_cable” extension
2. subsection **11.1.1.2** *Footing* in “IM_footing” extension
3. subsection **11.1.1.3** *Railing* in “IM_railing” extension
4. subsection **11.1.1.4** *Fence* in “IM_fence” extension
5. subsection **11.1.1.5** *Surface structure* in “IM_surfaceStructure” extension
6. subsection **11.1.1.6** *Generic feature* in “IM_planfeature” extension

Additionally, all plan features may be type coded in **<Feature>** using section **14.3** *Type coding* in “IM_coding” extension.

8 Waterway signs

The individual *waterway signs* can be described as **<RoadSign>** elements.
See subsection **4.5.2** *Road signs* for details.

9 Area structures

Area structures encompass descriptions of surfaces and boundaries that complement routes. Such descriptions are for e.g. landscaping, noise barriers and geostructures. The file in which these structures are described contains the header information presented in section 1 *File headers*. Area structures are described as *defined areas* **<Parcel>** with **<CoordGeom>**, or as *surfaces* **<Surface>**. The surface description consist of triangular meshes as explained in section 2 *Base data*, where it is also possible to attach random points **<DataPoints>** and breaklines **<BreakLines>** to the surfaces.

9.1 Landscaping

Landscaping encompasses the areas surrounding a design entity. The description consists of a surface description, which is described in further detail in section 2.4 *Terrain model*. The surface description consists of a triangle mesh, which may have random points or breaklines attached to it. Structural surfaces or cross-sections are usually not defined for landscaping.

The landscaping *plan information* is defined in the optional extension “IM_plan”. A *surface code* (**SurfaceCoding**) is set for the surfaces. It is also possible to set an alternative type coding (**proprietaryInfraCoding**).

Contents of landscaping:

- surface model
- structural model

Optional surface information

- random points **<DataPoints>**
- breaklines **<BreakLines>**

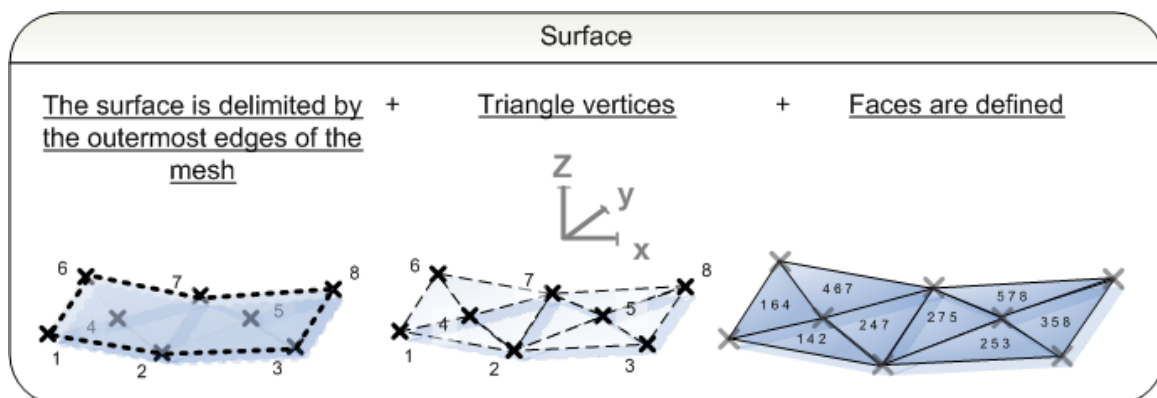


Figure 9.1: Surface model

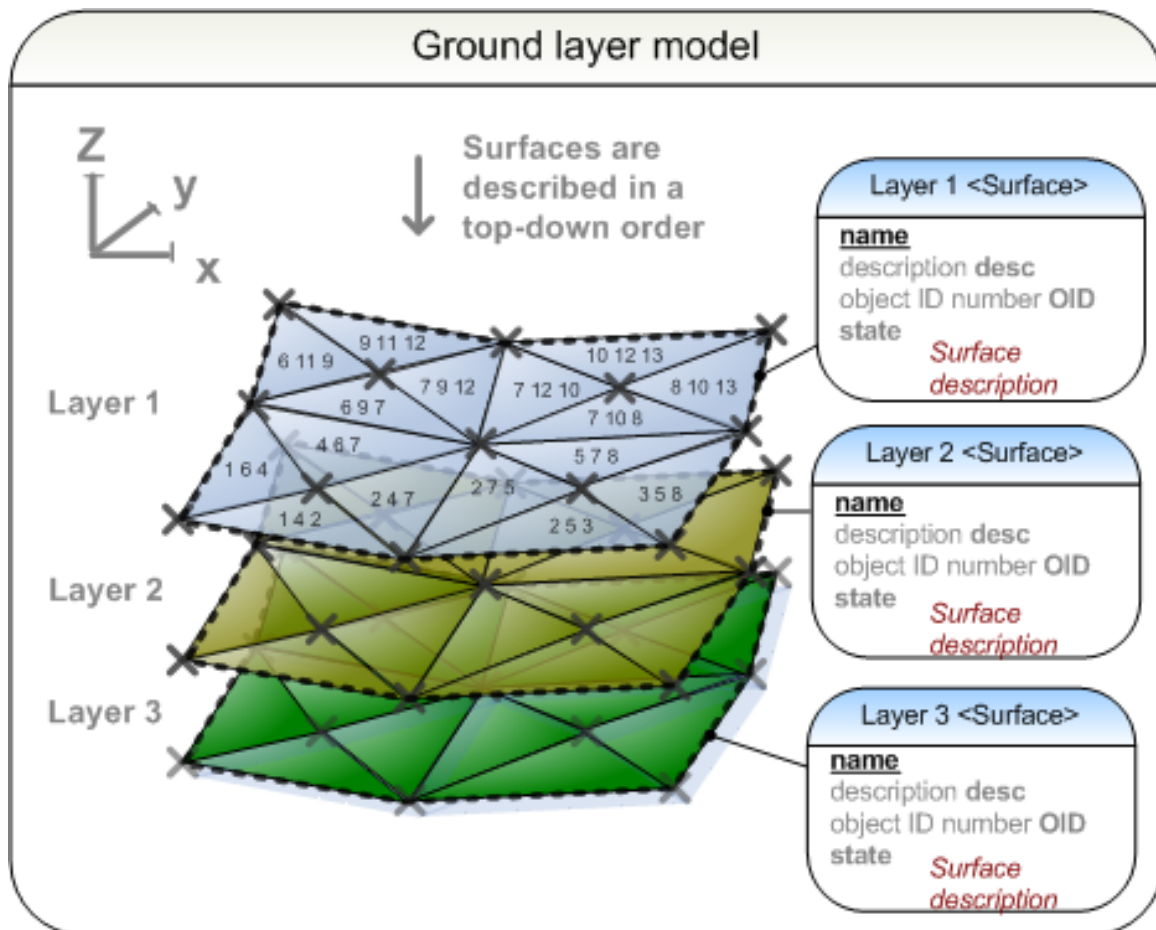


Figure 9.2: Ground layer model

9.2 Noise barriers

Noise barriers are a central part of route planing. The process of describing the components is described in detail in section **2.4 Terrain model**. The The surface description consists of a triangle mesh, which may have random points or breaklines attached to it. Structural surfaces or cross-sections are usually not defined for noise barriers.

The noise barrier *plan information* is defined in the optional extension “IM_plan”. A *surface code* (**SurfaceCoding**) is set for the surfaces . It is also possible to set an alternative type coding (**proprietaryInfraCoding**).

Contents of noise barrier:

- surface model
- structural model

Optional surface information

- random points <**DataPoints**>
- breaklines <**BreakLines**>

9.3 Geostructures

Geostructures refer to area-like structures that can be easily described in terms of surfaces. Examples include stacking of excess mass and subgrade reinforcement, which are possible to describe as surfaces such as described in section 2.5 *Ground layer model*.

The process of describing the components is described in detail in section 2.4 *Terrain model*. The surface description consists of a triangle mesh, which may have random points or breaklines attached to it. Structural surfaces or cross-sections are usually not defined for geostructures.

The geostructure *plan information* is defined in the optional extension “IM_plan”. A *surface code* (**SurfaceCoding**) is set for the surfaces. It is also possible to set an alternative type coding (**proprietaryInfraCoding**).

Contents of geostructures:

- surface model
- structural model
- Optional surface information
- random points
- breaklines

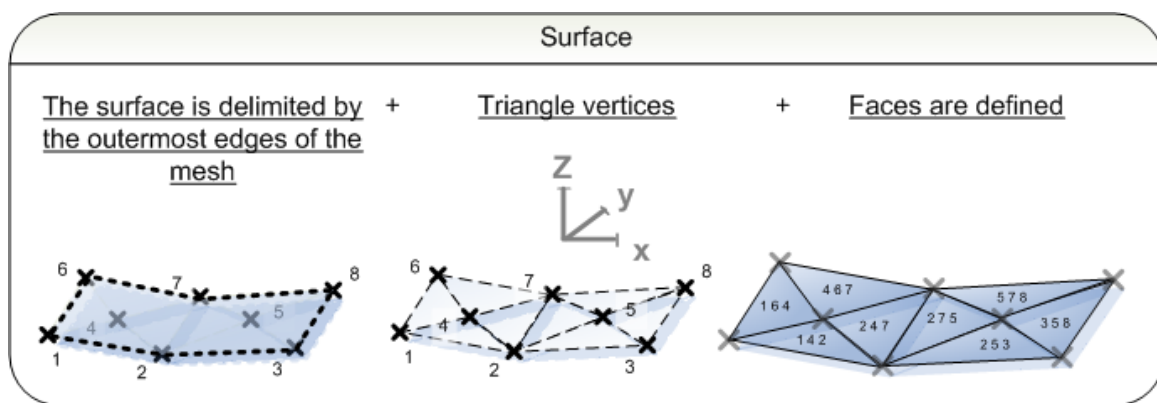


Figure 9.3: Surface model

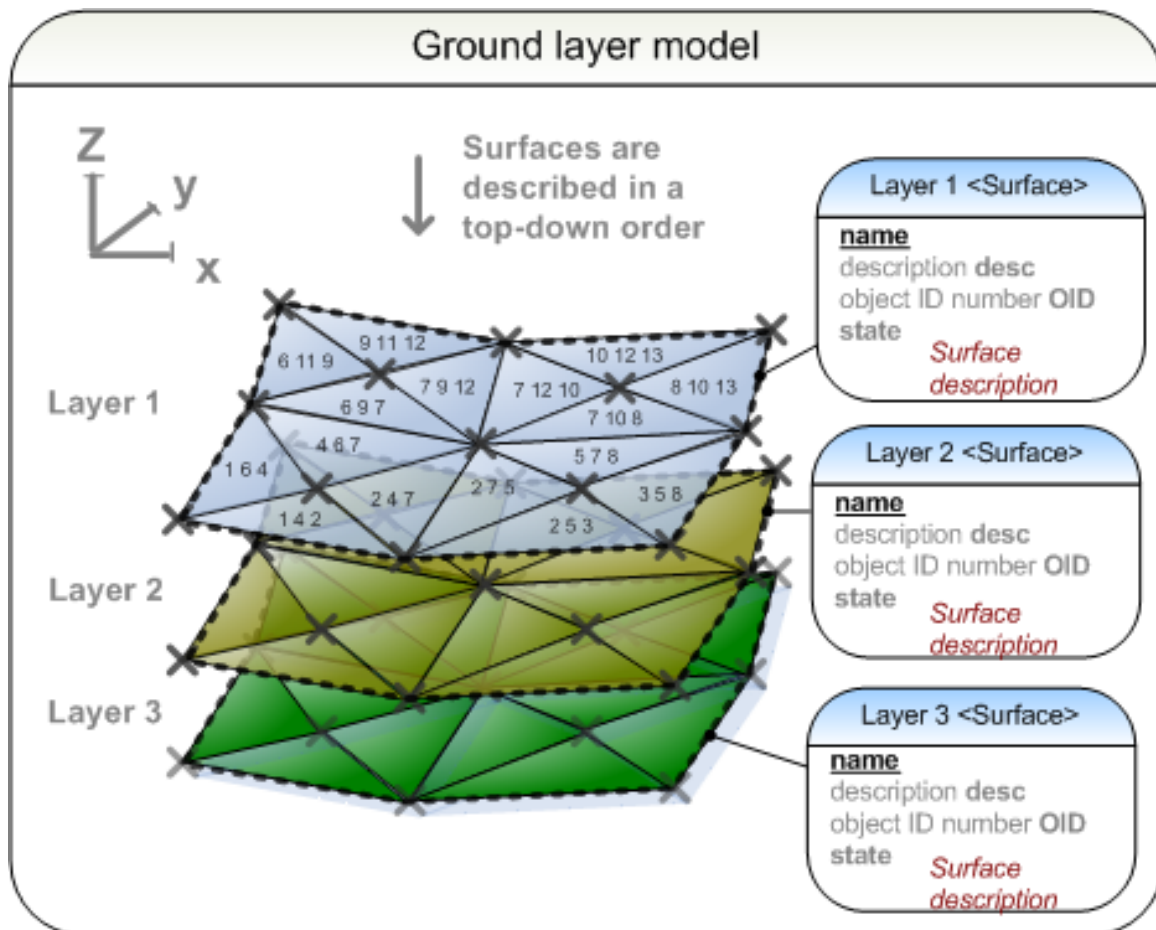


Figure 9.4: Ground layer model

9.4 Surface structures

Surface structures potentially span across multiple routes (such as urban street surface plans), or need not be associated with any route plan. These structures are described as *defined areas* under **<Parcels>** collection, having one or several **<Parcel>** elements. Optional *plan information* is defined in the extension “IM_plan” for **<Parcels>** collection, which may also be given a unique **name**. Each **<Parcel>** shall have a unique **name**, and either its location as **<Center>** or its boundaries as **<CoordGeom>**.

The properties of *surface structure* are defined in “IM_surfaceStructure”:

Type *IM_surfaceStructure* **<Feature>** is defined per Table 2.13 above.

Surface structure may also have calculated area or volume quantities assigned as “IM_quantity” extension:

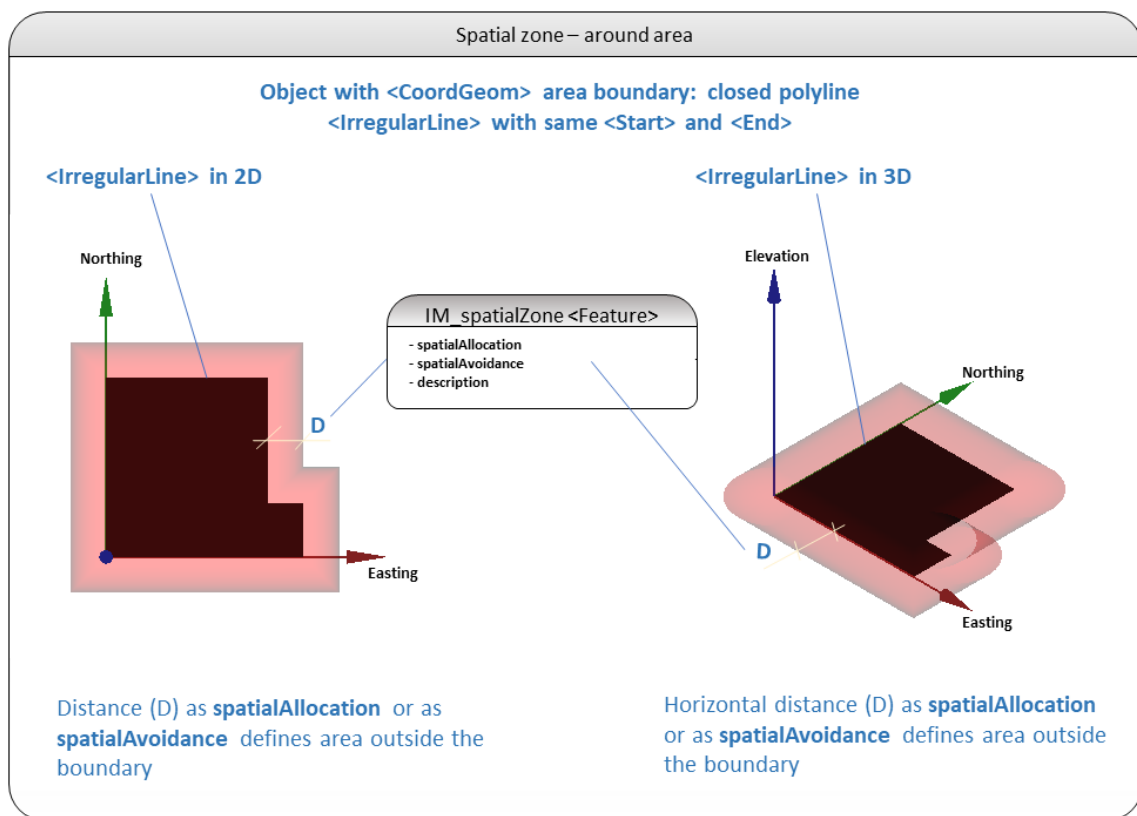
Type *IM_quantity* **<Feature>** is defined per Table 2.5 above.

An area defined as a **<Parcel>** may have an adjoining area spatial allocation or spatial avoidance (or both). These can be described using “IM_spatialZone” extension:

An *IM_spatialZone* **<Feature>** shall have the fields defined in Table 9.1:

Table 9.1: IM_spatialZone <Feature> fields.

Field	Type	Use	Description
Zero or one <i>description</i> <Property> element	<i>description</i> <Property>	Optional	Spatial zone description
Zero or one <i>spatialAllocation</i> <Property> element	<i>spatialAllocation</i> <Property>	Optional	Spatial allocation
Zero or one <i>spatialAvoidance</i> <Property> element	<i>spatialAvoidance</i> <Property>	Optional	Spatial avoidance
code	<i>string</i>	Required	Fixed value, IM_spatialZone
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name


Figure 9.5: Spatial Zone Area

10 Pipenetworks

The **<PipeNetworks>** in Inframodel covers different drainage and utility network types: storm drain, combined sewer, sewer, French drain, culvert, water pipe, district heating, district cooling, gas, waste disposal and cable networks.

A network model can be transferred in a separate file or (e.g. in case of route drainage plan) in the same file as other (route) plan content. The metric units used in network plan and the coordinate system are defined in section 1 *File headers*.

The definition of a utility network in LandXML is a topological one. The structures (manholes, drain wells etc.) are nodes and the pipes (or cables) are links that connect these nodes. As the pipes are always defined between two nodes, the free ends of pipes are modelled as virtual structures, of which there are two types, inlets and outlets.

The networks are presented in the parent element **<PipeNetworks>** whose child elements are individual **<PipeNetwork>** elements. A **<PipeNetwork>** consists of structures and pipes.

- Structures **<Structs>**
 - types: circular and rectangular wells, virtual (inlet/outlet) structures, equipment and pipe joints, extensions and inflexions (virtual joints).
 - are given an explicit location (northing, easting, elevation)
 - pipe inlet/outlet in a structure is an **<Invert>**
- Pipes **<Pipes>**
 - types: circular pipe, elliptic pipe, egg-shaped pipe, rectangular pipe and open channel.
 - have a relative location by the named ending and starting structure
 - the length is given as the distance between the centers of the terminal structures at the elevation of the inverts.
 - the exact length may be calculated as the distance between the coordinates of the pipe start and end given in the extension “IM_pipe”.
 - pressurized pipes can be described in Inframodel extension “IM_pipe” using the property “pressureClass”.

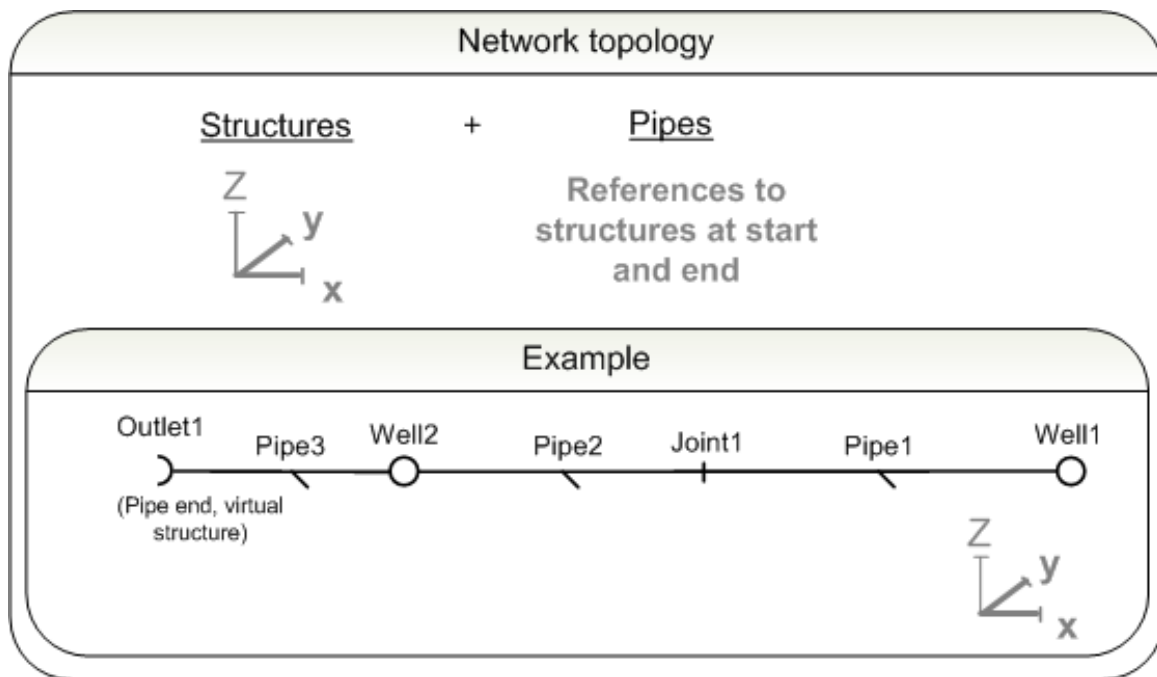


Figure 10.1: Pipe networks

The pipe network is, depending on the situation, usually described in its entirety. LandXML schema exposes some limitations on how a network can be delimited. E.g. describing the inverts of a delimiting storm drain requires additional pipes and corresponding terminal structures to be in the model. The process of delimiting the network is described in further detail in section **10.4 Structures** and section **10.5 Pipes**.

10.1 Pipe networks

A file may contain multiple *utility network groups* **<PipeNetworks>**. It is mandatory to give a **name** and optional to give a *description* **desc** and **state**.

The names of *utility network groups* are unique within the file. If a *network group* contains networks with different states, the **state** shall not be set for the group.

A **<PipeNetworks>** shall have the fields defined in Table 10.1:

Table 10.1: **<PipeNetworks>** fields.

Field	Type	Use	Description
At least one <PipeNetwork> element	<PipeNetwork>	Required	Pipe network, defined by <Structs> and <Pipes>
Zero or one IM_plan <Feature> element	IM_plan <Feature>	Optional	Inframodel plan information <Feature> extension

Field	Type	Use	Description
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

10.1.1 Plan information

The *plan information* of a *network group* is set in the optional “IM_plan” extension under **<PipeNetworks>** element. If the project consists of several sub-divisions, which progress at different rate, the plan content can be divided into several *network group* **<PipeNetworks>** elements according to those divisions, or alternatively *plan information* may be set for each *network* in separate “IM_plan” under **<PipeNetwork>** elements. The plan information contains the **planName** the **planCode**, the **planState**, and the *plan description* **planDesc**. The *plan state* is set according to a scale agreed on by the parties. An example is presented in the tabel below.

Type *IM_plan* <Feature> is defined per Table 2.3 above.

10.2 Pipe network

Individual networks are described in **<PipeNetwork>** elements, organized under their parent element **<PipeNetworks>**, the *network group*. The number of *networks* in one *network group* is unlimited. The *pipe network* defines a topological model, with mandatory **name** and **pipeNetType**. The **state** and *description* **desc** are optional.

A <PipeNetwork> shall have the fields defined in Table 10.2:

Table 10.2: <PipeNetwork> fields.

Field	Type	Use	Description
One <Structs> element	<Structs>	Required	Structures of pipe network
One <Pipes> element	<Pipes>	Required	Pipes of pipe network
Zero or one <i>IM_plan</i> <Feature> element	<i>IM_plan</i> <Feature>	Optional	Inframodel plan information <Feature> extension
Zero or one <i>IM_pipeNetworkType</i> <Feature> element	<i>IM_pipeNetworkType</i> <Feature>	Optional	Inframodel additional pipe network type <Feature> extension

Field	Type	Use	Description
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Required	Unique name
pipeNetType	<i>pipeNetworkType</i>	Required	Pipe network type
desc	<i>string</i>	Optional	Description
state	<i>stateType</i>	Optional	State

A *pipeNetworkType* is defined per Table 10.3:

Table 10.3: <pipeNetworkType> definition.

Type	Basetype	Restriction	Description
pipeNetworkType	<i>string</i>	One of (sanitary, storm, water, other)	

10.2.1 Pipe network type extensions

When the *network type* is not one of those covered by the **pipeNetType** (attribute set to “other”), the “IM_pipeNetworkType” extension can be used to specify the type.

An *IM_pipeNetworkType* <Feature> shall have the fields defined in Table 10.4:

Table 10.4: IM_pipeNetworkType <Feature> fields.

Field	Type	Use	Description
One <i>pipeNetworkType</i> <Property> element	<i>pipeNetworkType</i> <Property>	Required	Pipe network type
code	<i>string</i>	Required	Fixed value, IM_pipeNetworkType
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

A *pipeNetworkType* <Property> shall have the fields defined in Table 10.5:

Table 10.5: pipeNetworkType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, pipeNetworkType
value	<i>one of (districtheating , districtcooling, gas, waste transport piping, electric, telecommunication)</i>	Required	Pipe network type

10.3 Units

In Inframodel, the units as specified in section 1 *File headers* shall always be used. The LandXML capability to assing separate units for structures and pipes in networks is not allowed.

10.4 Structures

The topological network consists of different kinds of structures whose exact location is defined in the file. The different structures in the **<PipeNetwork>** compose the *structure group <Structs>*, that has no attributes.

A <Struct> shall have the fields defined in Table 10.6:

Table 10.6: <Struct> fields.

Field	Type	Use	Description
One <Center> element	<Center>	Required	3D Center Point
One <CircStruct> or <RectStruct> or <InletStruct> or <OutletStruct> or <Connection> element	<CircStruct> or <RectStruct> or <InletStruct> or <OutletStruct> or <Connection>	Required	One of structure types or pipe connection
At least one <Invert> element	<Invert>	Required	Invert
Zero or one IM_struct <Feature> element	IM_struct <Feature>	Optional	Inframodel struct details <Feature> extension
Zero or one IM_coding <Feature> element	IM_coding <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one IM_proprietaryCoding <Feature> element	IM_proprietaryCoding <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension

Field	Type	Use	Description
Zero or one <i>IM_spatialZone</i> <Feature> element	<i>IM_spatialZone</i> <Feature>	Optional	Inframodel spatial zone <Feature> extension
name	<i>nameType</i>	Required	Unique name
desc	<i>string</i>	Optional	Description
elevRim	<i>double</i>	Optional	Rim elevation
elevSump	<i>double</i>	Required	Sump elevation
state	<i>stateType</i>	Optional	State

LandXML standard structure types:

- Circular structure
- Rectangular structure
- Inlet structure (free intake end of pipe)
- Outlet structure (free exhaust end of pipe)
- Connections
- Pipe joints, extensions and inflexions
- Equipment (as specified in Inframodel extension)

Delimiting the network is a special case of using connection, described in further detail in the subsection **10.4.4 Pipe connections**.

An *IM_struct* <Feature> shall have the fields defined in Table 10.7:

Table 10.7: *IM_struct* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>structLabel</i> <Property> element	<i>structLabel</i> <Property>	Optional	Struct name or identifier
Zero or one <i>rimType</i> <Property> element	<i>rimType</i> <Property>	Optional	Rim type
Zero or one <i>rimMaterial</i> <Property> element	<i>rimMaterial</i> <Property>	Optional	Rim material
Zero or one <i>rimLoad</i> <Property> element	<i>rimLoad</i> <Property>	Optional	Rim load bearing capacity
Zero or one <i>rimDiameter</i> <Property> element	<i>rimDiameter</i> <Property>	Optional	Rim diameter
Zero or one <i>rimCenter</i> <Property> element	<i>rimCenter</i> <Property>	Optional	Rim centerpoint location
Zero or one <i>heightDeposit</i> <Property> element	<i>heightDeposit</i> <Property>	Optional	Deposit height

Field	Type	Use	Description
Zero or one <i>volumeDeposit</i> <Property> element	<i>volumeDeposit</i> <Property>	Optional	Deposit volume
Zero or one <i>constructionDate</i> <Property> element	<i>constructionDate</i> <Property>	Optional	Construction date
Zero or one <i>renewalDate</i> <Property> element	<i>renewalDate</i> <Property>	Optional	Renewal date
Zero or one <i>renewalDesc</i> <Property> element	<i>renewalDesc</i> <Property>	Optional	Renewal description
One <i>bottomThickness</i> <Property> element	<i>bottomThickness</i> <Property>	Required	Well bottom thickness
code	<i>string</i>	Required	Fixed value, IM_struct
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

10.4.0.1 General data

All structural elements in the file are assigned with unique names.

Type <Struct> is defined per Table 10.6 above.

10.4.0.2 Center

Circular and rectangular structures: The center of the cross-section at the inside bottom level of the structure is set by space-separated 3D-coordinates in the <Center> element.

Pipe inlets and outlets: The pipe end is defined by space-separated 3D-coordinates in the <Center> element.

Pipe connections: The <Center> of a connection, joint or point of intersection at the mounting level is set using space-separated 3D coordinates.

Equipments: The <Center> of a piece of equipment at the mounting level is set using space-separated 3D coordinates.

10.4.0.3 Inverts

The inlets and outlets in a structure for adjoining pipes are described using the element <Invert>. The required attributes of invert are: the elevation <elev>, the flow direction <flowDir> and the reference to the adjoining pipe <refPipe>. The elevation is given according to “elevType” set for the adjoining pipe in “IM_pipe” extension (crown level for pressure pipes and the invert level for others).

An <Invert> shall have the fields defined in Table 10.8:

Table 10.8: <Invert> fields.

Field	Type	Use	Description
desc	<i>string</i>	Optional	Description
elev	<i>double</i>	Required	Elevation
flowDir	<i>inOut</i>	Required	Flow direction
refPipe	<i>pipeNameRef</i>	Required	Reference to pipe

10.4.0.4 Details

It is possible to set additional properties for structures in “IM_struct” extension.

Common properties applicable to all types of structures are **structLabel** and the dates of different actions, such as the **constructionDate** and the **renewalDate**. The dates of the actions are typically give in years. It is also possible to describe the renewal in further detail, e.g. the method used, using the **renowalDesc** attribute.

Additionally, common to the three structure types below are **rimType** and the rim load bearing class **rimLoad**, as well as **bottomThickness**.

Circular structures:

When describing a conical well the **diameter** attribute describes the inner diameter at the bottom level. The **rimDiameter** describes the inner diameter of the rim and the **rimCenter** defines the 3D coordinates of the rim center. The parameters of the sump are its depth **heightDeposit** and its volume **volumeDeposit**.

Rectangular structure:

The parameters of the sump are its depth **heightDeposit** and its volume **volumeDeposit**.

Equipment:

It is possible to define more detailed type information of a piece of equipment between two pipes, e.g. a Valve using the attributes **equipmentType**, **equipmentCode** and an equipment description **equipmentDesc**.

10.4.0.5 Spatial allocation and avoidance

Related to structure geometry an area or a volume for spatial allocation or avoidance may be defined as “IM_spatialZone” extension. Both spatialAllocation and spatialAvoidance are given as single metric value (in file length units). The allocation and avoidance geometry is interpreted according to the structure geometry definition as a radius around the vertical line defined by structure <Center> 3D coordinates at the bottom level and @elevRim.

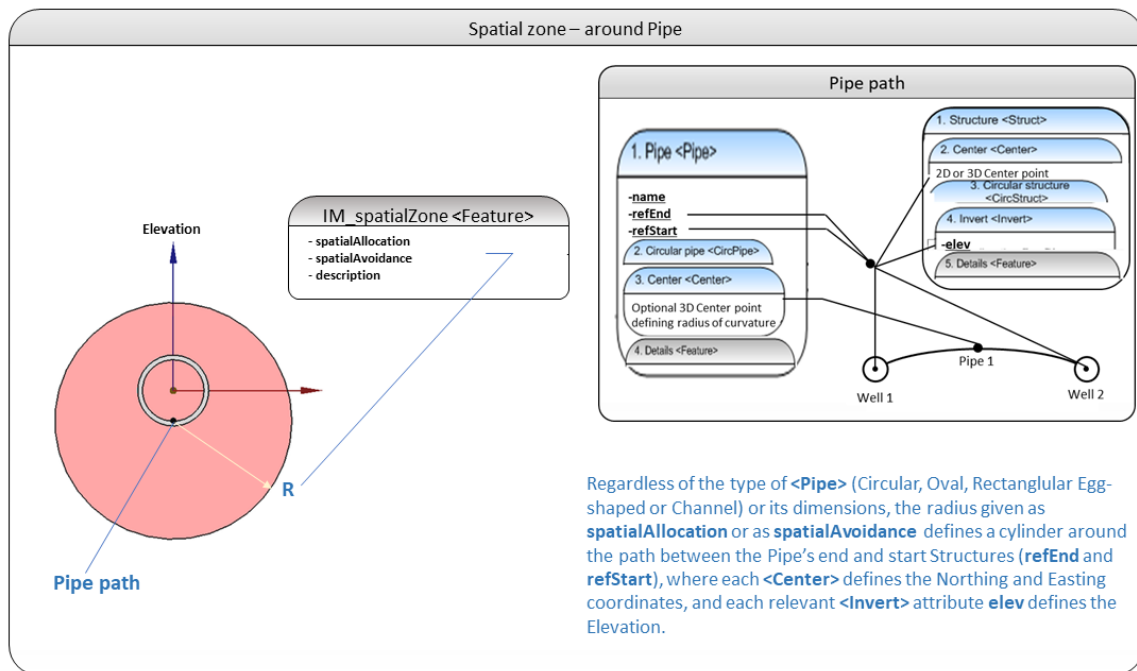


Figure 10.2: Spatial Zone Pipe

More information about the IM_spatialZone <Feature> extension can be found from section **14.20 Spatial allocation and spatial avoidance**.

10.4.1 Circular structures

Inspection wells of French drains are an example of a circular structure. Circular structures are defined using the structure element **<Struct>** and its child elements.

1. Structure element **<Struct>**
2. Structure center **<Center>**
3. Circular structures **<CircStruct>**
4. Inverts **<Invert>**
5. Structure details **<Feature>** "IM_struct" extension

The figure below illustrates the representation of a drain well, including a sump. The sump is defined in the extension "IM_struct" by defining the sump height and volume.

The **body diameter** at the bottom of the well, the **description desc**, the **material** and the **thickness** of the shell material. The well cone and sump are described in further detail in the extension "IM_struct"

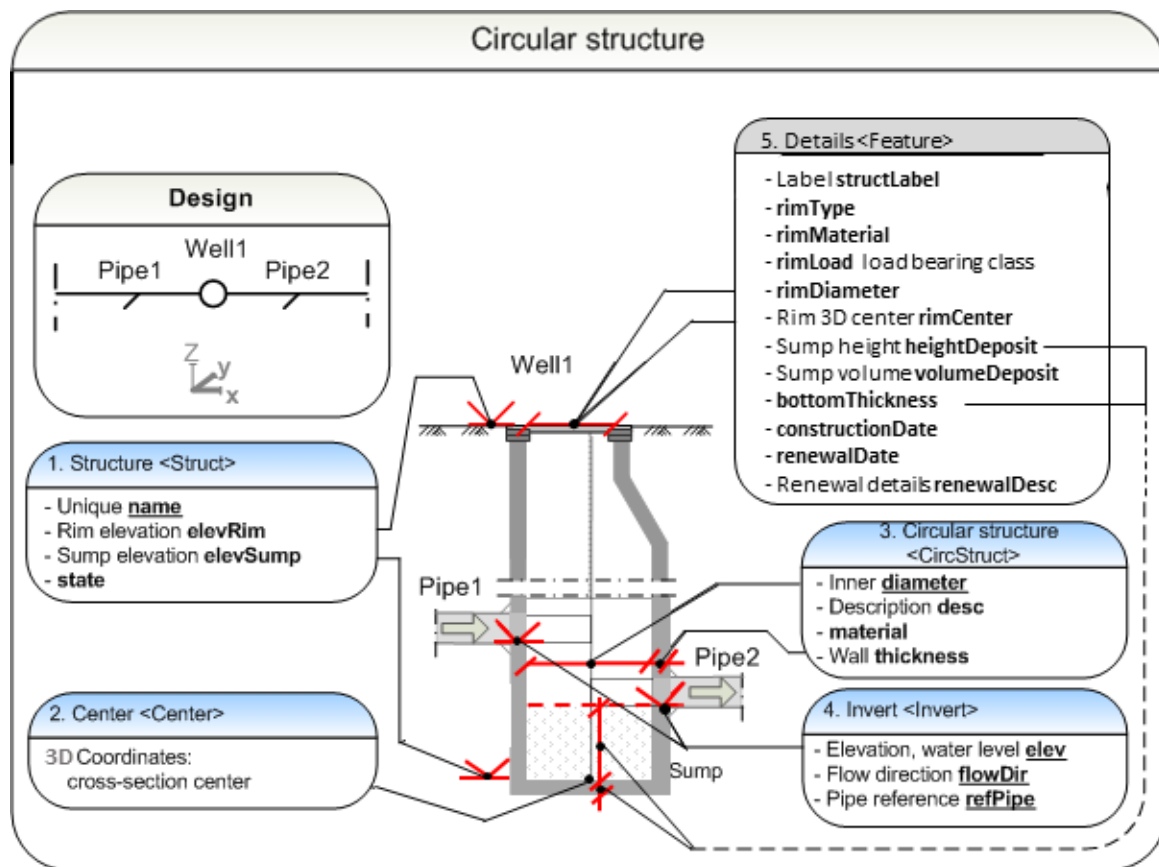


Figure 10.3: Circular structures

A <CircStruct> shall have the fields defined in Table 10.9:

Table 10.9: <CircStruct> fields.

Field	Type	Use	Description
diameter	<i>double</i>	Required	Diameter
desc	<i>string</i>	Required	Description
material	<i>IMPipeNetworkMaterialType</i>	Required	Material
thickness	<i>double</i>	Required	Wall thickness(total thickness including the profile in case of corrugated materials)

An *IMPipeNetworkMaterialType* is defined per Table 10.10:

Table 10.10: <IMPipeNetworkMaterialType> definition.

Type	Basetype	Restriction	Description
IMPipeNetworkMaterialType	string	One of (wood, brick, rock, concrete, aluminium, copper, lead, steel, acidResistantSteel, stainlessSteel, castIron, grayCastIron, sphericalGraphiteCastIron, asbestCement, vitrifiedClay, plastic, PE, PEH, PEL, PEM, PP, PVC, fiberComposite, rubber, bitum, other)	List of materials in Inframodel Pipe Network Structs and Pipes (@material)

10.4.2 Rectangular structures

The illustration below describes the description method of a rectangular structure. The structure is defined using the **<Struct>** element and its child elements:

1. General data **<Struct>**
2. Center **<Center>**
3. Rectangular structure **<RectStruct>**
4. Pipe invert **<Invert>**
5. Structure details **<Feature>** “IM_struct” extension

The illustration demonstrates the method of description of a rectangular well, including a sump as an example. The sump is defined by its depth and/or volume.

The mandatory attributes of a rectangular structure are the **length** the direction of the long edge **lengthDir**, the **width** of the short edge width, the **material** of the structure and the **thickness** of the surface structure.

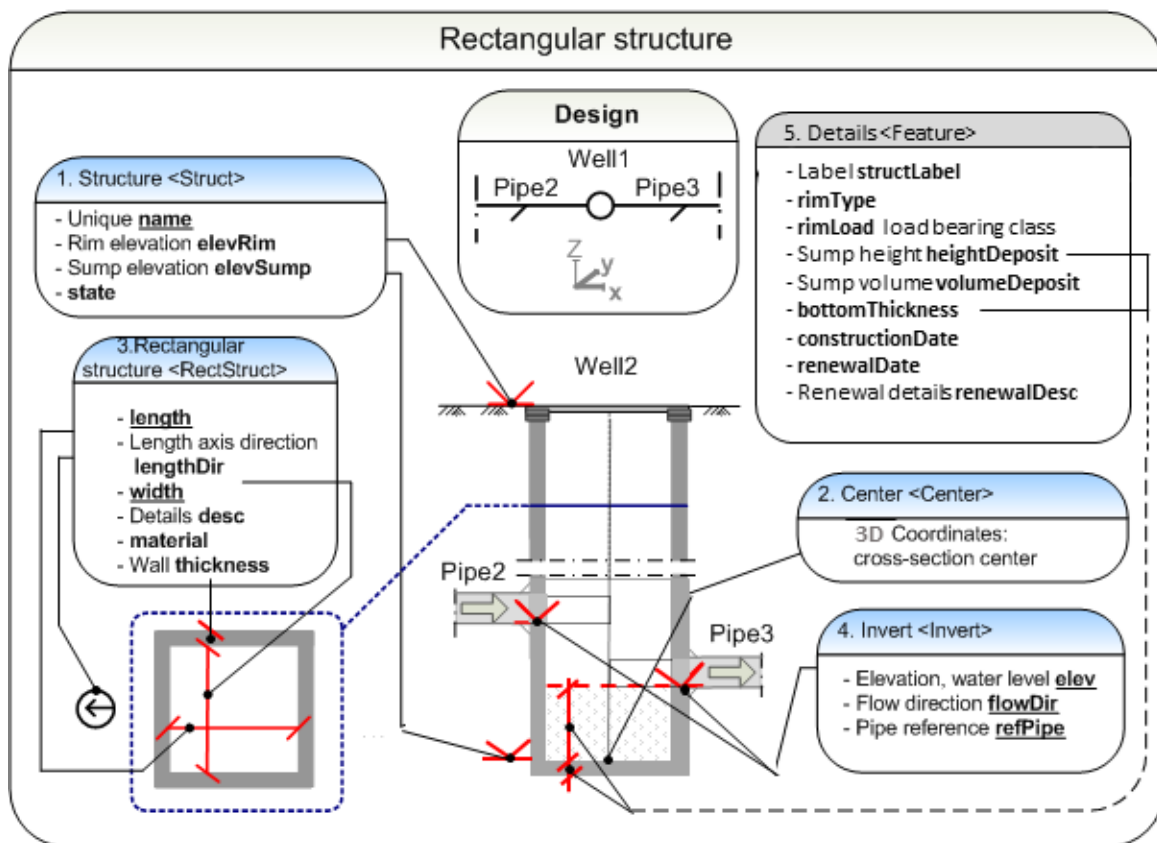


Figure 10.4: Rectangular structures

A *<RectStruct>* shall have the fields defined in Table 10.11:

Table 10.11: *<RectStruct>* fields.

Field	Type	Use	Description
length	<i>double</i>	Required	Length
lengthDir	<i>direction</i>	Required	Direction
width	<i>double</i>	Required	Width
desc	<i>string</i>	Required	Description
material	<i>IMPipeNetworkMaterialType</i>	Required	Material
thickness	<i>double</i>	Required	Wall thickness(total thickness including the profile in case of corrugated materials)

Type *<IMPipeNetworkMaterialType>* is defined per Table 10.10 above.

10.4.3 Pipe inlets and outlets

Pipe inlets and outlets are end of the pipe network pipes. The following illustration demonstrates the method of description. The virtual structures of the pipe ends are defined using the structure attribute **<Struct>** and

its child elements:

1. General description **<Struct>**
2. Center **<Center>**
3. Pipe inlet or pipe outlet **<OutletStruct>**
4. Inverts **<Invert>**
5. Structure details **<Feature>** “IM_struct” extension

The illustration below demonstrates how pipe inlets and outlets are described. The example demonstrates an outlet.

An **<InletStruct>** shall have the fields defined in Table 10.12:

Table 10.12: **<InletStruct>** fields.

Field	Type	Use	Description
-------	------	-----	-------------

An **<OutletStruct>** shall have the fields defined in Table 10.13:

Table 10.13: **<OutletStruct>** fields.

Field	Type	Use	Description
-------	------	-----	-------------

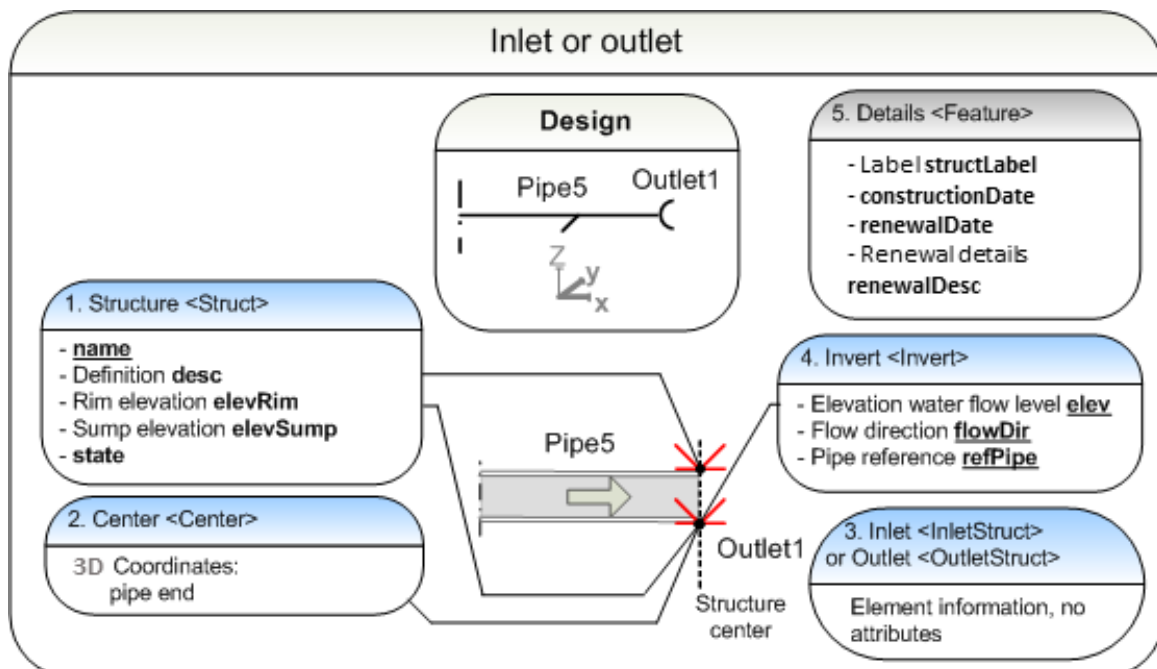


Figure 10.5: Inlets and outlets

10.4.4 Pipe connections

Pipe connections, joints and points of intersection are defined by the **<Connection>** elements. The illustration below demonstrates the mode of description, which contains the attributes of the structure **<Struct>** and its child elements:

1. General data **<Struct>**
2. Center **<Center>**
3. Pipe connections, joints or points of intersection **<Connection>**
4. Inverts **<Invert>**
5. Details **<Feature>** “IM_struct” structural extension

The illustration demonstrates the description method of a point of intersection.

When using the element to delimit a pipe network, the terminal drainage well is connected to a pipe that terminates in a **<Connection>** element. It is thus possible to also describe the connections of the outermost wells in the plan network. It is advisable to name the elements in a fashion that is clearly different from the rest of the plan.

When the element describes a delimiting element, all attributes are not used. The elevation of the rim *elevRim*, the elevation of the sump *elevSump* and the state of the structure are left undefined (given value “NaN”).

When a **<Connection>** is used for modeling a bend (inflexion) of a flexible pipe of cable (being actually continuous), its name and description *desc* should indicate that it is not a physical connection.

Connections, joints or points of intersection are defined using the **<Connection>** element, that has no attributes.

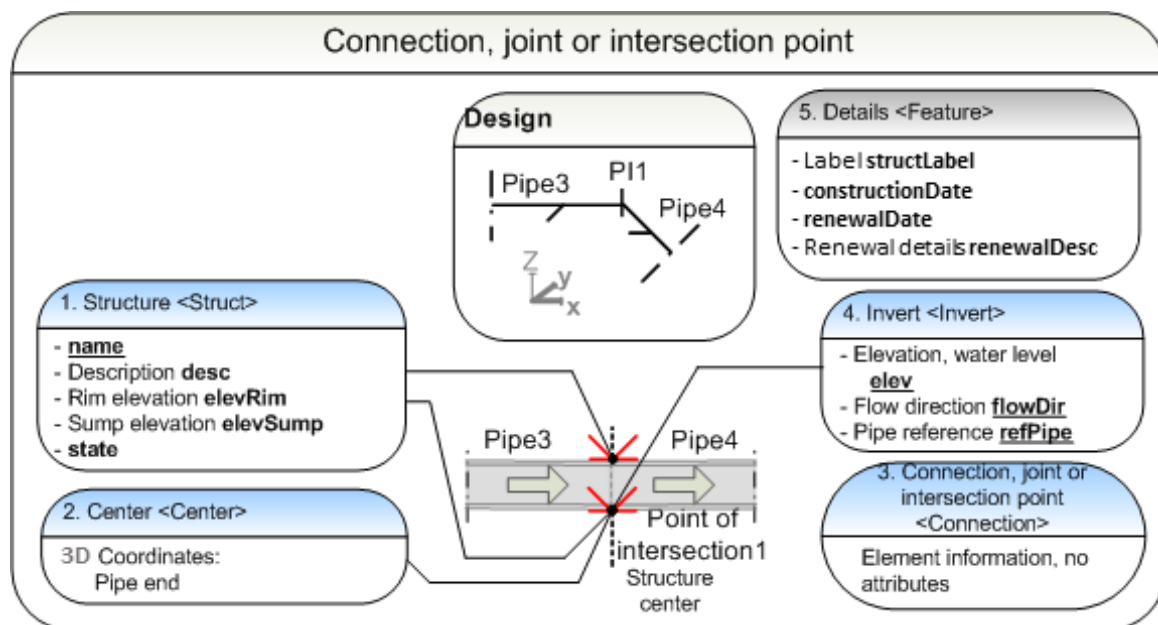


Figure 10.6: Pipe connections

Connection element has no attributes or child elements.

10.4.5 Equipment

Equipment is defined using the **<Connection>** element. The illustration bellow describes the mode of description of a *structure* **<Struct>** and its child elements:

1. General data **<Struct>**
2. Center **<Center>**
3. Equipment **<Connection>**
4. Inverts **<Invert>**
5. Details **<Feature>** “IM_struct” structural extension

The illustration demonstrates the mode of description of a valve:

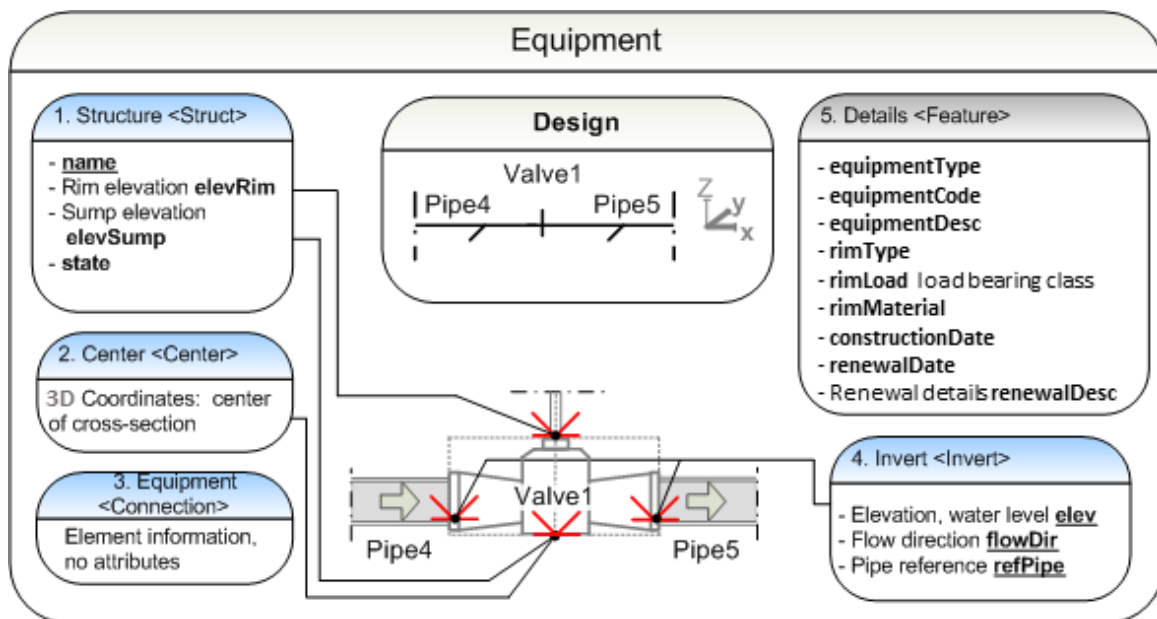


Figure 10.7: Equipment

A *piece of equipment* is defined using the element **<Connection>**, that has no attributes. Details of the equipment are defined in the extension “IM_struct”.

An *IM_struct* **<Feature>** shall have the fields defined in Table 10.14:

Table 10.14: IM_struct **<Feature>** fields.

Field	Type	Use	Description
Zero or one <i>structLabel</i> <Property> element	<i>structLabel</i> <Property>	Optional	Struct name or identifier
Zero or one <i>rimType</i> <Property> element	<i>rimType</i> <Property>	Optional	Rim type
Zero or one <i>rimLoad</i> <Property> element	<i>rimLoad</i> <Property>	Optional	Rim load bearing capacity
Zero or one <i>constructionDate</i> <Property> element	<i>constructionDate</i> <Property>	Optional	Construction date

Field	Type	Use	Description
Zero or one <i>renewalDate</i> <Property> element	<i>renewalDate</i> <Property>	Optional	Renewal date
Zero or one <i>renewalDesc</i> <Property> element	<i>renewalDesc</i> <Property>	Optional	Renewal description
Zero or one <i>equipmentType</i> <Property> element	<i>equipmentType</i> <Property>	Optional	Equipment type
Zero or one <i>equipmentDesc</i> <Property> element	<i>equipmentDesc</i> <Property>	Optional	Equipment description
Zero or one <i>equipmentCode</i> <Property> element	<i>equipmentCode</i> <Property>	Optional	Equipment code
Zero or one <i>equipmentCodeDesc</i> <Property> element	<i>equipmentCodeDesc</i> <Property>	Optional	Equipment code description
Zero or one <i>equipmentCodeSource</i> <Property> element	<i>equipmentCodeSource</i> <Property>	Optional	Equipment code source
code	<i>string</i>	Required	Fixed value, IM_struct
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

10.5 Pipes

The topological network consists of different kinds of structures, whose exact location is given. The pipes that compose the **<PipeNetwork>** are described as a *structure collection* **<Pipes>** that has no attributes. A **<Pipe>** is defined between two structure nodes **<Struct>** by referring to them in the start- and end-attributions.

NOTE: In the special case of “network” where there is only one single pipe (such as a culvert with no specified connection at either end), the referenced start and end structure shall be **<InletStruct>** and **<OutletStruct>**, respectively.

Available pipe types in the LandXML standard:

- Circular pipe
- Egg-shaped pipe
- Elliptic pipe
- Rectangular pipe
- Channel

A *<Pipes>* shall have the fields defined in Table 10.15:

Table 10.15: <Pipes> fields.

Field	Type	Use	Description
At least one <Pipe> element	<Pipe>	Required	Individual pipe
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension

10.5.1 Pipe

When using a pipe to delimit a network, its refEnd or refStart shall be to a <Connection> with a name that clearly distinguishes it from other content in the file.

When a pipe has several segments, with <Connection> elements (physical or virtual) between them, each segment must be a <Pipe> element with unique **name**. If the whole pipe (e.g. from one well to another) needs a name, it should be given in **desc** attribute of every segment. In particular, this should be provided for continuous flexible pipes or cables having inflexions modeled as virtual connections.

A <Pipe> shall have the fields defined in Table 10.16:

Table 10.16: <Pipe> fields.

Field	Type	Use	Description
One <CircPipe> or <EggPipe> or <ElliPipe> or <RectPipe> or <Channel> element	<CircPipe> or <EggPipe> or <ElliPipe> or <RectPipe> or <Channel>	Required	
Zero or one <Center> element	<Center>	Optional	3D Center Point
Zero or one <i>IM_cable</i> <Feature> element	<i>IM_cable</i> <Feature>	Optional	Inframodel cable properties <Feature> extension
Zero or one <i>IM_pipe</i> <Feature> element	<i>IM_pipe</i> <Feature>	Optional	Inframodel pipe details <Feature> extension
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension

Field	Type	Use	Description
Zero or one <i>IM_spatialZone</i> <Feature> element	<i>IM_spatialZone</i> <Feature>	Optional	Inframodel spatial zone <Feature> extension
name	<i>nameType</i>	Required	Unique name
refEnd	<i>structNameRef</i>	Required	Reference to struct or connection where the pipe endpoint is connected.
refStart	<i>structNameRef</i>	Required	Reference to struct or connection where the pipe startpoint is connected.
slope	<i>double</i>	Required	Pipe slope
desc	<i>string</i>	Optional	Description
length	<i>double</i>	Optional	Pipe length
state	<i>stateType</i>	Optional	State

10.6 Center

The pipe curvature is defined by space-separated 3D-coordinates in the <Center> element.

More details can be found from section **10.7 Details**

10.7 Details

It is optional to define details in inframodel file transfers using “IM_pipe” extension. The label of pipe can be given using a pipeLabel. The start and end coordinates of a pipe are defined by three parameters:

1. elevation type elevType
2. start coordinate pipeStart and the
3. end coordinate pipeEnd.

The jointType sets the type of joints and connections used for the pipe. When the defined pipe is a pressurized sewer the pressureClass of the pipe is defined. Procedure details are defined with the constructionDate and renewalDate. It is also possible to define detailed information of the renewal using the renewal description renewalDesc, e.g. the method used to renew the pipe.

Rectangular and elliptic pipe: When start and end coordinates are defined, elevation type shall be given as one of the elevTypes illustrated for circular pipes.

Egg pipe: When start and end coordinates are defined, elevation type shall be given as one of the elevTypes illustrated for circular pipes, center being at the level where the cross section is the widest.

When the start and end coordinates are given with elevation values, elevation type shall be given as one of the enumerated elevTypes shown below.

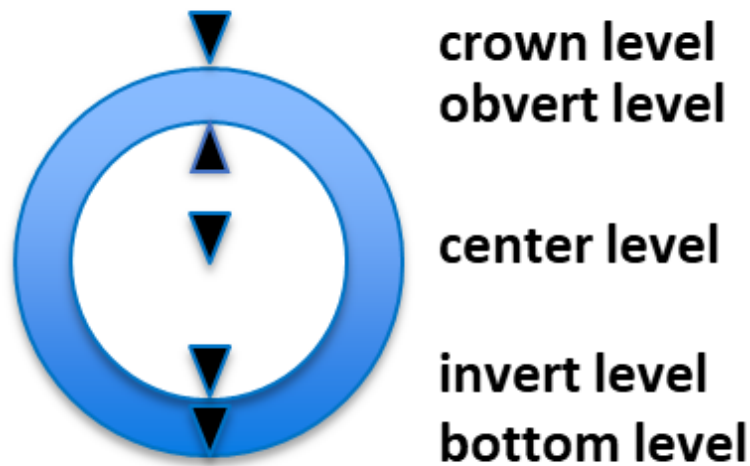


Figure 10.8: Elevation type

An *IM_pipe* <Feature> shall have the fields defined in Table 10.17:

Table 10.17: *IM_pipe* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>pipeLabel</i> <Property> element	<i>pipeLabel</i> <Property>	Optional	Pipe name or identifier
One <i>elevType</i> <Property> element	<i>elevType</i> <Property>	Required	Elevation type
Zero or one <i>pipeStart</i> <Property> element	<i>pipeStart</i> <Property>	Optional	Pipe start location. Note, this field is informational, actual pipe start point is set by <Struct> referenced by <Pipe>.refStart
Zero or one <i>pipeEnd</i> <Property> element	<i>pipeEnd</i> <Property>	Optional	Pipe end location. Note, this field is informational, actual pipe end point is set by <Struct> referenced by <Pipe>.refEnd
Zero or one <i>jointType</i> <Property> element	<i>jointType</i> <Property>	Optional	Pipe joint type
Zero or one <i>constructionDate</i> <Property> element	<i>constructionDate</i> <Property>	Optional	Construction date
Zero or one <i>renewalDate</i> <Property> element	<i>renewalDate</i> <Property>	Optional	Renewal date

Field	Type	Use	Description
Zero or one <i>renewalDesc</i> <Property> element	<i>renewalDesc</i> <Property>	Optional	Renewal description
Zero or one <i>strengthClass</i> <Property> element	<i>strengthClass</i> <Property>	Optional	Pipe strength class
Zero or one <i>surfaceProfile</i> <Property> element	<i>surfaceProfile</i> <Property>	Optional	Surface profile of struct/pipe
code	<i>string</i>	Required	Fixed value, IM_pipe
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Cable: When the <Pipe> represents a cable, and its start and end coordinates are defined, elevation type shall be given as one of the *elevTypes* illustrated for circular pipes. Also, cable-specific properties in separate “IM_cable” extension:

An *IM_cable* <Feature> shall have the fields defined in Table 10.18:

Table 10.18: *IM_cable* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>label</i> <Property> element	<i>label</i> <Property>	Optional	Cable label
Zero or one <i>owner</i> <Property> element	<i>owner</i> <Property>	Optional	Cable owner
Zero or one <i>networkType</i> <Property> element	<i>networkType</i> <Property>	Optional	Cable network type
Zero or one <i>cableType</i> <Property> element	<i>cableType</i> <Property>	Optional	Cable type
code	<i>string</i>	Required	Fixed value, IM_cable
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

10.7.0.1 Spatial allocation and avoidance

Related to pipe geometry an area or a volume for spatial allocation or avoidance may be defined as “IM_spatialZone” extension. Both *spatialAllocation* and *spatialAvoidance* are given as single metric value (in file length units). The allocation and avoidance geometry is interpreted according to the pipe geometry definition as a radius around the pipe path defined by start and end structure <Center> coordinates and their **<Invert>** elevation values .

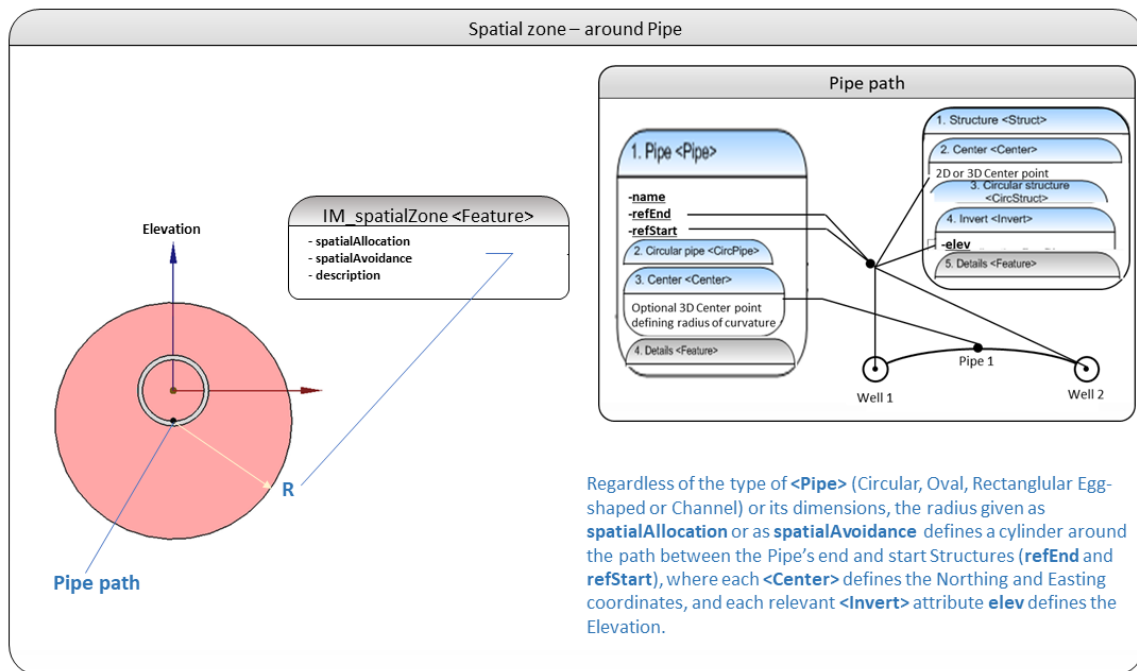


Figure 10.9: Spatial zone - around Pipe

10.7.0.2 Circular pipes

The following illustration describes the definition of a round pipe. The definition employs the element <Pipe> and its child elements.

1. Pipe <Pipe>
2. Circular pipe <CircPipe>
3. Center <Center>
4. Pipe details <Feature> "IM_pipe" extension

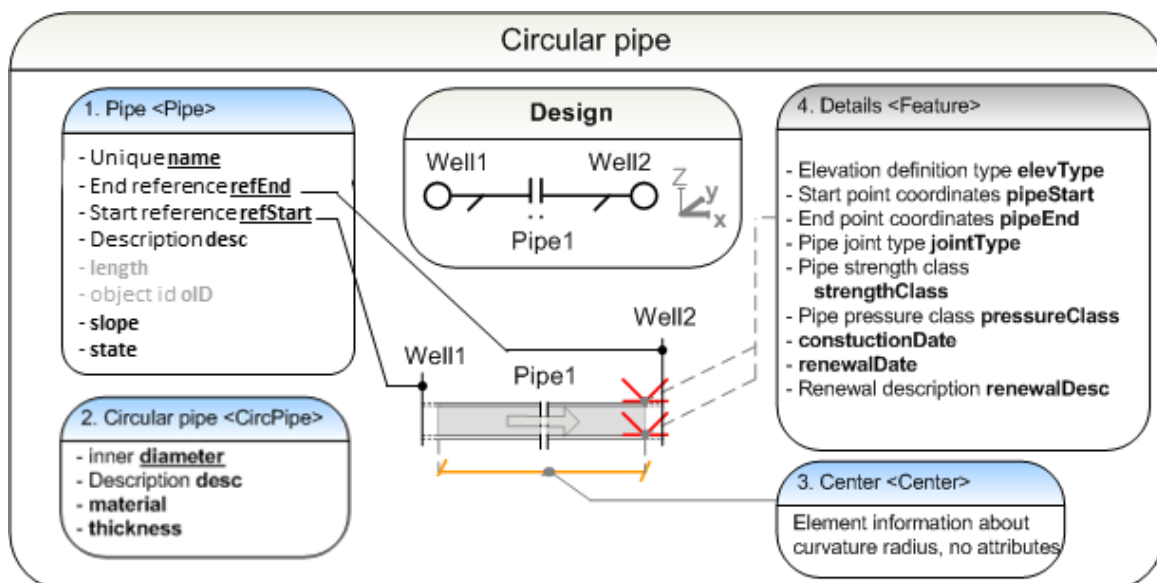


Figure 10.10: Circular pipe

When defining a network-limiting pipe the **diameter** is the only defined attribute. Other attributes are not

defined.

A *<CircPipe>* shall have the fields defined in Table 10.19:

Table 10.19: *<CircPipe>* fields.

Field	Type	Use	Description
diameter	<i>double</i>	Required	Diameter
material	<i>IMPipeNetworkMaterialType</i>	Required	Material
thickness	<i>double</i>	Required	Wall thickness(total thickness including the profile in case of corrugated materials)
desc	<i>string</i>	Optional	Description

Type *<IMPipeNetworkMaterialType>* is defined per Table 10.10 above.

10.7.1 Egg pipes

The following illustration describes the definition of an egg-shaped pipe The definition employs the element *<Pipe>* and its child elements.

1. Pipe **<Pipe>**
2. Egg pipe **<EggPipe>**
3. Center **<Center>**
4. Pipe details **<Feature>** “IM_pipe” extension

Illustration of egg-shaped pipe representation.

When defining a network-limiting pipe the *height* **height** and *width* **span** are set. Other attributes are not set.

An *<EggPipe>* shall have the fields defined in Table 10.20:

Table 10.20: *<EggPipe>* fields.

Field	Type	Use	Description
height	<i>double</i>	Required	Height
span	<i>double</i>	Required	Span
material	<i>IMPipeNetworkMaterialType</i>	Required	Material
thickness	<i>double</i>	Required	Wall thickness(total thickness including the profile in case of corrugated materials)
desc	<i>string</i>	Optional	Description

Type `<IMPipeNetworkMaterialType>` is defined per Table 10.10 above.

10.7.2 Elliptic pipes

The following illustration describes the definition of an elliptical pipe. The definition employs the element `<Pipe>` and its child elements.

1. Pipe `<Pipe>`
2. elliptic pipe `<ElliPipe>`
3. Center `<Center>`
4. Pipe details `<Feature>` “IM_pipe” extension

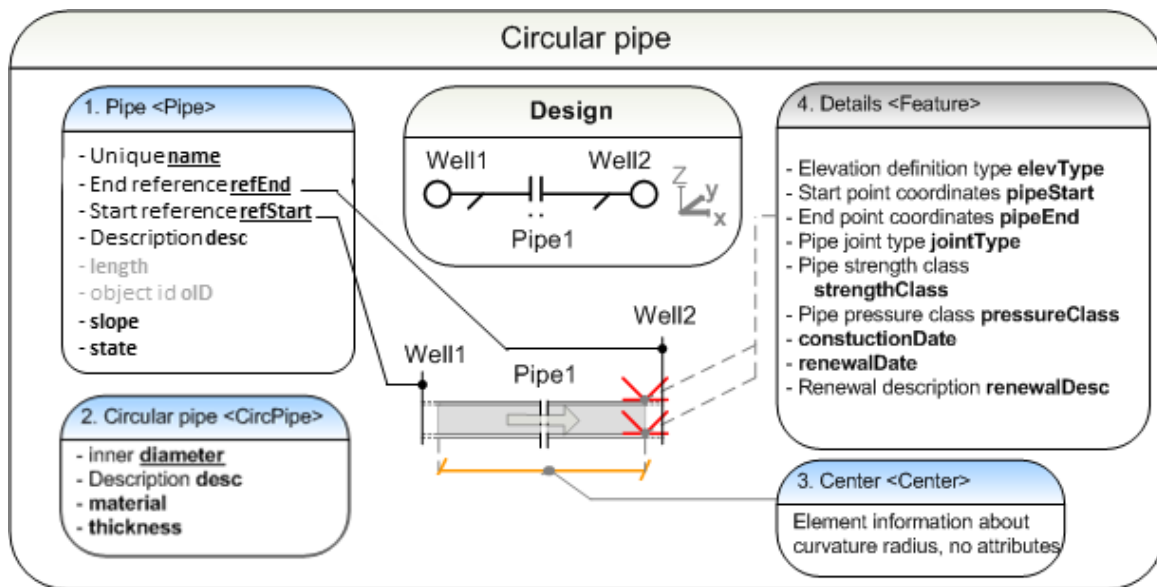


Figure 10.11: Circular pipe

When defining a network mandatory attributes are *height* **height** ja *width* **span**. Other attributes are not set.

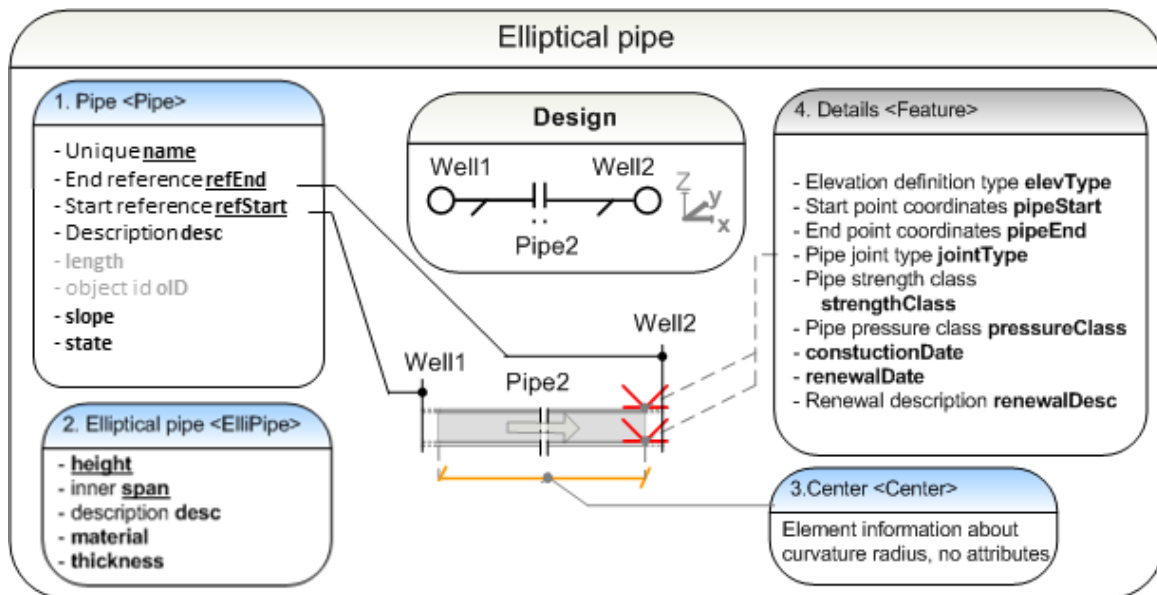


Figure 10.12: Elliptic pipe

An `<ElliPipe>` shall have the fields defined in Table 10.21:

Table 10.21: `<ElliPipe>` fields.

Field	Type	Use	Description
height	<i>double</i>	Required	Height
span	<i>double</i>	Required	Span
material	<i>IMPipeNetworkMaterialType</i>	Required	Material
thickness	<i>double</i>	Required	Wall thickness(total thickness including the profile in case of corrugated materials)
desc	<i>string</i>	Optional	Description

Type `<IMPipeNetworkMaterialType>` is defined per Table 10.10 above.

10.7.3 Rectangular pipes

The following illustration describes the definition of a rectangular pipe. The definition employs the element `<Pipe>` and its child elements.

1. Pipe **<Pipe>**
2. Rectangular pipe **<RectPipe>**
3. Center **<Center>**
4. Pipe details **<Feature>** "IM_pipe" extension

When defining a network-limiting pipe the **height** and **width** are mandatory attributes. Other attributes are not set.

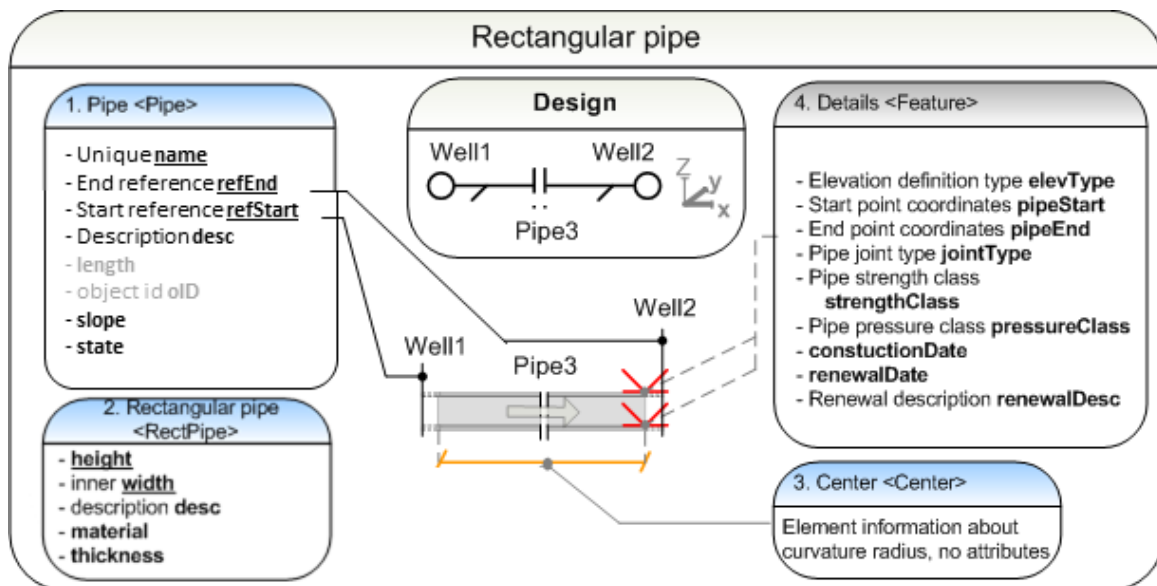


Figure 10.13: Rectangular pipe

A *<RectPipe>* shall have the fields defined in Table 10.22:

Table 10.22: *<RectPipe>* fields.

Field	Type	Use	Description
height	<i>double</i>	Required	Height
width	<i>double</i>	Required	Width
material	<i>IMPipeNetworkMaterialType</i>	Required	Material
thickness	<i>double</i>	Required	Wall thickness(total thickness including the profile in case of corrugated materials)
desc	<i>string</i>	Optional	Description

Type *<IMPipeNetworkMaterialType>* is defined per Table 10.10 above.

10.7.4 Channels

The following illustration describes the definition of a channel pipe. The definition employs the element *<Pipe>* and its child elements.

1. Pipe **<Pipe>**
2. channel **<Channel>**
3. Center **<Center>**
4. Pipe details **<Feature>** "IM_pipe" extension

Illustration of channel description:

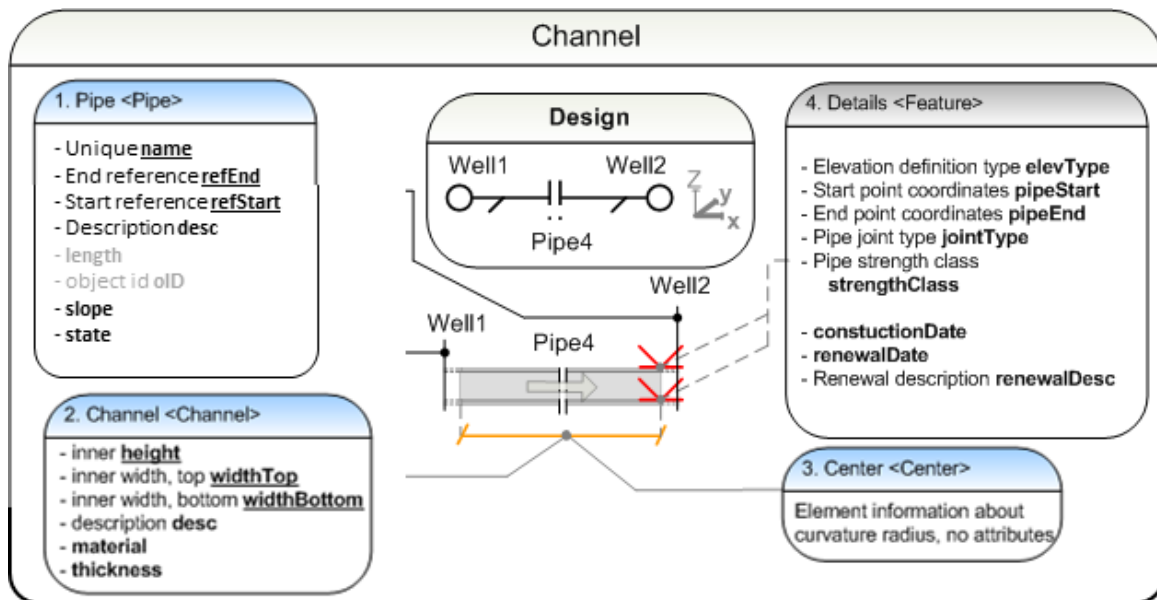


Figure 10.14: Channel

When defining a network mandatory attributes are *height* **height**, *top width* **widthTop** and *bottom width* **widthBottom**. Other attributes are not set.

A <Channel> shall have the fields defined in Table 10.23:

Table 10.23: <Channel> fields.

Field	Type	Use	Description
height	double	Required	Height
widthTop	double	Required	Width at top
widthBottom	double	Required	Width at bottom
desc	string	Optional	Description
material	IMPipeNetworkMaterialType	Optional	Material
thickness	double	Optional	Wall thickness(total thickness including the profile in case of corrugated materials)

Type <IMPipeNetworkMaterialType> is defined per Table 10.10 above.

11 Planimetric features

In an Inframodel file, any plan contains the header information described in section 1 *File headers*. The design content in the file is defined in metric units, using an adequate number of decimals for accuracy. For example, when the length unit in use is meter, the values must be defined to at least six decimal places.

Planimetric features, such as *cables*, lightpole or signage *footings*, *railings* and *fences*, or any other objects not otherwise defined by LandXML schema are described using **<PlanFeature>** element. These features often interface with route (road, street, railway or waterway), in which case they are described under **<Roadways>** collection in the same file, as defined in chapters 4, 5 and 6. This chapter covers those planimetric features not directly assigned to any particular route, and surface structures potentially spanning across multiple routes (such as urban street surface plans).

11.1 Plan features

The planimetric features not directly assigned to any particular route (road, street, railway or waterway) are grouped in *planimetric features collections* under **<PlanFeatures>** consisting of one or several **<PlanFeature>** elements. How the planimetric features are arranged into collections (e.g. by types) is subject to project guidelines or other usage agreements. Each collection shall be given a unique **name**.

A **<PlanFeatures>** shall have the fields defined in Table 11.1:

Table 11.1: **<PlanFeatures>** fields.

Field	Type	Use	Description
At least one <PlanFeature> element	<PlanFeature>	Required	A planimetric feature, such as building footprints, guard rails, tree lines, lightpoles or signage.
Zero or one IM_plan <Feature> element	IM_plan <Feature>	Optional	Inframodel plan information <Feature> extension
Zero or one IM_coding <Feature> element	IM_coding <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one IM_proprietaryCoding <Feature> element	IM_proprietaryCoding <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
name	<i>nameType</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description

Field	Type	Use	Description
state	<i>stateType</i>	Optional	State

Planimetric features collections may be type coded in **<Feature>** using “IM_coding” extension, as well as given alternative or additional type codes in “IM_proprietaryCoding” extension. In particular, this may be practical when the collection contains surface structure definitions associated with the topmost surface of a plan, all having the same type code.

Planimetric features collections may also be assigned custom properties in **<Feature>** using “IM_userDefined-Properties” extension.

11.1.1 Plan feature

The individual planimetric features are each described under **<PlanFeature>**, having a mandatory and unique **name** and either **<Location>** or line geometry as **<CoordGeom>**.

Type *<PlanFeature>* is defined per Table 4.4 above.

<PlanFeature> *geometry* is described in **<CoordGeom>**. For linear features, e.g. *cables*, *railings* and *fences* it shall be either single **line* geometry (also circular** or spiral** curve arcs may be used) or **polyline** geometry. In case of *surface structure* boundaries **closed polyline** geometry shall be given (same value in both **<Start>** and **<End>** of **<IrregularLine>**). For point features, such as *footings*, location is given in **<Location>** element as a two or three dimensional point:

Related to their *geometry* planimetric features may have an area or a volume for spatial allocation or avoidance defined as “IM_spatialZone” extension. Both **spatialAllocation** and **spatialAvoidance** are given as single metric value (in file length units), interpreted according to the type of the plan feature geometry definition:

Point geometry as **<Location>**:

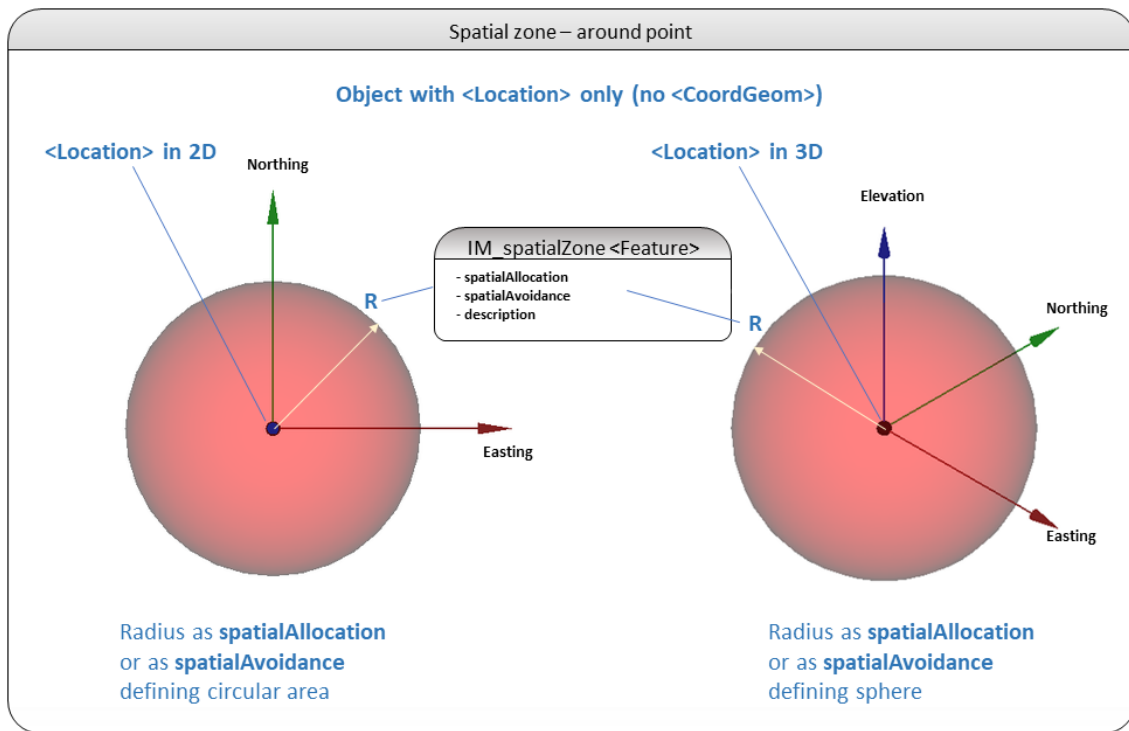


Figure 11.1: Spatial Zone Point

Line or Polyline as <CoordGeom>:

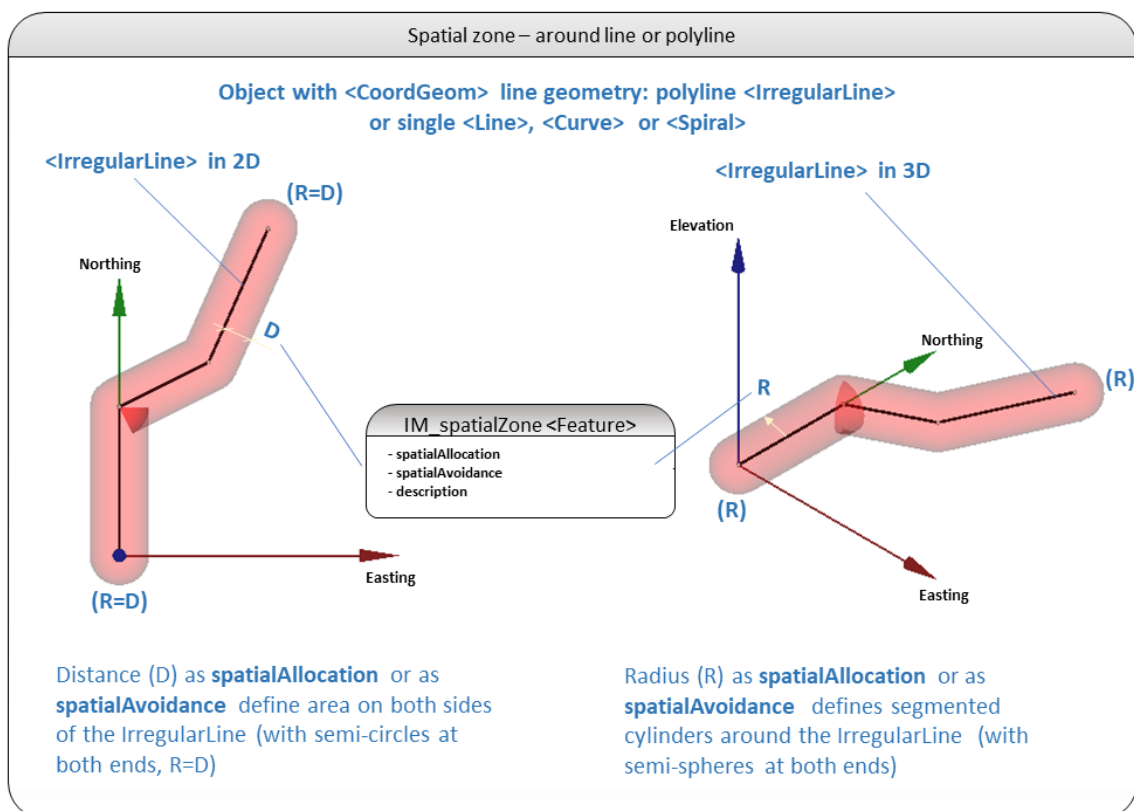


Figure 11.2: Spatial Zone Line

Closed polyline area as <CoordGeom>:

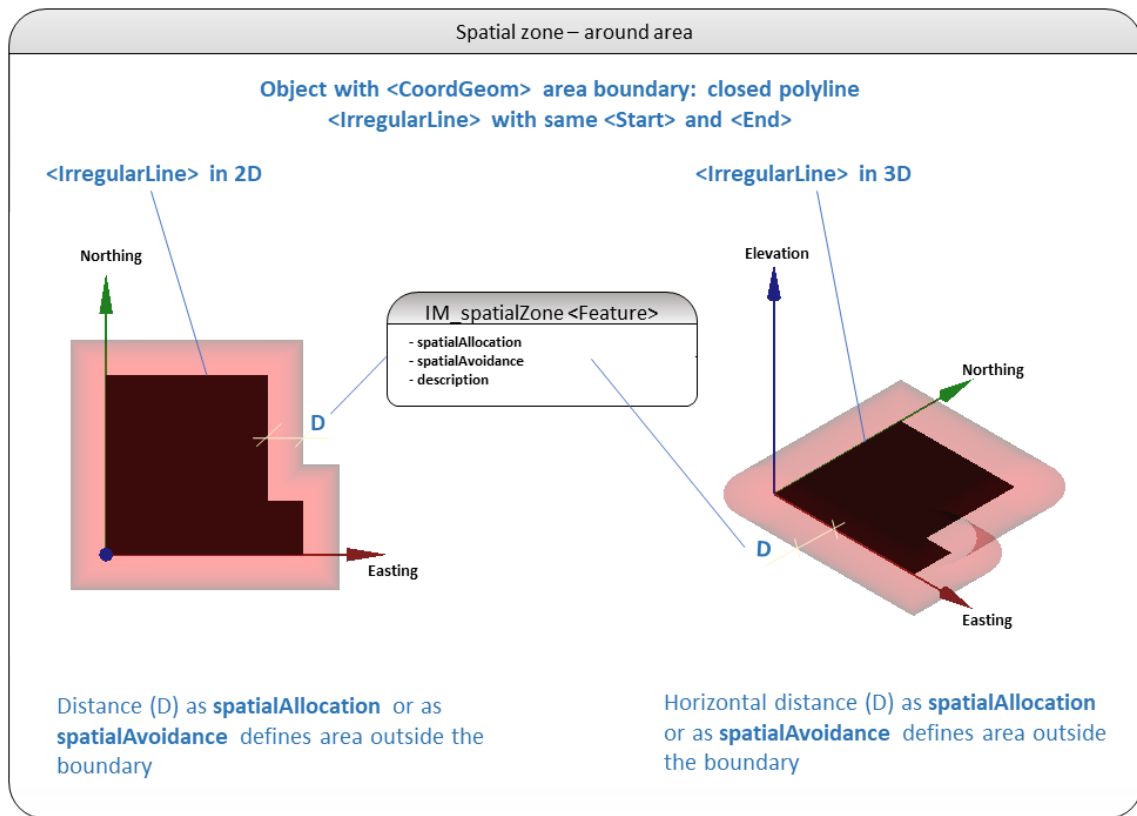


Figure 11.3: Spatial Zone Area

Details of **<PlanFeature>** are described as **<Feature>** extension, defined for each type as follows:

1. Cable information in subsection **11.1.1.1 Cable** “IM_cable” extension
2. Footing information in subsection **11.1.1.2 Footing** “IM_footing” extension
3. Railing information in subsection **11.1.1.3 Railing** “IM_railing” extension
4. Fence information in subsection **11.1.1.4 Fence** “IM_fence” extension
5. Surface structure properties in subsection **11.1.1.5 Surface structure** “IM_surfaceStructure” extension
6. Generic plan feature in subsection **11.1.1.6 Generic feature** “IM_planfeature” extension

All plan features may be type coded in **<Feature>** using “IM_coding” extension, as well as given alternative or additional type codes in “IM_proprietaryCoding” extension. All plan features may also be assigned custom properties in **<Feature>** using “IM_userDefinedProperties” extension.

11.1.1.1 Cable

Details of **<PlanFeature>** in “IM_cable” **<Feature>**

Type *IM_cable* <Feature> is defined per Table 10.18 above.

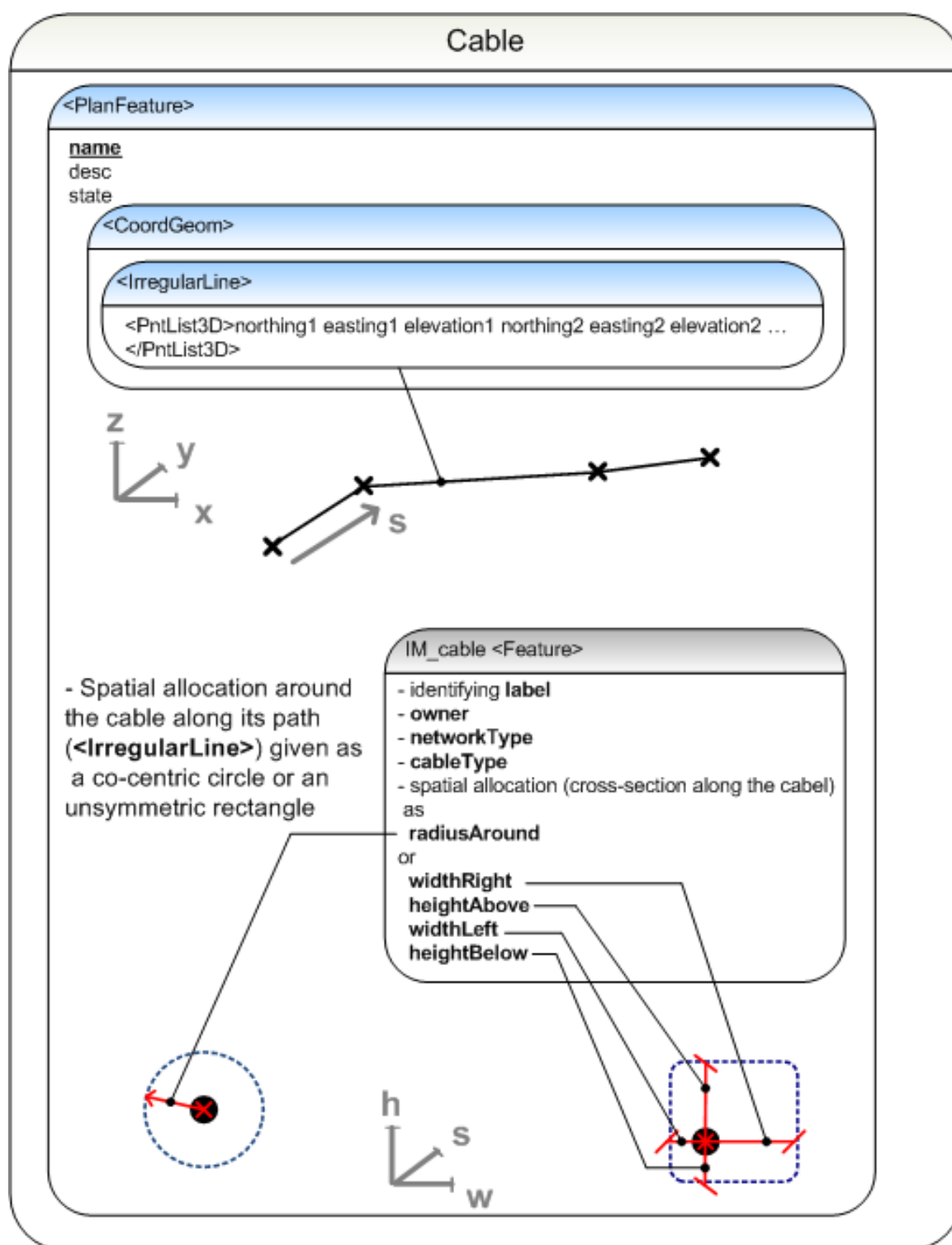


Figure 11.4: Plan Feature cable

11.1.1.2 Footing

Details of **<PlanFeature>** in “IM_footing” **<Feature>**

An *IM_footing* **<Feature>** shall have the fields defined in Table 11.2:

Table 11.2: IM_footing <Feature> fields.

Field	Type	Use	Description
One <i>footingHeight</i> <Property> element	<i>footingHeight</i> <Property>	Required	Footing height
Zero or one <i>radiusAround</i> <Property> element	<i>radiusAround</i> <Property>	Optional	Footing radius
Zero or one <i>length</i> <Property> element	<i>length</i> <Property>	Optional	Footing length
Zero or one <i>lengthDir</i> <Property> element	<i>lengthDir</i> <Property>	Optional	Footing direction
Zero or one <i>width</i> <Property> element	<i>width</i> <Property>	Optional	Footing width
Zero or one <i>label</i> <Property> element	<i>label</i> <Property>	Optional	Footing label
Zero or one <i>footingMaterial</i> <Property> element	<i>footingMaterial</i> <Property>	Optional	Footing material
Zero or one <i>foundationType</i> <Property> element	<i>foundationType</i> <Property>	Optional	Foundation type
code	<i>string</i>	Required	Fixed value, IM_footing
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

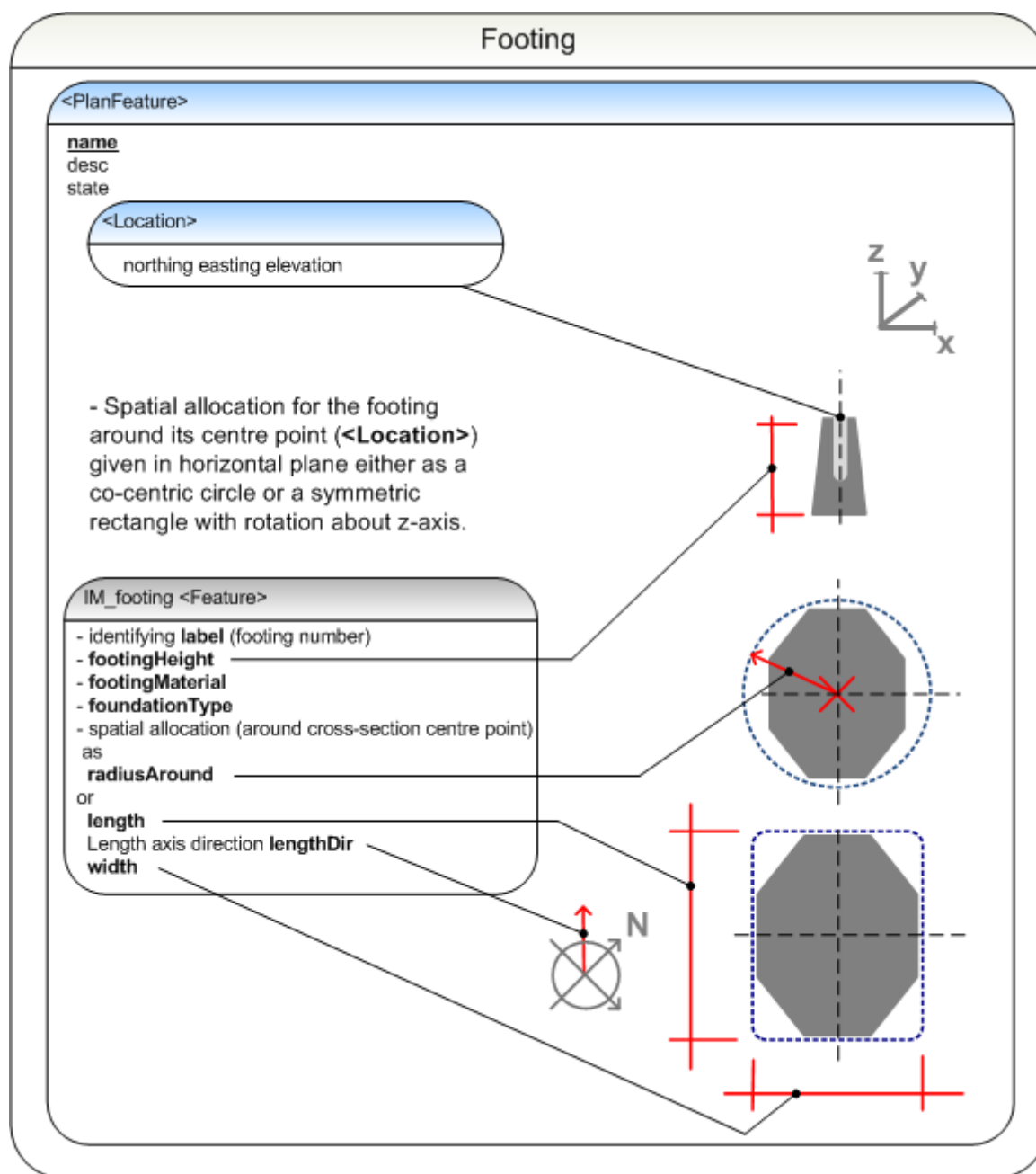


Figure 11.5: Plan Feature footing

11.1.1.3 Railing

In Inframodel, this type of planfeature covers guardrails and guide bars, as well as touch, bump and collision protection structures.

An *IM_railing* <Feature> shall have the fields defined in Table 11.3:

Table 11.3: IM_railing <Feature> fields.

Field	Type	Use	Description
One <i>label</i> <Property> element	<i>label</i> <Property>	Required	Railing label

Field	Type	Use	Description
One <i>railingType</i> <Property> element	<i>railingType</i> <Property>	Required	Railing type
One <i>purpose</i> <Property> element	<i>purpose</i> <Property>	Required	Purpose of railing
Zero or one <i>shockResistanceClass</i> <Property> element	<i>shockResistanceClass</i> <Property>	Optional	Shock resistance class
Zero or one <i>snowPlowingDurabilityCategory</i> <Property> element	<i>snowPlowingDurabilityCategory</i> <Property>	Optional	Category of snow plowing durability of the railing
Zero or one <i>normalizedWorkingWidth</i> <Property> element	<i>normalizedWorkingWidth</i> <Property>	Optional	Normalized working width of the railing
Zero or one <i>railingHeight</i> <Property> element	<i>railingHeight</i> <Property>	Optional	Height of the railing
Zero or one <i>railHeight</i> <Property> element	<i>railHeight</i> <Property>	Optional	Height of the guard rail or guide bar
Zero or one <i>railingMaterial</i> <Property> element	<i>railingMaterial</i> <Property>	Optional	Material of the railing
Zero or one <i>startType</i> <Property> element	<i>startType</i> <Property>	Optional	Type of railing start element
Zero or one <i>endType</i> <Property> element	<i>endType</i> <Property>	Optional	Type of railing end element
Zero or one <i>postSpacing</i> <Property> element	<i>postSpacing</i> <Property>	Optional	Post spacing
code	<i>string</i>	Required	Fixed value, IM_railing
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

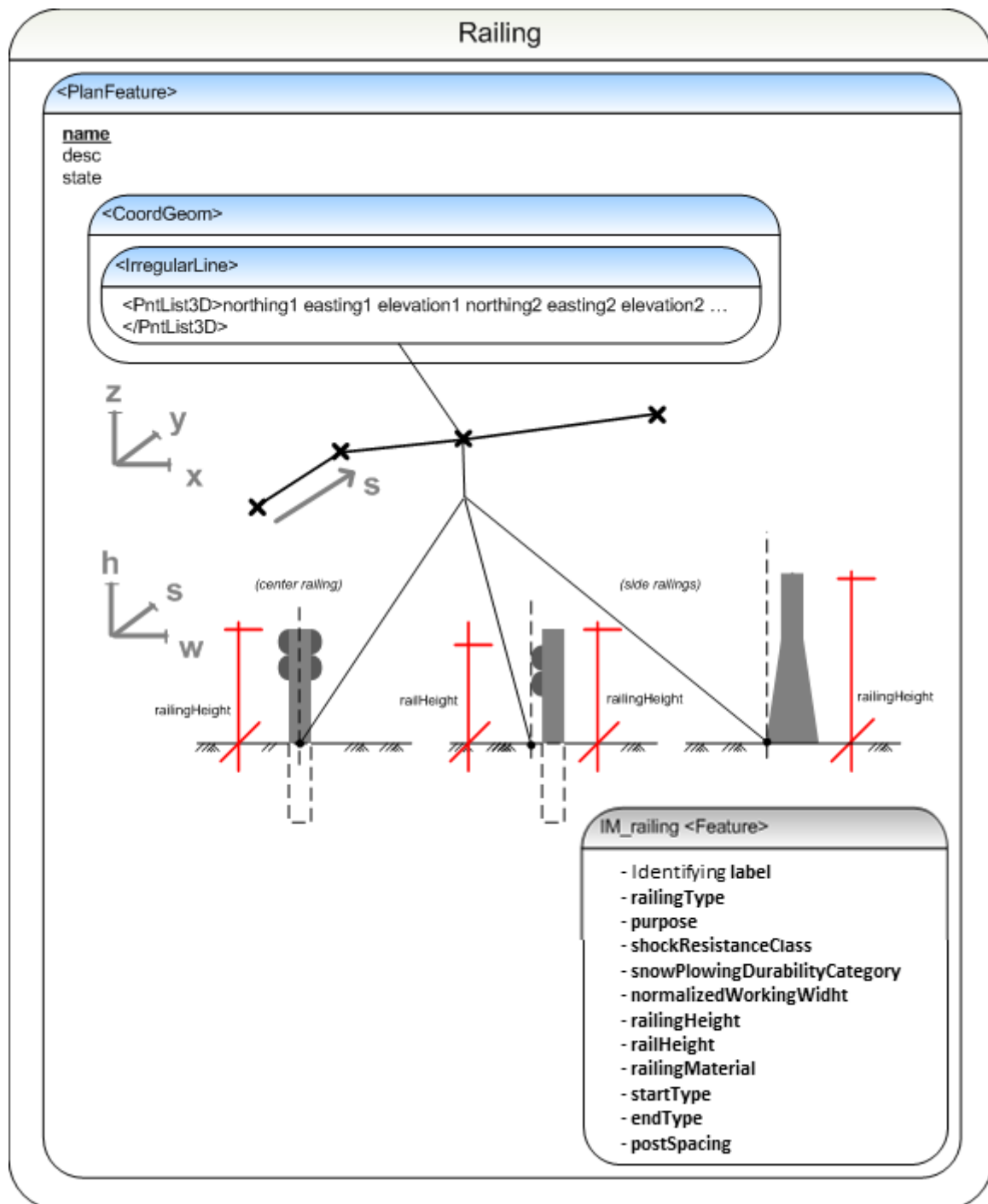


Figure 11.6: Plan Feature railing

11.1.1.4 Fence

In Inframodel, this type of planfeature covers fences, booms and barriers, as well as gates and openings such as fence manholes.

An *IM_fence* <Feature> shall have the fields defined in Table 11.4:

Table 11.4: IM_fence <Feature> fields.

Field	Type	Use	Description
One <i>label</i> <Property> element	<i>label</i> <Property>	Required	Fence label
One <i>fenceType</i> <Property> element	<i>fenceType</i> <Property>	Required	Type of fence
One <i>fenceHeight</i> <Property> element	<i>fenceHeight</i> <Property>	Required	Height of the fence
Zero or one <i>postMaterial</i> <Property> element	<i>postMaterial</i> <Property>	Optional	Material of posts
Zero or one <i>postShape</i> <Property> element	<i>postShape</i> <Property>	Optional	Shape of posts
Zero or one <i>postSpacing</i> <Property> element	<i>postSpacing</i> <Property>	Optional	Post spacing
code	<i>string</i>	Required	Fixed value, IM_fence
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

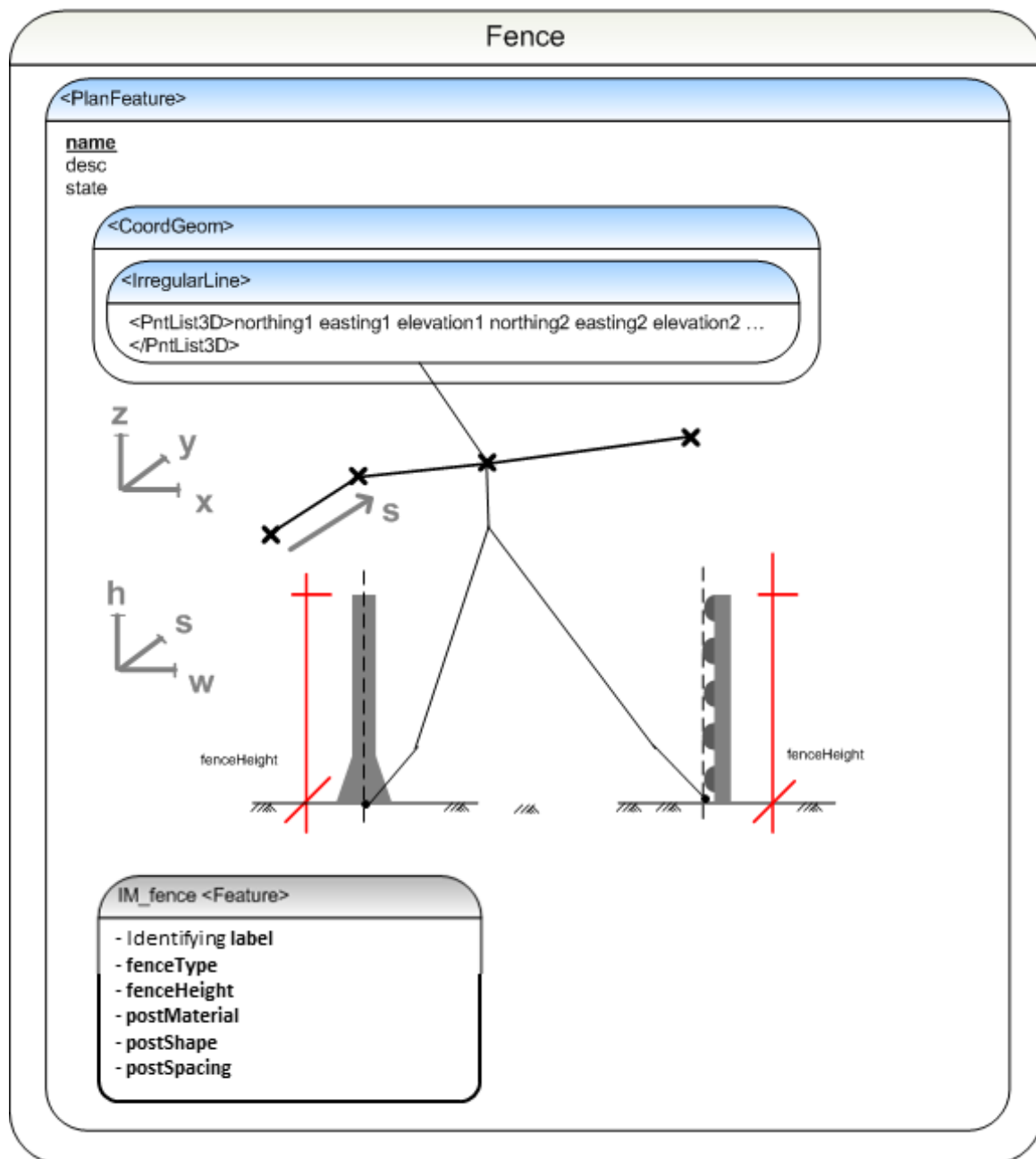


Figure 11.7: Plan Feature fence

11.1.1.5 Surface structure

In Inframodel, this type of planfeature covers surface structure properties (material, thickness).

Type *IM_surfaceStructure* <Feature> is defined per Table 2.13 above.

11.1.1.6 Generic feature

When no specific <Feature> extension defined above can be applied, these generic definitions can be used.

An *IM_planfeature* <Feature> shall have the fields defined in Table 11.5:

Table 11.5: IM_planfeature <Feature> fields.

Field	Type	Use	Description
One <i>type</i> <Property> element	<i>type</i> <Property>	Required	Type of plan feature
One <i>material</i> <Property> element	<i>material</i> <Property>	Required	Material of the plan feature
code	<i>string</i>	Required	Fixed value, IM_planfeature
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

11.1.1.7 Type coding

Objects could be type coded by using “IM_coding” or “IM_proprietaryCoding” <Feature> extensions. “IM_coding” and “IM_proprietaryCoding” are defined in section **14.2** *Type coding systems*.

12 AsBuilt data

An Inframodel file of any plan contains the header information described in section 1 *File headers*.

This section describes as-built data that can include both planned control points with tolerances, and the measured values with metadata (survey and its accuracy). Also the diffence vectors between planned (designed) and measured can be captured.

12.1 Control points

Collections of *coordinate geometry points* grouping control points on a planned surface, on a breakline or on a plan feature, with same tolerance values, are set in **<CgPoints>** elements. Each collection has a unique **name** and a **code**, and these collections can be nested; code is set to “control” in root level control points collection, and to any appropriate value in each subcollection. Optionally, a **description** can be given and a **state** can be set (usually “proposed” for planned points) for any collection of control points. Additionally, the collection may include an IM_coding **<Feature>** element classifying the surface, breakline or planimetric feature where the control points are assigned to. The tolerance values are set in IM_cgpoints **<Feature>** element.

A **<CgPoints>** shall have the fields defined in Table 12.1:

Table 12.1: <CgPoints> fields.

Field	Type	Use	Description
Zero or more <CgPoint> elements	<CgPoint>	Optional	Coordinate geometry (COGO) point.
Zero or one IM_cgpoints <Feature> element	IM_cgpoints <Feature>	Optional	Inframodel pipe details <Feature> extension
Zero or one IM_coding <Feature> element	IM_coding <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or one IM_proprietaryCoding <Feature> element	IM_proprietaryCoding <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension

12.1.1 Control point

Each individual control point is set under **<CgPoint>** element.

A **<CgPoint>** shall have the fields defined in Table 12.2:

Table 12.2: <CgPoint> fields.

Field	Type	Use	Description
content	0 to 3 of type Point	Required	
name	nameType	Optional	Unique name
desc	string	Optional	Description
code	string	Optional	Feature code
state	stateType	Optional	State
timeStamp	datetime	Optional	Time stamp as UTC
surveyOrder	string	Optional	Survey sequence number
pntRef	string	Optional	Reference to control <cgPoint>.name

12.1.2 Tolerances

Tolerance values for all the control points under **<CgPoints>** collection element are set in IM_cgpoints **<Feature>** element.

Horizontal tolerance is given either

1. as allowable deviation *toleranceXY* without specified direction from a given point <CgPoint>*northing easting*</CgPoint>
2. or as allowable deviations from a given point *northing easting*</CgPoint> in specified direction *dirA* (e.g. along a referenced alignment) as *toleranceAmin* and *toleranceAmax*, and in perpendicular (offset) direction *toleranceBmin* and *toleranceBmax*. Both directions, *dirA* and perpendicular, or only one of these may be specified. Also, in both directions it is possible to specify either min or max tolerance, or both. If any of these are specified, also *dirA* shall be.

In both cases, horizontal tolerances can be coupled with vertical tolerance values *toleranceZmin* and *toleranceZmax*. The required combination of tolerance values to be set in IM_cgpoints **<Feature>** must be agreed for each use case separately (in Common InfraBIM Requirements or other such guidelines).

An IM_cgpoints **<Feature>** shall have the fields defined in Table 12.3:

Table 12.3: IM_cgpoints <Feature> fields.

Field	Type	Use	Description
Zero or one <i>toleranceXY</i> <Property> element	<i>toleranceXY</i> <Property>	Optional	Tolerance in horizontal plane from set northing easting in distance units (non-negative number)

Field	Type	Use	Description
Zero or one <i>toleranceAmin</i> <Property> element	<i>toleranceAmin</i> <Property>	Optional	Horizontal minimum tolerance in direction dirA (e.g. along referenced alignment, behind = non-positive number)
Zero or one <i>toleranceAmax</i> <Property> element	<i>toleranceAmax</i> <Property>	Optional	Horizontal maximum tolerance in direction dirA (e.g. along referenced alignment, behind = non-positive number)
Zero or one <i>toleranceBmin</i> <Property> element	<i>toleranceBmin</i> <Property>	Optional	Horizontal minimum tolerance in direction perpendicular to dirA (e.g. towards/from referenced alignment, left = non-positive number)
Zero or one <i>toleranceBmax</i> <Property> element	<i>toleranceBmax</i> <Property>	Optional	Horizontal maximum tolerance in direction perpendicular to dirA (e.g. towards/from referenced alignment, left = non-positive number)
Zero or one <i>toleranceZmin</i> <Property> element	<i>toleranceZmin</i> <Property>	Optional	Height minimum tolerance below set elevation (non-positive number)
Zero or one <i>toleranceZmax</i> <Property> element	<i>toleranceZmax</i> <Property>	Optional	Height maximum tolerance below set elevation (non-positive number)
Zero or one <i>differenceXY</i> <Property> element	<i>differenceXY</i> <Property>	Optional	Horizontal difference between planned and measured point
Zero or one <i>dirDifferenceXY</i> <Property> element	<i>dirDifferenceXY</i> <Property>	Optional	Direction of horizontal difference between planned and measured point

Field	Type	Use	Description
Zero or one <i>differenceA</i> <Property> element	<i>differenceA</i> <Property>	Optional	Horizontal difference in direction dirA between planned and measured point
Zero or one <i>differenceB</i> <Property> element	<i>differenceB</i> <Property>	Optional	Tolerance in horizontal plane from set northing easting in distance units (non-negative number)
Zero or one <i>differenceZ</i> <Property> element	<i>differenceZ</i> <Property>	Optional	Tolerance in horizontal plane from set northing easting in distance units (non-negative number)
Zero or one <i>alignmentRef</i> <Property> element	<i>alignmentRef</i> <Property>	Optional	Reference to <Alignment>.name
Zero or one <i>dirA</i> <Property> element	<i>dirA</i> <Property>	Optional	Direction of positive A tolerance, derived from direction of referenced alignment
Zero or one <i>geometryType</i> <Property> element	<i>geometryType</i> <Property>	Optional	Type of geometry that the control point is part of
code	<i>string</i>	Required	Fixed value, IM_cgpoints
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

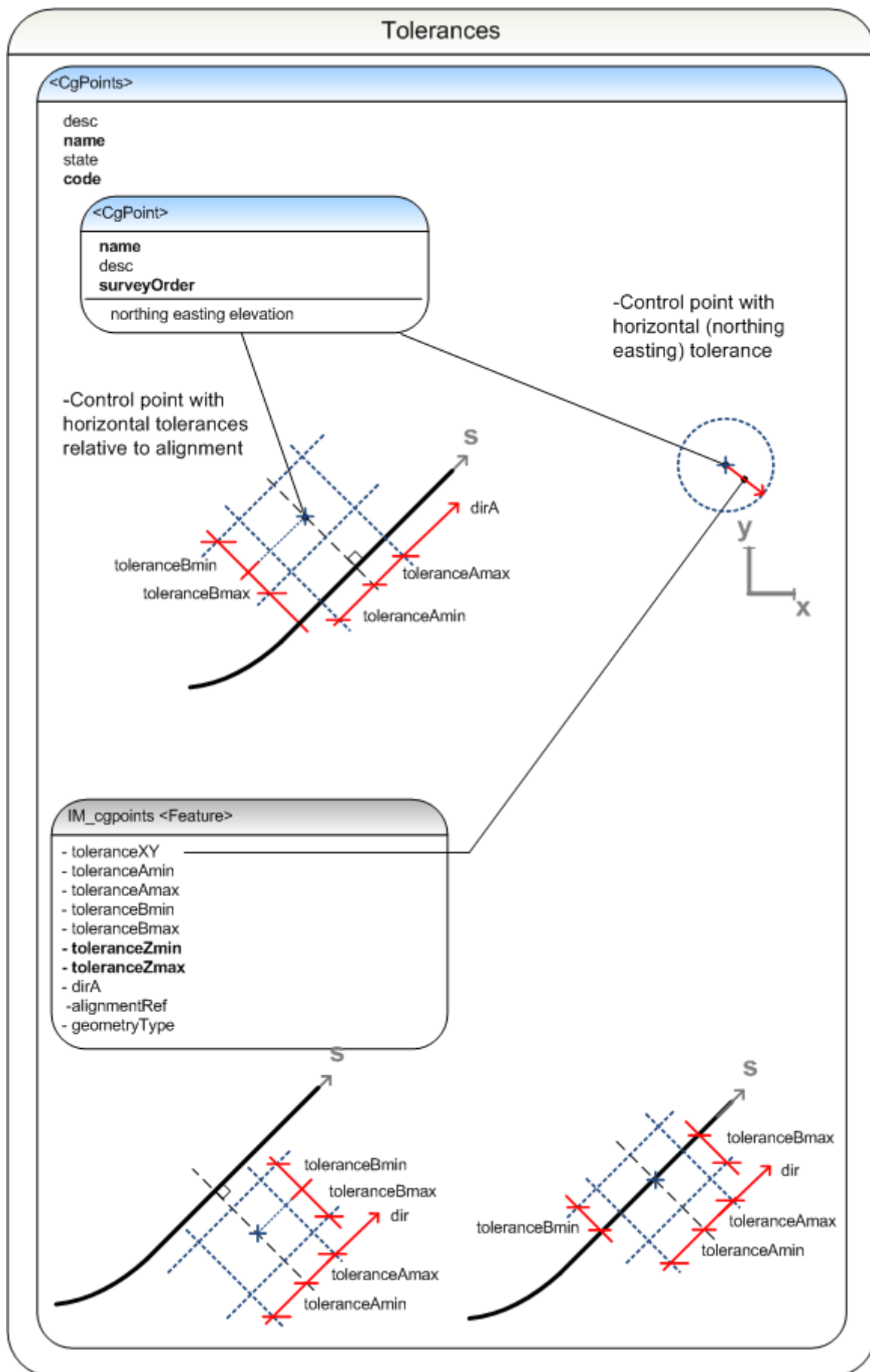


Figure 12.1: As-built plan

12.2 As-built survey

Measured as-built data is grouped in survey collection for each surface under **<Survey>** element.

A **<Survey>** shall have the fields defined in Table 12.4:

Table 12.4: <Survey> fields.

Field	Type	Use	Description
One <SurveyHeader> element	<SurveyHeader>	Required	Survey details
Zero or one <Equipment> element	<Equipment>	Optional	Survey equipment
Zero or one <CgPoints> element	<CgPoints>	Optional	A collection of COGO points. (Cg = COGO = Coordinate Geometry)
Zero or more IM_userDefinedProperties <Feature> elements	IM_userDefinedProperties <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
desc	string	Optional	Description
date	date	Optional	Survey date
startTime	datetime	Optional	Survey start time
endTime	datetime	Optional	Survey end time
state	stateType	Optional	State

A <SurveyHeader> shall have the fields defined in Table 12.5:

Table 12.5: <SurveyHeader> fields.

Field	Type	Use	Description
name	nameType	Optional	
desc	string	Optional	
purpose	purposeType	Optional	
surveyor	string	Optional	
surveyorFirm	string	Optional	
surveyPurpose	string	Optional	
type	surveyType	Optional	
submissionDate	date	Optional	

12.2.1 Survey equipment

Under each **<Survey>** element, an **<Equipment>** element (no attributes) shall specify the details of the survey instrument used as **<InstrumentDetails>**, with attributes **id** (mandatory identification) and optional **manufacturer**, **model** and **serialNumber**. These details are extended further under **<Corrections>** element (no attributes used in Inframodel) in “IM_survey” extension **<Feature>**.

An **<Equipment>** shall have the fields defined in Table 12.6:

Table 12.6: <Equipment> fields.

Field	Type	Use	Description
One <InstrumentDetails> element	<InstrumentDetails>	Required	Survey instrument details

An **<InstrumentDetails>** shall have the fields defined in Table 12.7:

Table 12.7: <InstrumentDetails> fields.

Field	Type	Use	Description
Zero or one IM_survey <Feature> element	IM_survey <Feature>	Optional	Inframodel survey equipment coding <Feature> extension
id	<i>id</i>	Required	Identifier
manufacturer	<i>string</i>	Optional	Manufacturer
model	<i>string</i>	Optional	Model
serialNumber	<i>string</i>	Optional	Serial number

An **IM_survey <Feature>** shall have the fields defined in Table 12.8:

Table 12.8: IM_survey <Feature> fields.

Field	Type	Use	Description
One accuracyType <Property> element	accuracyType <Property>	Required	Survey method and its accuracy (from Finnish “Infra Rakentajakoodaus” code list)
One accuracyDescription <Property> element	accuracyDescription <Property>	Required	Survey method and its accuracy description, ie. “gnss guided excavator”
code	<i>string</i>	Required	Fixed value, IM_survey

Field	Type	Use	Description
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

12.2.2 Survey points

The survey points belonging to the survey are grouped under . element with attributes described above in section **12.2 As-built survey** the name is used to identify the corresponding collection of control points, and code can be set to “survey” at the root of nested collections.

Each survey point may use pntRef to identify the corresponding control point (set as explained in subsection **12.2.1 Survey equipment**).

It is also recommended to set surveyOrder to survey sequence number (if known).

Deviation values may be set with “IM_cgpoints” <Feature> extension

Type <CgPoints> is defined per Table 12.1 above.

Type <CgPoint> is defined per Table 12.2 above.

An IM_cgpoints <Feature> shall have the fields defined in Table 12.9:

Table 12.9: IM_cgpoints <Feature> fields.

Field	Type	Use	Description
Zero or one <i>differenceXY</i> <Property> element	<i>differenceXY</i> <Property>	Optional	Horizontal difference between planned and measured point
Zero or one <i>dirDifferenceXY</i> <Property> element	<i>dirDifferenceXY</i> <Property>	Optional	Direction of horizontal difference between planned and measured point
Zero or one <i>differenceA</i> <Property> element	<i>differenceA</i> <Property>	Optional	Horizontal difference in direction dirA between planned and measured point
Zero or one <i>differenceB</i> <Property> element	<i>differenceB</i> <Property>	Optional	Tolerance in horizontal plane from set northing easting in distance units (non-negative number)
Zero or one <i>differenceZ</i> <Property> element	<i>differenceZ</i> <Property>	Optional	Tolerance in horizontal plane from set northing easting in distance units (non-negative number)

Field	Type	Use	Description
Zero or one <i>alignmentRef</i> <Property> element	<i>alignmentRef</i> <Property>	Optional	Reference to <Alignment>.name
Zero or one <i>dirA</i> <Property> element	<i>dirA</i> <Property>	Optional	Direction of positive A tolerance, derived from direction of referenced alignment
Zero or one <i>geometryType</i> <Property> element	<i>geometryType</i> <Property>	Optional	Type of geometry that the control point is part of
code	<i>string</i>	Required	Fixed value, IM_cgpoints
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

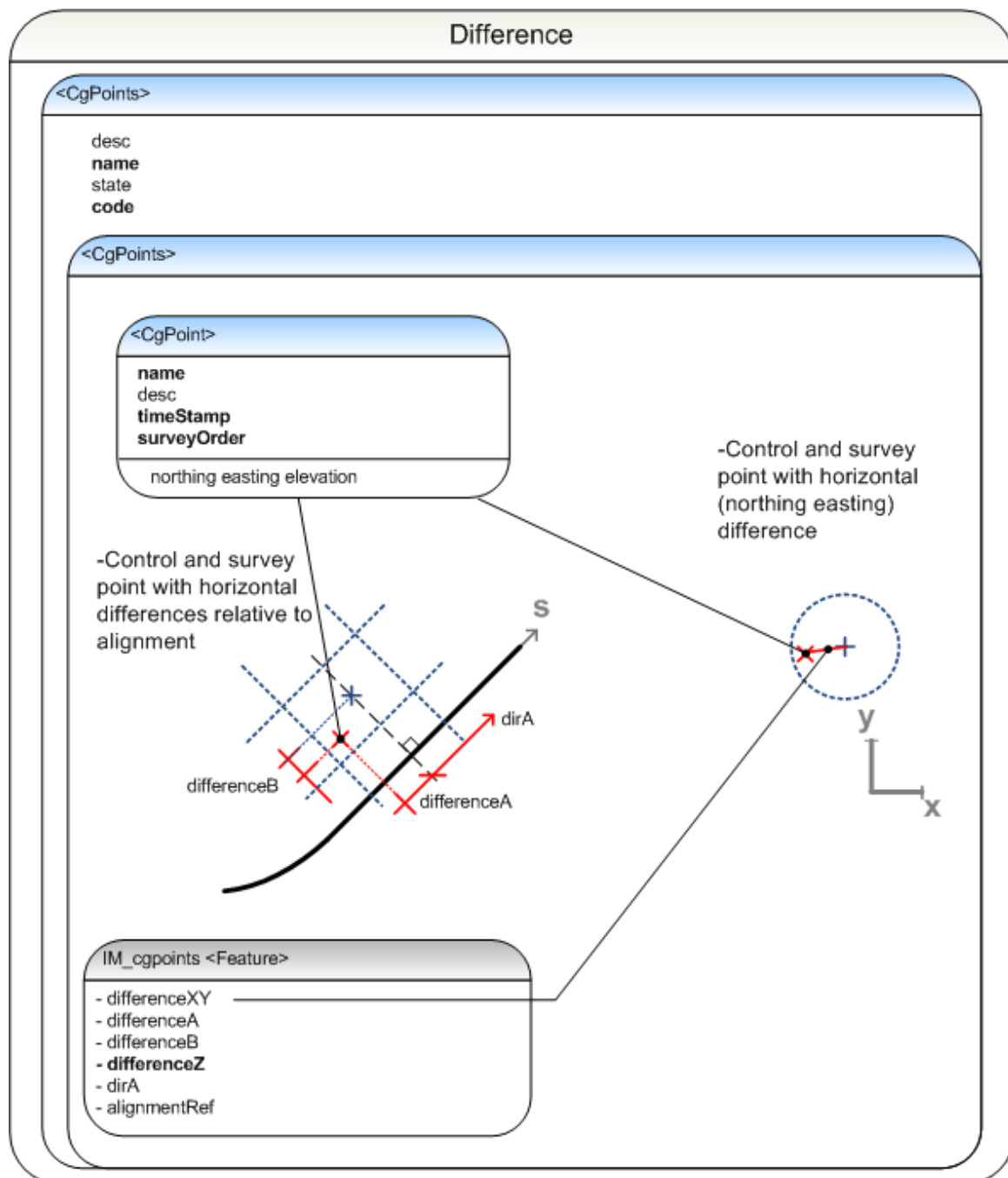


Figure 12.2: As-built survey

13 Deep foundations

An Inframodel file of any plan contains the header information described in section 1 *File headers*.

This section describes deep foundations (such as pile groups) transferred using **<any>** element under **<LandXML>**. The xml schema (im.xsd) for these extensions is available at Inframodel schema page.

13.1 Pile groups

Collection of **pile groups** in a soil stabilization plan.

An *<im:PileGroups>* shall have the fields defined in Table 13.1:

Table 13.1: *<im:PileGroups>* fields.

Field	Type	Use	Description
At least one <i><im:PileGroup></i> element	<i><im:PileGroup></i>	Required	Group of piles
Zero or more <i><Feature></i> elements	<i><Feature></i>	Optional	Used to include additional information that is not explicitly defined by the LandXML schema. Each Property element defines one piece of data.
name	<i>string</i>	Optional	Unique name
desc	<i>string</i>	Optional	Description

13.2 Pile group

Each individual pile group in a pile groups collection representing a stabilization plan is described under **<im:PileGroup>** element. Each pile group shall have an individual identifier **oID**, and may have a **name** and **description**. A number of other attributes may be used to describe the material properties of the pilings; spacing between piles (center to center) in the grid can be specified in **pileSpacing** attribute. The size of an individual pile can be specified in subelement **<im:CircPile>** or **<im:SquarePile>**, respectively (choice by the type of pile cross section).

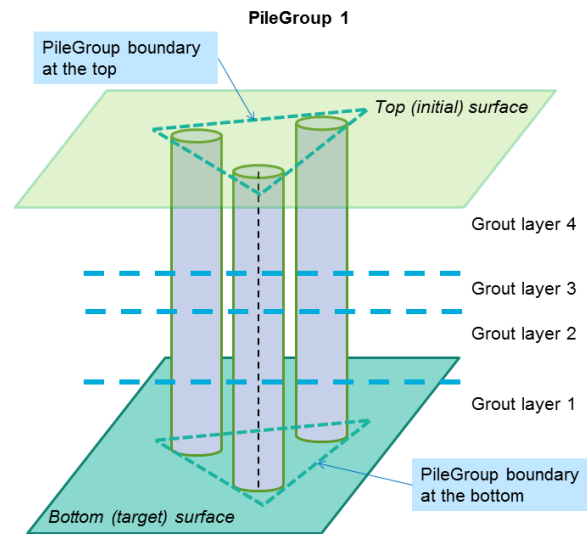


Figure 13.1: Pile group

An *<im:PileGroup>* shall have the fields defined in Table 13.2:

Table 13.2: *<im:PileGroup>* fields.

Field	Type	Use	Description
One <i><TopBoundary></i> element	<i><TopBoundary></i>	Required	A sequential space delimited list of 2D coordinates with a minimum of 3 points (6 values).
One <i><BottomBoundary></i> element	<i><BottomBoundary></i>	Required	A sequential space delimited list of 2D coordinates with a minimum of 3 points (6 values).
One <i><im:CircPile></i> or <i><im:SquarePile></i> element	<i><im:CircPile></i> or <i><im:SquarePile></i>	Optional	Circular or square pile
Zero or more <i><im:Pile></i> elements	<i><im:Pile></i>	Optional	Individual pile

Field	Type	Use	Description
Zero or more <Feature> elements	<Feature>	Optional	Used to include additional information that is not explicitly defined by the LandXML schema. Each Property element defines one piece of data.
oID	string	Required	Pile ID
name	string	Optional	Unique name
desc	string	Optional	Description
targetShearResistance	string	Optional	Pile target shear resistance. Unit Kpa.
groutType	string	Optional	Pile grout type
groutIntensity	string	Optional	Pile grout intensity. Unit kg/m3.
withdrawalRate	string	Optional	Pile withdrawal augering rate. Unit mm/r.
minRotationRate	string	Optional	Minimum rotation rate. Unit r/min.
maxRotationRate	string	Optional	Maximum rotation rate. Unit r/min.
minInjectionPressure	double	Optional	Minimum grout injection pressure. Unit bar.
maxInjectionPressure	double	Optional	Maximum grout injection pressure. Unit bar.
airInjectionRate	string	Optional	Air injection pressure. Unit l/min.
pileSpacing	string	Optional	Pile spacing (center to center). Unit m.

The size of an individual pile in a pile group is set in <im:CircPile> or <im:SquarePile>, respectively. All piles in a group are expected to share the same shape and size.

An <im:CircPile> shall have the fields defined in Table 13.3:

Table 13.3: <im:CircPile> fields.

Field	Type	Use	Description
diameter	<i>string</i>	Required	Diameter

An <im:SquarePile> shall have the fields defined in Table 13.4:

Table 13.4: <im:SquarePile> fields.

Field	Type	Use	Description
width	<i>string</i>	Required	Width

The top and bottom boundaries of a pile group are given in **<im:TopBoundary>** and **<im:BottomBoundary>** elements as irregular lines, defined by point lists (min. 3 points with 2D northing easting coordinates separated by spaces); the delimiting surfaces at the top and at the bottom may set by reference in **surfaceRef** attribute to the **<Surface>** element with geometry definition, or the coding of the initial and target surfaces may be given in “IM_coding” **<im:Feature>**.

13.3 Pile

Individual piles in a pile group are described under **<im:Pile>** element. Each pile shall have an individual identifier **oID** and **status**, as well as optional **topPoint**, **bottomPoint**, **horizontalDirection** and **slope**.

An <im:Pile> shall have the fields defined in Table 13.5:

Table 13.5: <im:Pile> fields.

Field	Type	Use	Description
Zero or more <im:GroutLayer> elements	<im:GroutLayer>	Optional	Pile grout layers
oID	<i>string</i>	Required	Pile ID
status	<i>string</i>	Required	Pile status (started, finished, under construction, ...)
topPoint	<i>string</i>	Optional	top centre point of pile
bottomPoint	<i>string</i>	Optional	bottom centre point of pile
horizontalDirection	<i>string</i>	Optional	Pile horizontal direction. Unit deg.
slope	<i>string</i>	Optional	Pile slope. Unit deg.

An `<im:GroutLayer>` shall have the fields defined in Table 13.6:

Table 13.6: `<im:GroutLayer>` fields.

Field	Type	Use	Description
startLevel	<i>string</i>	Optional	start level of grout layer in elevationUnits
endLevel	<i>string</i>	Optional	end level of grout layer in elevationUnits
groutIntensity	<i>string</i>	Optional	grout intensity of piles in this pile group, as measure of weight by volume (kg/m3)

14 Inframodel <Feature> extensions

Inframodel transfer files fully conform to the LandXML v1.2 schema (with one exception*), but some extensions have been made using the Feature-mechanism.

This section lists these Inframodel extensions, providing an index to the Inframodel Feature Dictionary: the **<FeatureDictionary>** element in Inframodel transfer file with the **name** “*inframodel*” (specifying the **<Feature>** elements in the file with attribute **source** as “*inframodel*” and the attribute **code** being labeled with “*IM_*” -prefix).

NB: In addition to these extensions, Inframodel specifies many restrictions on the use of LandXML elements and their attributes. These restrictions are described in this document. Also, further extensions have been specified in separate schema (im.xsd) for 1) Local Coordinate Transformation (section **1.6** *Type coding systems*) 2) Metadata (section **1.10** *Metadata*) and 3) Deep Foundations (section **13** *Deep foundations*).

* LandXML v1.2 <choice> declaration in <Roadways> collection has been changed to <sequence> in Inframodel v4.1.0, whereby both <Roadway> elements and feature extensions may appear in the same <Roadways> collection.

14.1 Local coordinate transformation definition by point pairs

Local coordinate system may be defined as set of control points sourceCRS-targetCRS point pairs under “*IM_coordTransformation*” <Feature> extension.

An *IM_coordTransformation* <Feature> shall have the fields defined in Table 14.1:

Table 14.1: *IM_coordTransformation* <Feature> fields.

Field	Type	Use	Description
One <i>sourceCRSname</i> <Property> element	<i>sourceCRSname</i> <Property>	Required	Name of the source geographic coordinate system
One <i>sourceEPSGcode</i> <Property> element	<i>sourceEPSGcode</i> <Property>	Required	EPSG code of the source geographic coordinate system
At least 3 <i>IM_controlPoint</i> <Feature> elements	<i>IM_controlPoint</i> <Feature>	Required	Inframodel localization control point <Feature> extension
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, <i>IM_</i> -coordTransformation
source	<i>string</i>	Required	Fixed value, <i>inframodel</i>

Where:

A *sourceCRSname* <Property> shall have the fields defined in Table 14.2:

Table 14.2: sourceCRSname <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, sourceCRSname
value	<i>string</i>	Required	Source CRS name, ie. WGS84

A *sourceEPSGcode* <Property> shall have the fields defined in Table 14.3:

Table 14.3: sourceEPSGcode <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, sourceEPSGcode
value	<i>string</i>	Required	EPSG code of source CRS, ie. 4326

An *IM_controlPoint* <Feature> shall have the fields defined in Table 14.4:

Table 14.4: IM_controlPoint <Feature> fields.

Field	Type	Use	Description
One <i>useHorizontal</i> <Property> element	<i>useHorizontal</i> <Property>	Required	Choose if the point should be used for calculation of horizontal transformation.
One <i>useVertical</i> <Property> element	<i>useVertical</i> <Property>	Required	Choose if the point should be used for calculation of vertical transformation.
One <i>latitude</i> <Property> element	<i>latitude</i> <Property>	Required	Sets point latitude in source CRS in radians
One <i>longitude</i> <Property> element	<i>longitude</i> <Property>	Required	Sets point longitude in source CRS in radians
One <i>altitude</i> <Property> element	<i>altitude</i> <Property>	Required	Sets point altitude value in source CRS in meters
One <i>northing</i> <Property> element	<i>northing</i> <Property>	Required	Sets point northing coordinate value in target system in meters
One <i>easting</i> <Property> element	<i>easting</i> <Property>	Required	Sets point easting coordinate value in target system in meters

Field	Type	Use	Description
One <i>elevation</i> <Property> element	<i>elevation</i> <Property>	Required	Sets point elevation value in target system in meters
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, IM_controlPoint
source	<i>string</i>	Required	Fixed value, inframodel

where:

An *useHorizontal* <Property> shall have the fields defined in Table 14.5:

Table 14.5: *useHorizontal* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>useHorizontal</i>
value	<i>one of (yes, no)</i>	Required	One of (yes,no). Set to [yes] if the point should be used for calculating the horizontal transformation

An *useVertical* <Property> shall have the fields defined in Table 14.6:

Table 14.6: *useVertical* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>useVertical</i>
value	<i>one of (yes, no)</i>	Required	One of (yes,no). Set to [yes] if the point should be used for calculating the vertical transformation

A *latitude* <Property> shall have the fields defined in Table 14.7:

Table 14.7: *latitude* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, latitude
value	<i>float</i>	Required	Latitude value in radians

A *longitude* <Property> shall have the fields defined in Table 14.8:

Table 14.8: longitude <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, longitude
value	<i>float</i>	Required	Longitude value in radians

An *altitude* <Property> shall have the fields defined in Table 14.9:

Table 14.9: altitude <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, altitude
value	<i>float</i>	Required	Altitude in meters in source CRS

A *northing* <Property> shall have the fields defined in Table 14.10:

Table 14.10: northing <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, northing
value	<i>float</i>	Required	Northing coordinate in target system in meters

An *easting* <Property> shall have the fields defined in Table 14.11:

Table 14.11: easting <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, easting
value	<i>float</i>	Required	Easting coordinate in target sytem in meters

An *elevation* <Property> shall have the fields defined in Table 14.12:

Table 14.12: elevation <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, elevation

Field	Type	Use	Description
value	<i>float</i>	Required	Elevation in target system in meters

14.2 Type coding systems

Type coding systems define the purpose of geometric elements (points, lines and surfaces) as well as other types of elements (such as plan features, pipes and structures in pipe networks, etc.) in Inframodel file transfers.

The information is set in two phases:

1. The type coding systems are declared in the project information **<Project>** using “IM_codings” and/or “IM_proprietaryCodings” feature extension and
2. individual type codes and their descriptions are set under each element using “IM_coding” and/or “IM_proprietaryCoding” feature extension.

The main systems set in the project information using “IM_codings” are:

1. The terrain coding system (**terrainCoding**) usually existing breaklines and points of interest on visible surfaces
2. The surface coding system (**surfaceCoding**) usually individual existing or planned surfaces
3. Infra object type coding system (**infraCoding**) usually individual planned objects and features, not modelled as surfaces.

It is possible to set the same system for several categories. It is also possible to set alternative or additional type coding systems (e.g. InfraRYL) for use within an organization or in a software used.

4. Optional, alternative or additional type coding systems (individually named **proprietaryInfraCoding**) set by defining one or more “IM_proprietaryCodings” <Feature>.

An *IM_codings* <Feature> shall have the fields defined in Table 14.13:

Table 14.13: *IM_codings* <Feature> fields.

Field	Type	Use	Description
One <i>terrainCoding</i> <Property> element	<i>terrainCoding</i> <Property>	Required	Sets terrain coding system
One <i>terrainCodingDesc</i> <Property> element	<i>terrainCodingDesc</i> <Property>	Required	Terrain coding system description
Zero or one <i>terrainCodingSourceRef</i> <Property> element	<i>terrainCodingSourceRef</i> <Property>	Optional	Sets terrain coding system source, ie. vayla.fi
One <i>surfaceCoding</i> <Property> element	<i>surfaceCoding</i> <Property>	Required	Surface coding system
One <i>surfaceCodingDesc</i> <Property> element	<i>surfaceCodingDesc</i> <Property>	Required	Surface coding system description

Field	Type	Use	Description
Zero or one <i>surfaceCodingSourceRef</i> <Property> element	<i>surfaceCodingSourceRef</i> <Property>	Optional	Surface coding system source
One <i>infraCoding</i> <Property> element	<i>infraCoding</i> <Property>	Required	Feature coding system
One <i>infraCodingDesc</i> <Property> element	<i>infraCodingDesc</i> <Property>	Required	Feature coding system description
Zero or one <i>infraCodingSourceRef</i> <Property> element	<i>infraCodingSourceRef</i> <Property>	Optional	Feature coding system source
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, IM_codings
source	<i>string</i>	Required	Fixed value, inframodel

where:

A *terrainCoding* <Property> shall have the fields defined in Table 14.14:

Table 14.14: *terrainCoding* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>terrainCoding</i>
value	<i>string</i>	Required	Terrain coding system name, ie. Infra

A *terrainCodingDesc* <Property> shall have the fields defined in Table 14.15:

Table 14.15: *terrainCodingDesc* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>terrainCodingDesc</i>
value	<i>string</i>	Required	Terrain coding system description, ie. Finnish Transport Infrastructure Agency terrain coding

A *terrainCodingSourceRef* <Property> shall have the fields defined in Table 14.16:

Table 14.16: terrainCodingSourceRef <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, terrainCodingSourceRef
value	string	Required	Terrain coding source, ie. vayla.fi

A *surfaceCoding* <Property> shall have the fields defined in Table 14.17:

Table 14.17: surfaceCoding <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, surfaceCoding
value	string	Required	Surface coding system name, ie. InfraBIM

A *surfaceCodingDesc* <Property> shall have the fields defined in Table 14.18:

Table 14.18: surfaceCodingDesc <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, surfaceCodingDesc
value	string	Required	Surface coding system description, ie. Finnish InfraBIM surface coding

A *surfaceCodingSourceRef* <Property> shall have the fields defined in Table 14.19:

Table 14.19: surfaceCodingSourceRef <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, surfaceCodingSourceRef
value	string	Required	Surface coding source, ie. buildingsmart.fi

An *infraCoding* <Property> shall have the fields defined in Table 14.20:

Table 14.20: infraCoding <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, infraCoding

Field	Type	Use	Description
value	<i>string</i>	Required	Feature coding system name, ie. InfraBIM

An *infraCodingDesc* <Property> shall have the fields defined in Table 14.21:

Table 14.21: *infraCodingDesc* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>infraCodingDesc</i>
value	<i>string</i>	Required	Feature coding system description, ie. Finnish InfraBIM coding

An *infraCodingSourceRef* <Property> shall have the fields defined in Table 14.22:

Table 14.22: *infraCodingSourceRef* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>infraCodingSourceRef</i>
value	<i>string</i>	Required	Feature coding source, ie. buildingsmart.fi

Proprietary codelists are defined using “IM_proprietaryCodings” <Feature>

An *IM_proprietaryCodings* <Feature> shall have the fields defined in Table 14.23:

Table 14.23: *IM_proprietaryCodings* <Feature> fields.

Field	Type	Use	Description
One <i>proprietaryInfraCoding</i> <Property> element	<i>proprietaryInfraCoding</i> <Property>	Required	Proprietary feature coding system
One <i>proprietaryInfraCodingDesc</i> <Property> element	<i>proprietaryInfraCodingDesc</i> <Property>	Required	Proprietary feature coding system description
Zero or one <i>proprietaryInfraCodingSourceRef</i> <Property> element	<i>proprietaryInfraCodingSourceRef</i> <Property>	Optional	Proprietary feature coding system source
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, <i>IM_proprietaryCodings</i>

Field	Type	Use	Description
source	<i>string</i>	Required	Fixed value, inframodel

where:

A *proprietaryInfraCoding* <Property> shall have the fields defined in Table 14.24:

Table 14.24: *proprietaryInfraCoding* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>proprietaryInfraCoding</i>
value	<i>string</i>	Required	Proprietary feature coding system name, ie. by organization or software

A *proprietaryInfraCodingDesc* <Property> shall have the fields defined in Table 14.25:

Table 14.25: *proprietaryInfraCodingDesc* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>proprietaryInfraCodingDesc</i>
value	<i>string</i>	Required	Proprietary feature coding system description, ie. my own coding system

A *proprietaryInfraCodingSourceRef* <Property> shall have the fields defined in Table 14.26:

Table 14.26: *proprietaryInfraCodingSourceRef* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>proprietaryInfraCodingSourceRef</i>
value	<i>string</i>	Required	Proprietary feature coding source, ie. xy.com

More information can be found from section **1.6** *Type coding systems*

14.3 Type coding

Individual type codes are set for the following elements in inframodel file transfers:

- data point groups, in the <**DataPoints**> element
- breaklines in the <**BreakLine**> element

- surfaces, in the **<Surface>** element
- alignments, in the **<Alignment>** element
- string line layers, in the “**IM_stringlineLayers**” **<Feature>** element
- other infrastructures in appropriate elements, such as **<PlanFeature>**, **<Pipe>** or **<Struct>**

Individual type codes are set in the individual element, or in the parent element, whose children inherit the values. Type codes are set using the “IM_coding” feature extension, either as *terrain codes* **terrainCoding**, *surface codes* **surfaceCoding** or *object/feature codes* **infraCoding**. Alternative or additional type codes are set using the “IM_proprietaryCoding”, where the proprietary code is in **proprietaryInfraCoding** and the name of the coding system in **proprietaryInfraCodingSource** (as declared in **proprietaryInfraCoding** in “IM_proprietaryCodings”).

An *IM_coding* **<Feature>** shall have the fields defined in Table 14.27:

Table 14.27: *IM_coding* **<Feature>** fields.

Field	Type	Use	Description
Zero or one <i>terrainCoding</i> <Property> element	<i>terrainCoding</i> <Property>	Optional	Terrain code
Zero or one <i>terrainCodingDesc</i> <Property> element	<i>terrainCodingDesc</i> <Property>	Optional	Terrain code description
Zero or one <i>surfaceCoding</i> <Property> element	<i>surfaceCoding</i> <Property>	Optional	Surface code
Zero or one <i>surfaceCodingDesc</i> <Property> element	<i>surfaceCodingDesc</i> <Property>	Optional	Surface code description
Zero or one <i>infraCoding</i> <Property> element	<i>infraCoding</i> <Property>	Optional	Feature code
Zero or one <i>infraCodingDesc</i> <Property> element	<i>infraCodingDesc</i> <Property>	Optional	Feature code description
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, <i>IM_coding</i>
source	<i>string</i>	Required	Fixed value, <i>inframodel</i>

where:

A *terrainCoding* **<Property>** shall have the fields defined in Table 14.28:

Table 14.28: *terrainCoding* **<Property>** fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>terrainCoding</i>
value	<i>string</i>	Required	Terrain code

A *terrainCodingDesc* <Property> shall have the fields defined in Table 14.29:

Table 14.29: *terrainCodingDesc* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, terrainCodingSourceRef
value	string	Required	Terrain code

A *surfaceCoding* <Property> shall have the fields defined in Table 14.30:

Table 14.30: *surfaceCoding* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, surfaceCoding
value	string	Required	Surface code

A *surfaceCodingDesc* <Property> shall have the fields defined in Table 14.31:

Table 14.31: *surfaceCodingDesc* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, surfaceCoding
value	string	Required	Surface code

An *infraCoding* <Property> shall have the fields defined in Table 14.32:

Table 14.32: *infraCoding* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, infraCoding
value	string	Required	Feature code

An *infraCodingDesc* <Property> shall have the fields defined in Table 14.33:

Table 14.33: *infraCodingDesc* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, infraCodingDesc

Field	Type	Use	Description
value	string	Required	Feature code

An *IM_proprietaryCoding* <Feature> shall have the fields defined in Table 14.34:

Table 14.34: *IM_proprietaryCoding* <Feature> fields.

Field	Type	Use	Description
One <i>proprietaryInfraCoding</i> <Property> element	<i>proprietaryInfraCoding</i> <Property>	Required	Proprietary feature code
One <i>proprietaryInfraCodingDesc</i> <Property> element	<i>proprietaryInfraCodingDesc</i> <Property>	Required	Proprietary feature code description
Zero or one <i>proprietaryInfraCodingSource</i> <Property> element	<i>proprietaryInfraCodingSource</i> <Property>	Optional	Proprietary feature coding system source
name	<i>nameType</i>	Optional	Optional unique name
code	string	Required	Fixed value, <i>IM_proprietaryCodings</i>
source	string	Required	Fixed value, <i>inframodel</i>

where:

A *proprietaryInfraCoding* <Property> shall have the fields defined in Table 14.35:

Table 14.35: *proprietaryInfraCoding* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, <i>proprietaryInfraCoding</i>
value	string	Required	Proprietary feature code

A *proprietaryInfraCodingDesc* <Property> shall have the fields defined in Table 14.36:

Table 14.36: *proprietaryInfraCodingDesc* <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, <i>proprietaryInfraCodingDesc</i>
value	string	Required	Proprietary feature code description

Details:

subsection **2.1.2** *Type coding*

section **3.2** *Naming and Type coding*

section **10.4** *Structures*

section **10.5** *Pipes*

subsection **11.1.1.7** *Type coding*

14.4 User defined properties

Custom properties may be defined by using “IM_userDefinedProperties” <Feature>

An *IM_userDefinedProperties* <Feature> shall have the fields defined in Table 14.37:

Table 14.37: *IM_userDefinedProperties* <Feature> fields.

Field	Type	Use	Description
One <i>propertyLabel</i> <Property> element	<i>propertyLabel</i> <Property>	Required	User defined property name
One <i>propertyValue</i> <Property> element	<i>propertyValue</i> <Property>	Required	User defined property value
Zero or one <i>propertyDescription</i> <Property> element	<i>propertyDescription</i> <Property>	Optional	User defined property description
Zero or one <i>propertySource</i> <Property> element	<i>propertySource</i> <Property>	Optional	User defined property source
name	<i>nameType</i>	Optional	Optional unique name
code	<i>string</i>	Required	Fixed value, IM_userDefinedProperties
source	<i>string</i>	Required	Fixed value, inframodel

where:

A *propertyLabel* <Property> shall have the fields defined in Table 14.38:

Table 14.38: *propertyLabel* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, propertyLabel
value	<i>string</i>	Required	User defined property name, ie. Manufacturer

A *propertyValue* <Property> shall have the fields defined in Table 14.39:

Table 14.39: propertyValue <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, propertyValue
value	string	Required	User defined property value, ie. CompanyXY

A *propertyDescription* <Property> shall have the fields defined in Table 14.40:

Table 14.40: propertyDescription <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, propertyDescription
value	string	Required	User defined property description, ie. Manufacturer name

A *propertySource* <Property> shall have the fields defined in Table 14.41:

Table 14.41: propertySource <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, propertySource
value	string	Required	User defined property source, ie. companyxy.com

14.5 Plan information

The details of a plan are described for each project part:

- surfaces (data points, breaklines, surfaces), in the **<Surfaces>** element
- route planning (road, street, railway ja waterway planning), in the **<Alignments>** element
- water supply and sewerage, in the **<PipeNetworks>** element.

If the project consists of sub-projects that have different rates of progress, the plan contents of the file are divided into sub-projects according to the same division. The plan information contains the **planName**, **planCode**, the **planState** and the plan description **planDesc**. The plan state is described according to a scheme agreed on by the parties of the project.

An *IM_plan* <Feature> shall have the fields defined in Table 14.42:

Table 14.42: IM_plan <Feature> fields.

Field	Type	Use	Description
One <i>planName</i> <Property> element	<i>planName</i> <Property>	Required	Plan name
Zero or one <i>planCode</i> <Property> element	<i>planCode</i> <Property>	Optional	Plan code
Zero or one <i>planDesc</i> <Property> element	<i>planDesc</i> <Property>	Optional	Plan description
Zero or one <i>planState</i> <Property> element	<i>planState</i> <Property>	Optional	Plan state
code	<i>string</i>	Required	Fixed value, IM_plan
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *planName* <Property> shall have the fields defined in Table 14.43:

Table 14.43: *planName* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>planName</i>
value	<i>string</i>	Required	Plan name

A *planCode* <Property> shall have the fields defined in Table 14.44:

Table 14.44: *planCode* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>planCode</i>
value	<i>string</i>	Required	Plan code

A *planState* <Property> shall have the fields defined in Table 14.45:

Table 14.45: planState <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, planState
value	<i>one of (zoning, preliminary plan, general plan, plan, construction plan, construction , maintenance , removal)</i>	Required	Plan state

A *planDesc* <Property> shall have the fields defined in Table 14.46:

Table 14.46: planDesc <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, planDesc
value	<i>string</i>	Required	Additional information

Details:

section **2.1** *Plan information*

subsection **3.3.1** *Plan information*

subsection **10.1.1** *Plan information*

14.6 Quantity information

Calculated area of a surface (<**Surface**>) or the volume below (between two surfaces) can be transferred using “IM_quantity” extension. These quantities may also be assigned to a part of a surface (<**Surface**>.<**SourceData**>.<**Boundaries**>.<**Boundary**>) or an area defined as a <**Parcel**>.

An *IM_quantity* <Feature> shall have the fields defined in Table 14.47:

Table 14.47: IM_quantity <Feature> fields.

Field	Type	Use	Description
Zero or one <i>areaQuantity</i> <Property> element	<i>areaQuantity</i> <Property>	Optional	Area
Zero or one <i>volumeQuantity</i> <Property> element	<i>volumeQuantity</i> <Property>	Optional	Volume
Zero or one <i>quantityCoding</i> <Property> element	<i>quantityCoding</i> <Property>	Optional	Quantity code

Field	Type	Use	Description
Zero or one <i>quantityCodingDescription</i> <Property> element	<i>quantityCodingDescription</i> <Property>	Optional	Quantity code description
Zero or one <i>quantityCodingSource</i> <Property> element	<i>quantityCodingSource</i> <Property>	Optional	Quantity coding system name
Zero or one <i>description</i> <Property> element	<i>description</i> <Property>	Optional	Description or note
Zero or one <i>surfaceRef</i> <Property> element	<i>surfaceRef</i> <Property>	Optional	Reference to the name of <Surface> element (applicable if volume calculation is performed between two surface elements)
code	<i>string</i>	Required	Fixed value, IM_quantity
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

An *areaQuantity* <Property> shall have the fields defined in Table 14.48:

Table 14.48: *areaQuantity* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>areaQuantity</i>
value	<i>string</i>	Required	Area in square meters

A *volumeQuantity* <Property> shall have the fields defined in Table 14.49:

Table 14.49: *volumeQuantity* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>volumeQuantity</i>
value	<i>string</i>	Required	Area in qubic meters

A *quantityCoding* <Property> shall have the fields defined in Table 14.50:

Table 14.50: quantityCoding <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, quantityCoding
value	string	Required	Quantity code

A *quantityCodingDescription* <Property> shall have the fields defined in Table 14.51:

Table 14.51: quantityCodingDescription <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, quantityCodingDescription
value	string	Required	Quantity code description

A *quantityCodingSource* <Property> shall have the fields defined in Table 14.52:

Table 14.52: quantityCodingSource <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, quantityCodingSource
value	string	Required	Quantity coding system name, as defined under <Project> -> IM_proprietaryCodings<Feature>

14.7 Soil properties

Soil properties of terrain model or ground layer model are captured in “IM_soil” feature extension.

An *IM_soil* <Feature> shall have the fields defined in Table 14.53:

Table 14.53: IM_soil <Feature> fields.

Field	Type	Use	Description
Zero or one <i>soiltypeGEO</i> <Property> element	<i>soiltypeGEO</i> <Property>	Optional	Soil type classification according to Geotekninen maaluokitus (VTT 1974)

Field	Type	Use	Description
Zero or one <i>soiltypeISO</i> <Property> element	<i>soiltypeISO</i> <Property>	Optional	Soil type classification according to SFS-EN ISO 14688-1 or SFS-EN ISO 14688-2
Zero or one <i>qualificationClass</i> <Property> element	<i>qualificationClass</i> <Property>	Optional	Soil type qualification according to table 10 in TIEH 2100029-04
Zero or one <i>frostSwellingFactorWet</i> <Property> element	<i>frostSwellingFactorWet</i> <Property>	Optional	Soil type frost swelling factor in wet conditions
Zero or one <i>frostSwellingFactorDry</i> <Property> element	<i>frostSwellingFactorDry</i> <Property>	Optional	Soil type frost swelling factor in dry conditions
Zero or one <i>elasticModulusWet</i> <Property> element	<i>elasticModulusWet</i> <Property>	Optional	Soil type modulus of elasticity in wet conditions (MPa)
Zero or one <i>elasticModulusDry</i> <Property> element	<i>elasticModulusDry</i> <Property>	Optional	Soil type modulus of elasticity in dry conditions (MPa)
code	<i>string</i>	Required	Fixed value, IM_soil
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *soiltypeGEO* <Property> shall have the fields defined in Table 14.54:

Table 14.54: *soiltypeGEO* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, planDesc
value	<i>string</i>	Required	Soil type classification according to Geotekninen maaluokitus (VTT 1974)

A *soiltypeISO* <Property> shall have the fields defined in Table 14.55:

Table 14.55: soiltypeISO <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, planDesc
value	<i>string</i>	Required	Soil type classification according to SFS-EN ISO 14688-1 or SFS-EN ISO 14688-2

A *qualificationClass* <Property> shall have the fields defined in Table 14.56:

Table 14.56: qualificationClass <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, planDesc
value	<i>string</i>	Required	Soil type qualification according to table 10 in TIEH 2100029-04

A *frostSwellingFactorWet* <Property> shall have the fields defined in Table 14.57:

Table 14.57: frostSwellingFactorWet <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, frostSwellingFactorWet
value	<i>string</i>	Required	Soil type frost swelling factor in wet conditions

A *frostSwellingFactorDry* <Property> shall have the fields defined in Table 14.58:

Table 14.58: frostSwellingFactorDry <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, frostSwellingFactorDry
value	<i>string</i>	Required	Soil type frost swelling factor in dry conditions

An *elasticModulusWet* <Property> shall have the fields defined in Table 14.59:

Table 14.59: elasticModulusWet <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, elasticModulusWet
value	<i>string</i>	Required	Soil type modulus of elasticity in wet conditions (MPa)

An *elasticModulusDry* <Property> shall have the fields defined in Table 14.60:

Table 14.60: elasticModulusDry <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, elasticModulusDry
value	<i>string</i>	Required	Soil type modulus of elasticity in dry conditions (MPa)

Details:

section **2.4** *Terrain model*

section **2.5** *Ground layer model*

14.8 String line model

The string line model is composed of *line string alignments* <IrregularLine>. Their order with an <Alignment> is irrelevant. The string line model used in Inframodel is based on the Leica RoadRunner software.

The string line model is defined under an <Alignment> in the “**IM_stringlineLayers**” extension. The constituent line strings and their locations are set by layer in the “**IM_stringlineLayer**” child element. The order of description of the line strings does not matter, the alignments are identified by their unique name <Alignment>. name, which are listed in the element, separated by commas. A layer is assigned a **name** and optionally a **centerline**. When describing a layered structure the layers of the string line model are assigned surface codes **SurfaceCoding**. The same line string may belong to several different layers. Layer are listed starting from the top downwards.

An *IM_stringlineLayers* <Feature> shall have the fields defined in Table 14.61:

Table 14.61: IM_stringlineLayers <Feature> fields.

Field	Type	Use	Description
At least one <i>IM_stringlineLayer</i> <Feature> element	<i>IM_stringlineLayer</i> <Feature>	Required	Inframodel strigline layer <Feature> extension
code	<i>string</i>	Required	Fixed value, IM_stringlineLayers

Field	Type	Use	Description
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

An *IM_stringlineLayer* <Feature> shall have the fields defined in Table 14.62:

Table 14.62: *IM_stringlineLayer* <Feature> fields.

Field	Type	Use	Description
One <i>name</i> <Property> element	<i>name</i> <Property>	Required	Unique name
One <i>alignments</i> <Property> element	<i>alignments</i> <Property>	Required	List of alignments which are used to form the surface (sorted from left to right)
One <i>centreline</i> <Property> element	<i>centreline</i> <Property>	Required	Centre line of stringline layer
Zero or one <i>desc</i> <Property> element	<i>desc</i> <Property>	Optional	Description
Zero or one <i>IM_coding</i> <Feature> element	<i>IM_coding</i> <Feature>	Optional	Inframodel type coding <Feature> extension
Zero or more <i>IM_userDefinedProperties</i> <Feature> elements	<i>IM_userDefinedProperties</i> <Feature>	Optional	Inframodel user defined custom properties <Feature> extension
Zero or one <i>IM_proprietaryCoding</i> <Feature> element	<i>IM_proprietaryCoding</i> <Feature>	Optional	Inframodel proprietary feature coding <Feature> extension
code	<i>string</i>	Required	Fixed value, <i>IM_stringlineLayer</i>
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *name* <Property> shall have the fields defined in Table 14.63:

Table 14.63: name <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, name
value	<i>string</i>	Required	Unique name

A desc <Property> shall have the fields defined in Table 14.64:

Table 14.64: desc <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, desc
value	<i>string</i>	Required	Description

** ERROR: Type alignments_ltProperty__gt not found from ../schema/inframodel-raw.xsd! **

A centreline <Property> shall have the fields defined in Table 14.65:

Table 14.65: centreline <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, centreline
value	<i>string</i>	Required	Centre line of stringline layer

Details:

section **3.6** *String line model*

14.9 Cross-section parameters

Cross-section parameters contain parametric information considered crucial for each route type. They are set for the *stationing reference alignment* in the <CrossSects>. <CrossSect> “IM_crossSect” extension under the <Alignment> element. The chosen cross-section parameters are set in fields (<Property>). It is advisable to describe all parameters for each cross-section. If the some parameters change, the cross-section where the parameter begins to change and the end of the transition are described.

An *IM_crossSect* <Feature> shall have the fields defined in Table 14.66:

Table 14.66: IM_crossSect <Feature> fields.

Field	Type	Use	Description
Zero or one <i>pavementClass</i> <Property> element	<i>pavementClass</i> <Property>	Optional	Pavement class
Zero or one <i>pavementThickness</i> <Property> element	<i>pavementThickness</i> <Property>	Optional	Pamenent thickness
Zero or one <i>subgradeLoadCapacityClass</i> <Property> element	<i>subgradeLoadCapacityClass</i> <Property>	Optional	Subgrade load capacity class
Zero or one <i>slope</i> <Property> element	<i>slope</i> <Property>	Optional	Slope
Zero or one <i>tracks</i> <Property> element	<i>tracks</i> <Property>	Optional	Number of tracks
Zero or one <i>trackDist</i> <Property> element	<i>trackDist</i> <Property>	Optional	Track distance
Zero or one <i>thickness</i> <Property> element	<i>thickness</i> <Property>	Optional	Total thickness
Zero or one <i>bedWidth</i> <Property> element	<i>bedWidth</i> <Property>	Optional	Bed width
Zero or one <i>depth</i> <Property> element	<i>depth</i> <Property>	Optional	Waterway minimum depth
Zero or one <i>width</i> <Property> element	<i>width</i> <Property>	Optional	Waterway width
Zero or one <i>waterLevel</i> <Property> element	<i>waterLevel</i> <Property>	Optional	Dimensioning water level
code	<i>string</i>	Required	Fixed value, IM_crossSect
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *pavementClass* <Property> shall have the fields defined in Table 14.67:

Table 14.67: *pavementClass* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>pavementClass</i>
value	<i>string</i>	Required	Pavement class

A *pavementThickness* <Property> shall have the fields defined in Table 14.68:

Table 14.68: pavementThickness <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, pavementThickness
value	string	Required	Pavement thickness

A *subgradeLoadCapacityClass* <Property> shall have the fields defined in Table 14.69:

Table 14.69: subgradeLoadCapacityClass <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, subgradeLoadCapacityClass
value	string	Required	Subgrade load capacity

A *slope* <Property> shall have the fields defined in Table 14.70:

Table 14.70: slope <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, slope
value	string	Required	Slope, %, left to right, space as value separator

A *tracks* <Property> shall have the fields defined in Table 14.71:

Table 14.71: tracks <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, tracks
value	string	Required	Number of tracks

A *trackDist* <Property> shall have the fields defined in Table 14.72:

Table 14.72: trackDist <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, trackDist

Field	Type	Use	Description
value	<i>string</i>	Required	Track distance

A *bedWidth* <Property> shall have the fields defined in Table 14.73:

Table 14.73: *bedWidth* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>bedWidth</i>
value	<i>string</i>	Required	Bed width

A *depth* <Property> shall have the fields defined in Table 14.74:

Table 14.74: *depth* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>depth</i>
value	<i>string</i>	Required	Waterway minimum depth

A *waterLevel* <Property> shall have the fields defined in Table 14.75:

Table 14.75: *waterLevel* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>waterLevel</i>
value	<i>string</i>	Required	Dimensioning water level

Details:

subsection **4.3.1** *Cross-section parameters*

section **5.4** *Cross-sections and track information*

section **7.2** *Cross section parameters*

14.10 Structural layer properties

Material properties of layers in road, street or railway structural model are captured in “IM_structLayer” feature extension.

An *IM_structLayer* <Feature> shall have the fields defined in Table 14.76:

Table 14.76: IM_structLayer <Feature> fields.

Field	Type	Use	Description
Zero or one <i>material</i> <Property> element	<i>material</i> <Property>	Optional	Material name of the structural layer
Zero or one <i>grainSizeMin</i> <Property> element	<i>grainSizeMin</i> <Property>	Optional	Minimum grain size of the material
Zero or one <i>grainSizeMax</i> <Property> element	<i>grainSizeMax</i> <Property>	Optional	Maximum grain size of the material
Zero or one <i>nominalThickness</i> <Property> element	<i>nominalThickness</i> <Property>	Optional	Nominal thickness of the product
Zero or one <i>strength</i> <Property> element	<i>strength</i> <Property>	Optional	Material strength
Zero or one <i>E-module</i> <Property> element	<i>E-module</i> <Property>	Optional	E-module used in dimensioning road structure
Zero or one <i>loadCapacity</i> <Property> element	<i>loadCapacity</i> <Property>	Optional	Load-bearing capacity or the structure
Zero or one <i>pavementType</i> <Property> element	<i>pavementType</i> <Property>	Optional	Pavement type
Zero or one <i>surfacingMethod</i> <Property> element	<i>surfacingMethod</i> <Property>	Optional	Surfacing method
Zero or one <i>stoneStructureType</i> <Property> element	<i>stoneStructureType</i> <Property>	Optional	Stone structure type
Zero or one <i>stoneType</i> <Property> element	<i>stoneType</i> <Property>	Optional	Stone type
Zero or one <i>stabilizationMethod</i> <Property> element	<i>stabilizationMethod</i> <Property>	Optional	Stabilization method
Zero or one <i>lightweightFillMaterial</i> <Property> element	<i>lightweightFillMaterial</i> <Property>	Optional	Lightweight fill material
code	<i>string</i>	Required	Fixed value, IM_structLayer
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *material* <Property> shall have the fields defined in Table 14.77:

Table 14.77: material <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, material
value	string	Required	Material name of the structural layer according to applicable guideline, i.e: - Asfalttinormit 2023 (PANK Ry. 2023) - Tierakenteen suunnittelu (Väylä 2018) - Tietoa tiensuunnitteluun 71D (Tiehallinto 2005) - Paallysrakenteen stabilointi (Tiehallinto 2007) - Kevennysrakenteiden suunnittelu (Liikennevirasto 2011)

A *grainSizeMin* <Property> shall have the fields defined in Table 14.78:

Table 14.78: grainSizeMin <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, grainSizeMin
value	string	Required	Minimum grain size of the material in diameter units

A *grainSizeMax* <Property> shall have the fields defined in Table 14.79:

Table 14.79: grainSizeMax <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, grainSizeMax
value	string	Required	Maximum grain size of the material in diameter units

A *nominalThickness* <Property> shall have the fields defined in Table 14.80:

Table 14.80: nominalThickness <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, nominalThickness
value	string	Required	Nominal thickness of the product in meters

A *strength* <Property> shall have the fields defined in Table 14.81:

Table 14.81: strength <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, strength
value	string	Required	Material strength (MPa)

An *E_module* <Property> shall have the fields defined in Table 14.82:

Table 14.82: E_module <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, E-module
value	string	Required	E-module used in dimensioning road structure (MPa)

A *loadCapacity* <Property> shall have the fields defined in Table 14.83:

Table 14.83: loadCapacity <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, loadCapacity
value	string	Required	Load-bearing capacity or the structure (MPa)

A *pavementType* <Property> shall have the fields defined in Table 14.84:

Table 14.84: pavementType <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, pavementType
value	string	Required	Pavement type, ie. AB

A *surfacingMethod* <Property> shall have the fields defined in Table 14.85:

Table 14.85: surfacingMethod <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, surfacingMethod

Field	Type	Use	Description
value	string	Required	Surfacing method, ie. REM

A *stoneStructureType* <Property> shall have the fields defined in Table 14.86:

Table 14.86: stoneStructureType <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, stoneStructureType
value	string	Required	Stone laying pattern, ie. Suora ladonta

A *stoneType* <Property> shall have the fields defined in Table 14.87:

Table 14.87: stoneType <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, stoneType
value	string	Required	Stone type, ie. nupukivi

A *stabilizationMethod* <Property> shall have the fields defined in Table 14.88:

Table 14.88: stabilizationMethod <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, stabilizationMethod
value	string	Required	Stabilization method

A *lightweightFillMaterial* <Property> shall have the fields defined in Table 14.89:

Table 14.89: lightweightFillMaterial <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, lightweightFillMaterial
value	string	Required	Lightweight fill material, ie. vaahtolasi

Details:

subsection **4.4.1** *Structural layers*

section **5.5** *Terrain model and structural model of the track*

14.11 Road and street design - Road signs

Road signs belonging to a particular route design are described in **<Roadways>.<Roadway>.<Roadside>.<RoadSign>**, with detailed properties captured in “IM_roadSign” feature extension:

An *IM_roadSign* <Feature> shall have the fields defined in Table 14.90:

Table 14.90: IM_roadSign <Feature> fields.

Field	Type	Use	Description
Zero or one <i>signName</i> <Property> element	<i>signName</i> <Property>	Optional	Sign name
Zero or one <i>description</i> <Property> element	<i>description</i> <Property>	Optional	Description
One <i>signCode</i> <Property> element	<i>signCode</i> <Property>	Required	Sign code
Zero or one <i>sizeDesignation</i> <Property> element	<i>sizeDesignation</i> <Property>	Optional	Sign size designation
Zero or one <i>location</i> <Property> element	<i>location</i> <Property>	Optional	Sing location, space delimited 3d coordinate
Zero or one <i>mountType</i> <Property> element	<i>mountType</i> <Property>	Optional	Mounting type
Zero or one <i>material</i> <Property> element	<i>material</i> <Property>	Optional	Sign material
Zero or one <i>reflection</i> <Property> element	<i>reflection</i> <Property>	Optional	Reflection type
Zero or one <i>shape</i> <Property> element	<i>shape</i> <Property>	Optional	Sign shape
Zero or one <i>background</i> <Property> element	<i>background</i> <Property>	Optional	Sign background
Zero or one <i>state</i> <Property> element	<i>state</i> <Property>	Optional	Sign state
Zero or one <i>note</i> <Property> element	<i>note</i> <Property>	Optional	Note
code	<i>string</i>	Required	Fixed value, IM_roadSign
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *signName* <Property> shall have the fields defined in Table 14.91:

Table 14.91: *signName* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>signName</i>
value	<i>string</i>	Required	Sign name

A *signCode* <Property> shall have the fields defined in Table 14.92:

Table 14.92: *signCode* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>signCode</i>
value	<i>string</i>	Required	Sign code

A *sizeDesignation* <Property> shall have the fields defined in Table 14.93:

Table 14.93: *sizeDesignation* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>sizeDesignation</i>
value	<i>one of (Small, Regular, Large)</i>	Required	Sign size designation, ie. Regular

A *location* <Property> shall have the fields defined in Table 14.94:

Table 14.94: *location* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>location</i>
value	<i>string</i>	Required	Sing location as space delimited northing,easting and elevation

A *mountType* <Property> shall have the fields defined in Table 14.95:

Table 14.95: mountType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, mountType
value	<i>string</i>	Required	Mounting type

A *reflection* <Property> shall have the fields defined in Table 14.96:

Table 14.96: reflection <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, reflection
value	<i>string</i>	Required	Reflection type

A *shape* <Property> shall have the fields defined in Table 14.97:

Table 14.97: shape <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, shape
value	<i>one of (inverted triangular, triangular, circular, octagonal, rectangular, rhombic, other)</i>	Required	Sign shape

A *background* <Property> shall have the fields defined in Table 14.98:

Table 14.98: background <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, background
value	<i>string</i>	Required	Sign background

A *state* <Property> shall have the fields defined in Table 14.99:

Table 14.99: state <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, state
value	<i>string</i>	Required	Sign state

A note <Property> shall have the fields defined in Table 14.100:

Table 14.100: note <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, note
value	<i>string</i>	Required	Note

14.12 Railway design - KM post coordinates

To assign northing and easting coordinates to railway <Alignment>.<StationEquation>, the parameters are:

An *IM_kmPostCoords* <Feature> shall have the fields defined in Table 14.101:

Table 14.101: *IM_kmPostCoords* <Feature> fields.

Field	Type	Use	Description
One <i>northing</i> <Property> element	<i>northing</i> <Property>	Required	Northing coordinate value in target system in meters
One <i>easting</i> <Property> element	<i>easting</i> <Property>	Required	Easting coordinate value in target system in meters
code	<i>string</i>	Required	Fixed value, <i>IM_kmPostCoords</i>
source	<i>string</i>	Required	Fixed value, <i>inframodel</i>
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *northing* <Property> shall have the fields defined in Table 14.102:

Table 14.102: northing <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, northing
value	<i>float</i>	Required	Northing coordinate in target system in meters

An *easting* <Property> shall have the fields defined in Table 14.103:

Table 14.103: easting <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, easting
value	<i>float</i>	Required	Easting coordinate in target sytem in meters

Details:

section **5.3** *KM-posting*

14.13 Railway design - switches

Switch details at railway track <Alignment>.<CoordGeom>.<Line>, the parameters are:

An *IM_switch* <Feature> shall have the fields defined in Table 14.104:

Table 14.104: IM_switch <Feature> fields.

Field	Type	Use	Description
Zero or one <i>switchType</i> <Property> element	<i>switchType</i> <Property>	Optional	Switch type
Zero or one <i>switchHand</i> <Property> element	<i>switchHand</i> <Property>	Optional	Switch hand
Zero or one <i>switchJoint</i> <Property> element	<i>switchJoint</i> <Property>	Optional	Switch joint
code	<i>string</i>	Required	Fixed value, IM_switch
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *switchType* <Property> shall have the fields defined in Table 14.105:

Table 14.105: switchType <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, switchType
value	string	Required	Type of switch

A *switchHand* <Property> shall have the fields defined in Table 14.106:

Table 14.106: switchHand <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, switchHand
value	one of (left, right, both)	Required	Switch hand

A *switchJoint* <Property> shall have the fields defined in Table 14.107:

Table 14.107: switchJoint <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, centreline
value	string	Required	Centre line of stringline layer

Details:

subsection **5.4.3** *Switch information*

14.14 Utility networks - network type

When the attribute pipeNetType in >PipeNetwork> element is set to “other”, the type of utility network may be specified in “IM_pipeNetworkType” extension:

An *IM_pipeNetworkType* <Feature> shall have the fields defined in Table 14.108:

Table 14.108: IM_pipeNetworkType <Feature> fields.

Field	Type	Use	Description
One <i>pipeNetworkType</i> <Property> element	<i>pipeNetworkType</i> <Property>	Required	Pipe network type
code	string	Required	Fixed value, IM_pipeNetworkType

Field	Type	Use	Description
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *pipeNetworkType* <Property> shall have the fields defined in Table 14.109:

Table 14.109: *pipeNetworkType* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>pipeNetworkType</i>
value	<i>one of (districtheating , districtcooling, gas, waste transport piping, electric, telecommunication)</i>	Required	Pipe network type

14.15 Utility networks - structure details

It is possible to describe additional details of network structures described in inframodel file transfers. The parameters in “IM_struct” are:

An *IM_struct* <Feature> shall have the fields defined in Table 14.110:

Table 14.110: *IM_struct* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>structLabel</i> <Property> element	<i>structLabel</i> <Property>	Optional	Struct name or identifier
Zero or one <i>rimType</i> <Property> element	<i>rimType</i> <Property>	Optional	Rim type
Zero or one <i>rimMaterial</i> <Property> element	<i>rimMaterial</i> <Property>	Optional	Rim material
Zero or one <i>rimLoad</i> <Property> element	<i>rimLoad</i> <Property>	Optional	Rim load bearing capacity
Zero or one <i>rimDiameter</i> <Property> element	<i>rimDiameter</i> <Property>	Optional	Rim diameter
Zero or one <i>rimCenter</i> <Property> element	<i>rimCenter</i> <Property>	Optional	Rim centerpoint location
Zero or one <i>heightDeposit</i> <Property> element	<i>heightDeposit</i> <Property>	Optional	Deposit height

Field	Type	Use	Description
Zero or one <i>volumeDeposit</i> <Property> element	<i>volumeDeposit</i> <Property>	Optional	Deposit volume
Zero or one <i>constructionDate</i> <Property> element	<i>constructionDate</i> <Property>	Optional	Construction date
Zero or one <i>renewalDate</i> <Property> element	<i>renewalDate</i> <Property>	Optional	Renewal date
Zero or one <i>renewalDesc</i> <Property> element	<i>renewalDesc</i> <Property>	Optional	Renewal description
Zero or one <i>bottomThickness</i> <Property> element	<i>bottomThickness</i> <Property>	Optional	Well bottom thickness
Zero or one <i>equipmentType</i> <Property> element	<i>equipmentType</i> <Property>	Optional	Equipment type
Zero or one <i>equipmentDesc</i> <Property> element	<i>equipmentDesc</i> <Property>	Optional	Equipment description
Zero or one <i>equipmentCode</i> <Property> element	<i>equipmentCode</i> <Property>	Optional	Equipment code
Zero or one <i>equipmentCodeDesc</i> <Property> element	<i>equipmentCodeDesc</i> <Property>	Optional	Equipment code description
Zero or one <i>equipmentCodeSource</i> <Property> element	<i>equipmentCodeSource</i> <Property>	Optional	Equipment code source
Zero or one <i>surfaceProfile</i> <Property> element	<i>surfaceProfile</i> <Property>	Optional	Surface profile of struct/pipe
code	<i>string</i>	Required	Fixed value, IM_struct
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *structLabel* <Property> shall have the fields defined in Table 14.111:

Table 14.111: *structLabel* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, structLabel
value	<i>string</i>	Required	Struct label

A *rimType* <Property> shall have the fields defined in Table 14.112:

Table 14.112: rimType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, rimType
value	<i>one of (solid cover, grate , atrium grate, side inlet, drain cap)</i>	Required	Rim type, alternatives include [solid cover

A rimMaterial <Property> shall have the fields defined in Table 14.113:

Table 14.113: rimMaterial <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, rimMaterial
value	<i>one of (cast iron, stainless steel, concrete)</i>	Required	Rim material,alternatives include [stainless steel

A rimLoad <Property> shall have the fields defined in Table 14.114:

Table 14.114: rimLoad <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, rimLoad
value	<i>string</i>	Required	Rim load bearing capacity in kN, alternatives include e.g. [250kN

A rimDiameter <Property> shall have the fields defined in Table 14.115:

Table 14.115: rimDiameter <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, rimDiameter
value	<i>string</i>	Required	Rim diameter

A rimCenter <Property> shall have the fields defined in Table 14.116:

Table 14.116: rimCenter <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, rimCenter
value	string	Required	Rim centerpoint location as space delimited northing,east and elevation

A *heightDeposit* <Property> shall have the fields defined in Table 14.117:

Table 14.117: heightDeposit <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, heightDeposit
value	string	Required	Deposit height

A *volumeDeposit* <Property> shall have the fields defined in Table 14.118:

Table 14.118: volumeDeposit <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, volumeDeposit
value	string	Required	Deposit volume

A *constructionDate* <Property> shall have the fields defined in Table 14.119:

Table 14.119: constructionDate <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, constructionDate
value	string	Required	Construction date, yyyy or yyyyymm or yyyyymmdd

A *renewalDate* <Property> shall have the fields defined in Table 14.120:

Table 14.120: renewalDate <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, renewalDate

Field	Type	Use	Description
value	string	Required	Renewal date, yyyy or yyyyymm or yyyyymmdd

A *renewalDesc* <Property> shall have the fields defined in Table 14.121:

Table 14.121: renewalDesc <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, renewalDesc
value	string	Required	Renewal description, ie. cleansing

A *bottomThickness* <Property> shall have the fields defined in Table 14.122:

Table 14.122: bottomThickness <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, bottomThickness
value	string	Required	Well bottom thickness

An *equipmentType* <Property> shall have the fields defined in Table 14.123:

Table 14.123: equipmentType <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, equipmentType
value	string	Required	Equipment type

An *equipmentDesc* <Property> shall have the fields defined in Table 14.124:

Table 14.124: equipmentDesc <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, equipmentDesc
value	string	Required	Equipment description

An *equipmentCode* <Property> shall have the fields defined in Table 14.125:

Table 14.125: equipmentCode <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, equipmentCode
value	<i>string</i>	Required	Equipment code

An *equipmentCodeDesc* <Property> shall have the fields defined in Table 14.126:

Table 14.126: equipmentCodeDesc <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, equipmentCodeDesc
value	<i>string</i>	Required	Equipment code description

An *equipmentCodeSource* <Property> shall have the fields defined in Table 14.127:

Table 14.127: equipmentCodeSource <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, equipmentCodeSource
value	<i>string</i>	Required	Equipment code source

Details:

subsection **10.4.1** *Circular structures*

subsection **10.4.2** *Rectangular structures*

subsection **10.4.3** *Pipe inlets and outlets*

subsection **10.4.4** *Pipe connections*

subsection **10.4.5** *Equipment*

14.16 Utility networks - pipe details

It is possible to describe additional details of pipes of a network described in inframodel file transfers. The parameters in “IM_pipe” are:

An *IM_pipe* <Feature> shall have the fields defined in Table 14.128:

Table 14.128: IM_pipe <Feature> fields.

Field	Type	Use	Description
Zero or one <i>pipeLabel</i> <Property> element	<i>pipeLabel</i> <Property>	Optional	Pipe name or identifier
One <i>elevType</i> <Property> element	<i>elevType</i> <Property>	Required	Elevation type
Zero or one <i>pipeStart</i> <Property> element	<i>pipeStart</i> <Property>	Optional	Pipe start location. Note, this field is informational, actual pipe start point is set by <Struct> referenced by <Pipe>.refStart
Zero or one <i>pipeEnd</i> <Property> element	<i>pipeEnd</i> <Property>	Optional	Pipe end location. Note, this field is informational, actual pipe end point is set by <Struct> referenced by <Pipe>.refEnd
Zero or one <i>jointType</i> <Property> element	<i>jointType</i> <Property>	Optional	Pipe joint type
Zero or one <i>constructionDate</i> <Property> element	<i>constructionDate</i> <Property>	Optional	Construction date
Zero or one <i>renewalDate</i> <Property> element	<i>renewalDate</i> <Property>	Optional	Renewal date
Zero or one <i>renewalDesc</i> <Property> element	<i>renewalDesc</i> <Property>	Optional	Renewal description
Zero or one <i>strengthClass</i> <Property> element	<i>strengthClass</i> <Property>	Optional	Pipe strength class
Zero or one <i>surfaceProfile</i> <Property> element	<i>surfaceProfile</i> <Property>	Optional	Surface profile of struct/pipe
code	<i>string</i>	Required	Fixed value, IM_pipe
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *pipeLabel* <Property> shall have the fields defined in Table 14.129:

Table 14.129: pipeLabel <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, pipeLabel
value	<i>string</i>	Required	Pipe name or identifier

An *elevType* <Property> shall have the fields defined in Table 14.130:

Table 14.130: elevType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, elevType
value	<i>one of (crown level, obvert level, center level, invert level, bottom level)</i>	Required	Elevation type

A *pipeStart* <Property> shall have the fields defined in Table 14.131:

Table 14.131: pipeStart <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, pipeStart
value	<i>string</i>	Required	Location as space delimited northing, easting, elevation

A *pipeEnd* <Property> shall have the fields defined in Table 14.132:

Table 14.132: pipeEnd <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, pipeEnd
value	<i>string</i>	Required	Location as space delimited northing, easting, elevation

A *jointType* <Property> shall have the fields defined in Table 14.133:

Table 14.133: jointType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, jointType

Field	Type	Use	Description
value	<i>string</i>	Required	Pipe joint type, alternatives include: rubber seal, flange, connector, weld

A *strengthClass* <Property> shall have the fields defined in Table 14.134:

Table 14.134: *strengthClass* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>strengthClass</i>
value	<i>string</i>	Required	Pipe strength class

Details:

subsection **10.7.0.2** *Circular pipes*

subsection **10.7.1** *Egg pipes*

subsection **10.7.2** *Elliptic pipes*

subsection **10.7.3** *Rectangular pipes*

subsection **10.7.4** *Channels*

14.17 Utility networks - cable details

It is possible to describe additional details of cables of a network described in inframodel file transfers. The parameters in “IM_cable” are:

An *IM_cable* <Feature> shall have the fields defined in Table 14.135:

Table 14.135: *IM_cable* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>label</i> <Property> element	<i>label</i> <Property>	Optional	Cable label
Zero or one <i>owner</i> <Property> element	<i>owner</i> <Property>	Optional	Cable owner
Zero or one <i>networkType</i> <Property> element	<i>networkType</i> <Property>	Optional	Cable network type
Zero or one <i>cableType</i> <Property> element	<i>cableType</i> <Property>	Optional	Cable type
code	<i>string</i>	Required	Fixed value, <i>IM_cable</i>

Field	Type	Use	Description
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *label* <Property> shall have the fields defined in Table 14.136:

Table 14.136: label <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, label
value	<i>string</i>	Required	Label or other identifier

An *owner* <Property> shall have the fields defined in Table 14.137:

Table 14.137: owner <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, owner
value	<i>string</i>	Required	Cable owner

A *networkType* <Property> shall have the fields defined in Table 14.138:

Table 14.138: networkType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, networkType
value	<i>string</i>	Required	Cable network type

A *cableType* <Property> shall have the fields defined in Table 14.139:

Table 14.139: cableType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, cableType
value	<i>string</i>	Required	Cable type

Details:

subsection **11.1.1.1** *Cable*

14.18 Plan features

Planimetric features belonging to a particular route design are described in **<Roadways>.<Roadway>.<PlanFeature>**, or in other cases in **<PlanFeatures>.<PlanFeature>**. In addition of capability of being classified using “IM_coding” extension (and/or “IM_proprietaryCoding”) and having custom properties using “IM_userDefinedProperties”, detailed properties may be assigned by the type of plan feature:

An *IM_footing* **<Feature>** shall have the fields defined in Table 14.140:

Table 14.140: *IM_footing* **<Feature>** fields.

Field	Type	Use	Description
One <i>footingHeight</i> <Property> element	<i>footingHeight</i> <Property>	Required	Footing height
Zero or one <i>radiusAround</i> <Property> element	<i>radiusAround</i> <Property>	Optional	Footing radius
Zero or one <i>length</i> <Property> element	<i>length</i> <Property>	Optional	Footing length
Zero or one <i>lengthDir</i> <Property> element	<i>lengthDir</i> <Property>	Optional	Footing direction
Zero or one <i>width</i> <Property> element	<i>width</i> <Property>	Optional	Footing width
Zero or one <i>label</i> <Property> element	<i>label</i> <Property>	Optional	Footing label
Zero or one <i>footingMaterial</i> <Property> element	<i>footingMaterial</i> <Property>	Optional	Footing material
Zero or one <i>foundationType</i> <Property> element	<i>foundationType</i> <Property>	Optional	Foundation type
code	<i>string</i>	Required	Fixed value, <i>IM_footing</i>
source	<i>string</i>	Required	Fixed value, <i>inframodel</i>
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *footingHeight* **<Property>** shall have the fields defined in Table 14.141:

Table 14.141: footingHeight <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, footingHeight
value	string	Required	Footing height

A *footingMaterial* <Property> shall have the fields defined in Table 14.142:

Table 14.142: footingMaterial <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, footingMaterial
value	string	Required	Material

A *foundationType* <Property> shall have the fields defined in Table 14.143:

Table 14.143: foundationType <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, foundationType
value	one of (Shallow, Deep, Other)	Required	Foundation type

A *radiusAround* <Property> shall have the fields defined in Table 14.144:

Table 14.144: radiusAround <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, radiusAround
value	string	Required	Radius

A *length* <Property> shall have the fields defined in Table 14.145:

Table 14.145: length <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, length

Field	Type	Use	Description
value	<i>string</i>	Required	Length

A *lengthDir* <Property> shall have the fields defined in Table 14.146:

Table 14.146: *lengthDir* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>lengthDir</i>
value	<i>string</i>	Required	Direction

A *width* <Property> shall have the fields defined in Table 14.147:

Table 14.147: *width* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>width</i>
value	<i>string</i>	Required	Width

An *IM_railing* <Feature> shall have the fields defined in Table 14.148:

Table 14.148: *IM_railing* <Feature> fields.

Field	Type	Use	Description
One <i>label</i> <Property> element	<i>label</i> <Property>	Required	Railing label
One <i>railingType</i> <Property> element	<i>railingType</i> <Property>	Required	Railing type
One <i>purpose</i> <Property> element	<i>purpose</i> <Property>	Required	Purpose of railing
Zero or one <i>shockResistanceClass</i> <Property> element	<i>shockResistanceClass</i> <Property>	Optional	Shock resistance class
Zero or one <i>snowPlowingDurabilityCategory</i> <Property> element	<i>snowPlowingDurabilityCategory</i> <Property>	Optional	Category of snow plowing durability of the railing
Zero or one <i>normalizedWorkingWidth</i> <Property> element	<i>normalizedWorkingWidth</i> <Property>	Optional	Normalized working width of the railing

Field	Type	Use	Description
Zero or one <i>railingHeight</i> <Property> element	<i>railingHeight</i> <Property>	Optional	Height of the railing
Zero or one <i>railHeight</i> <Property> element	<i>railHeight</i> <Property>	Optional	Height of the guard rail or guide bar
Zero or one <i>railingMaterial</i> <Property> element	<i>railingMaterial</i> <Property>	Optional	Material of the railing
Zero or one <i>startType</i> <Property> element	<i>startType</i> <Property>	Optional	Type of railing start element
Zero or one <i>endType</i> <Property> element	<i>endType</i> <Property>	Optional	Type of railing end element
Zero or one <i>postSpacing</i> <Property> element	<i>postSpacing</i> <Property>	Optional	Post spacing
code	<i>string</i>	Required	Fixed value, IM_railing
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *railingType* <Property> shall have the fields defined in Table 14.149:

Table 14.149: *railingType* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, railingType
value	<i>string</i>	Required	Railing type ie. tubular spar railing

A *purpose* <Property> shall have the fields defined in Table 14.150:

Table 14.150: *purpose* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, purpose
value	<i>string</i>	Required	Purpose, ie. handrail

A *shockResistanceClass* <Property> shall have the fields defined in Table 14.151:

Table 14.151: shockResistanceClass <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, shockResistaceClass
value	string	Required	Shock resistance class

A *snowPlowingDurabilityCategory* <Property> shall have the fields defined in Table 14.152:

Table 14.152: snowPlowingDurabilityCategory <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, snowPlowingDurabilityCategory
value	string	Required	Snow plowing durability class

A *normalizedWorkingWidth* <Property> shall have the fields defined in Table 14.153:

Table 14.153: normalizedWorkingWidth <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, normalizedWorkingWidht
value	string	Required	Normalized working widht of the railing

A *railingHeight* <Property> shall have the fields defined in Table 14.154:

Table 14.154: railingHeight <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, railingHeight
value	string	Required	Height of the railing

A *railHeight* <Property> shall have the fields defined in Table 14.155:

Table 14.155: railHeight <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, railHeight

Field	Type	Use	Description
value	<i>string</i>	Required	Height of the guard rail or guide bar

A *railingMaterial* <Property> shall have the fields defined in Table 14.156:

Table 14.156: *railingMaterial* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>railingMaterial</i>
value	<i>string</i>	Required	Material of the railing

A *startType* <Property> shall have the fields defined in Table 14.157:

Table 14.157: *startType* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>startType</i>
value	<i>string</i>	Required	Type of railing start element

An *endType* <Property> shall have the fields defined in Table 14.158:

Table 14.158: *endType* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>endType</i>
value	<i>string</i>	Required	Type of railing end element

An *IM_fence* <Feature> shall have the fields defined in Table 14.159:

Table 14.159: *IM_fence* <Feature> fields.

Field	Type	Use	Description
One <i>label</i> <Property> element	<i>label</i> <Property>	Required	Fence label
One <i>fenceType</i> <Property> element	<i>fenceType</i> <Property>	Required	Type of fence
One <i>fenceHeight</i> <Property> element	<i>fenceHeight</i> <Property>	Required	Height of the fence

Field	Type	Use	Description
Zero or one <i>postMaterial</i> <Property> element	<i>postMaterial</i> <Property>	Optional	Material of posts
Zero or one <i>postShape</i> <Property> element	<i>postShape</i> <Property>	Optional	Shape of posts
Zero or one <i>postSpacing</i> <Property> element	<i>postSpacing</i> <Property>	Optional	Post spacing
code	<i>string</i>	Required	Fixed value, IM_fence
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *fenceType* <Property> shall have the fields defined in Table 14.160:

Table 14.160: *fenceType* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, fenceType
value	<i>string</i>	Required	Type of fence

A *fenceHeight* <Property> shall have the fields defined in Table 14.161:

Table 14.161: *fenceHeight* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, fenceHeight
value	<i>string</i>	Required	Height of the fence

A *postMaterial* <Property> shall have the fields defined in Table 14.162:

Table 14.162: *postMaterial* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, postMaterial
value	<i>string</i>	Required	Material of fence posts

A *postShape* <Property> shall have the fields defined in Table 14.163:

Table 14.163: *postShape* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>postShape</i>
value	<i>string</i>	Required	Shape of fence posts

A *postSpacing* <Property> shall have the fields defined in Table 14.164:

Table 14.164: *postSpacing* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>postSpacing</i>
value	<i>string</i>	Required	Post spacing

An *IM_surfaceStructure* <Feature> shall have the fields defined in Table 14.165:

Table 14.165: *IM_surfaceStructure* <Feature> fields.

Field	Type	Use	Description
One <i>material</i> <Property> element	<i>material</i> <Property>	Required	Surface material name
Zero or one <i>thickness</i> <Property> element	<i>thickness</i> <Property>	Optional	Surface material thickness
Zero or one <i>surfaceRef</i> <Property> element	<i>surfaceRef</i> <Property>	Optional	Reference to the name of <Surface> element
code	<i>string</i>	Required	Fixed value, <i>IM_surfaceStructure</i>
source	<i>string</i>	Required	Fixed value, <i>inframodel</i>
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *thickness* <Property> shall have the fields defined in Table 14.166:

Table 14.166: *thickness* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>postShape</i>

Field	Type	Use	Description
value	<i>string</i>	Required	Shape of fence posts

A *surfaceRef* <Property> shall have the fields defined in Table 14.167:

Table 14.167: *surfaceRef* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, postShape
value	<i>string</i>	Required	Shape of fence posts

An *IM_planfeature* <Feature> shall have the fields defined in Table 14.168:

Table 14.168: *IM_planfeature* <Feature> fields.

Field	Type	Use	Description
One <i>type</i> <Property> element	<i>type</i> <Property>	Required	Type of plan feature
One <i>material</i> <Property> element	<i>material</i> <Property>	Required	Material of the plan feature
code	<i>string</i>	Required	Fixed value, IM_planfeature
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *type* <Property> shall have the fields defined in Table 14.169:

Table 14.169: *type* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, type
value	<i>string</i>	Required	Type of plan feature

Details:

section **4.5** *Road signs and plan features*

section **5.6** *Rail plan features*

section **7.5** *Waterway plan features*

section 11 *Planimetric features*

14.19 As-built survey points

Inframodel enables transferring both planned control points with tolerances (**<Cgpoints>** as top-level collection), and the measured values (**<Survey>.<Cgpoints>**), both having detailed properties assign to the collection as “IM_cgpoints”:

An *IM_cgpoints* <Feature> shall have the fields defined in Table 14.170:

Table 14.170: IM_cgpoints <Feature> fields.

Field	Type	Use	Description
Zero or one <i>toleranceXY</i> <Property> element	<i>toleranceXY</i> <Property>	Optional	Tolerance in horizontal plane from set northing easting in distance units (non-negative number)
Zero or one <i>toleranceAmin</i> <Property> element	<i>toleranceAmin</i> <Property>	Optional	Horizontal minimum tolerance in direction dirA (e.g. along referenced alignment, behind = non-positive number)
Zero or one <i>toleranceAmax</i> <Property> element	<i>toleranceAmax</i> <Property>	Optional	Horizontal maximum tolerance in direction dirA (e.g. along referenced alignment, behind = non-positive number)
Zero or one <i>toleranceBmin</i> <Property> element	<i>toleranceBmin</i> <Property>	Optional	Horizontal minimum tolerance in direction perpendicular to dirA (e.g. towards/from referenced alignment, left = non-positive number)
Zero or one <i>toleranceBmax</i> <Property> element	<i>toleranceBmax</i> <Property>	Optional	Horizontal maximum tolerance in direction perpendicular to dirA (e.g. towards/from referenced alignment, left = non-positive number)

Field	Type	Use	Description
Zero or one <i>toleranceZmin</i> <Property> element	<i>toleranceZmin</i> <Property>	Optional	Height minimum tolerance below set elevation (non-positive number)
Zero or one <i>toleranceZmax</i> <Property> element	<i>toleranceZmax</i> <Property>	Optional	Height maximum tolerance below set elevation (non-positive number)
Zero or one <i>alignmentRef</i> <Property> element	<i>alignmentRef</i> <Property>	Optional	Reference to <Alignment>.name
Zero or one <i>dirA</i> <Property> element	<i>dirA</i> <Property>	Optional	Direction of positive A tolerance, derived from direction of referenced alignment
Zero or one <i>geometryType</i> <Property> element	<i>geometryType</i> <Property>	Optional	Type of geometry that the control point is part of
code	<i>string</i>	Required	Fixed value, IM_cgpoints
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *toleranceAmin* <Property> shall have the fields defined in Table 14.171:

Table 14.171: *toleranceAmin* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>toleranceAmin</i>
value	<i>string</i>	Required	Horizontal minimum tolerance in direction <i>dirA</i> (e.g. along referenced alignment, behind = non-positive number)

A *toleranceAmax* <Property> shall have the fields defined in Table 14.172:

Table 14.172: *toleranceAmax* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>toleranceAmax</i>

Field	Type	Use	Description
value	<i>string</i>	Required	Horizontal maximum tolerance in direction dirA (e.g. along referenced alignment, behind = non-positive number)

A *toleranceBmin* <Property> shall have the fields defined in Table 14.173:

Table 14.173: *toleranceBmin* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>toleranceBmin</i>
value	<i>string</i>	Required	Horizontal minimum tolerance in direction perpendicular to dirA (e.g. towards/from referenced alignment, left = non-positive number)

A *toleranceBmax* <Property> shall have the fields defined in Table 14.174:

Table 14.174: *toleranceBmax* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>toleranceBmax</i>
value	<i>string</i>	Required	Horizontal maximum tolerance in direction perpendicular to dirA (e.g. towards/from referenced alignment, left = non-positive number)

A *toleranceZmin* <Property> shall have the fields defined in Table 14.175:

Table 14.175: *toleranceZmin* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>toleranceZmin</i>
value	<i>string</i>	Required	Height minimum tolerance below set elevation (non-positive number)

A *toleranceZmax* <Property> shall have the fields defined in Table 14.176:

Table 14.176: toleranceZmax <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, toleranceZmax
value	<i>string</i>	Required	Height maximum tolerance below set elevation (non-positive number)

An *alignmentRef* <Property> shall have the fields defined in Table 14.177:

Table 14.177: alignmentRef <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, alignmentRef
value	<i>string</i>	Required	Reference to <Alignment>.name

A *dirA* <Property> shall have the fields defined in Table 14.178:

Table 14.178: dirA <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, dirA
value	<i>string</i>	Required	Direction of positive A tolerance, derived from direction of referenced alignment

A *geometryType* <Property> shall have the fields defined in Table 14.179:

Table 14.179: geometryType <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, geometryType
value	<i>one of (pointGroup , polyline, closedPolyline)</i>	Required	Type of geometry that the control point is part of

A *toleranceXY* <Property> shall have the fields defined in Table 14.180:

Table 14.180: toleranceXY <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, strengthClass
value	string	Required	Pipe strength class

A *differenceXY* <Property> shall have the fields defined in Table 14.181:

Table 14.181: differenceXY <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, strengthClass
value	string	Required	Pipe strength class

A *dirDifferenceXY* <Property> shall have the fields defined in Table 14.182:

Table 14.182: dirDifferenceXY <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, strengthClass
value	string	Required	Pipe strength class

A *differenceA* <Property> shall have the fields defined in Table 14.183:

Table 14.183: differenceA <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, strengthClass
value	string	Required	Pipe strength class

A *differenceB* <Property> shall have the fields defined in Table 14.184:

Table 14.184: differenceB <Property> fields.

Field	Type	Use	Description
label	string	Required	Fixed value, strengthClass

Field	Type	Use	Description
value	<i>string</i>	Required	Pipe strength class

A *differenceZ* <Property> shall have the fields defined in Table 14.185:

Table 14.185: *differenceZ* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, strengthClass
value	<i>string</i>	Required	Pipe strength class

Instrument accuracy code may be set with “IM_survey” <Feature> extension

An *IM_survey* <Feature> shall have the fields defined in Table 14.186:

Table 14.186: *IM_survey* <Feature> fields.

Field	Type	Use	Description
One <i>accuracyType</i> <Property> element	<i>accuracyType</i> <Property>	Required	Survey method and its accuracy (from Finnish “Infra Rakentajakoodaus” code list)
One <i>accuracyDescription</i> <Property> element	<i>accuracyDescription</i> <Property>	Required	Survey method and its accuracy description, ie. “gnss guided excavator”
code	<i>string</i>	Required	Fixed value, IM_survey
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

An *accuracyType* <Property> shall have the fields defined in Table 14.187:

Table 14.187: *accuracyType* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, accuracyType

Field	Type	Use	Description
value	<i>string</i>	Required	Survey method and its accuracy (from Finnish “Infra Rakentajakoodaus” code list)

An *accuracyDescription* <Property> shall have the fields defined in Table 14.188:

Table 14.188: *accuracyDescription* <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, <i>accuracyDescription</i>
value	<i>string</i>	Required	Survey method and its accuracy description, ie. “gnss guided excavator”

Details:

section **12.1** *Control points*

section **12.2** *As-built survey*

14.20 Spatial allocation and spatial avoidance

An area or a space allocated for some specific use, or a perimeter around an object to be avoided can be defined using “IM_spatialZone” extension, applicable to <PlanFeature>, <Parcel>, <Pipe> or <Struct>.

An *IM_spatialZone* <Feature> shall have the fields defined in Table 14.189:

Table 14.189: *IM_spatialZone* <Feature> fields.

Field	Type	Use	Description
Zero or one <i>description</i> <Property> element	<i>description</i> <Property>	Optional	Spatial zone description
Zero or one <i>spatialAllocation</i> <Property> element	<i>spatialAllocation</i> <Property>	Optional	Spatial allocation
Zero or one <i>spatialAvoidance</i> <Property> element	<i>spatialAvoidance</i> <Property>	Optional	Spatial avoidance
code	<i>string</i>	Required	Fixed value, IM_spatialZone
source	<i>string</i>	Required	Fixed value, inframodel
name	<i>nameType</i>	Optional	Optional unique name

Where:

A *description* <Property> shall have the fields defined in Table 14.190:

Table 14.190: description <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, description
value	<i>string</i>	Required	Spatial zone description

A *spatialAllocation* <Property> shall have the fields defined in Table 14.191:

Table 14.191: spatialAllocation <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, spatialAllocation
value	<i>string</i>	Required	Spatial allocation in meters

A *spatialAvoidance* <Property> shall have the fields defined in Table 14.192:

Table 14.192: spatialAvoidance <Property> fields.

Field	Type	Use	Description
label	<i>string</i>	Required	Fixed value, spatialAvoidance
value	<i>string</i>	Required	Spatial avoidance in meters

Details:

section **11** *Planimetric features*

section **9.4** *Surface structures*

section **10.4** *Structures*

section **10.5** *Pipes*

15 Annex 1. Inframodel käyttöohje (Q&A)

15.1 Johdanto

Inframodel (IM) on suomalaisten kehittämä, kansainväliseen LandXML-standardiin (LandXML versio 1.2) perustuva avoin formaatti infratietojen siirtoon. Inframodel-määrittelytyötä on tehty yhdessä alan toimijoiden kanssa buildingSMART Finlandin (bSF) Infran standardointiryhmässä.

Inframodel tiedonsiirtoa hyödynnetään mm. suunnitteluohjelmissa sekä mittaus-, koneohjaus- ja tarkastussovelluksissa. Yhtenäinen toimintatapa ja avoin tietomallipohjainen formaatti on tehostanut suunnittelua ja rakentamista.

Inframodel skeema ja dokumentaatio sisältävät tietoa sovelluskehittäjille ja käyttöohje tukee loppukäyttäjiä.

Inframodel-tiedonsiirtoformaatti tukeutuu Yleisiin inframallivaatimuksiin ja InfraBIM-nimikkeistöjärjestelmään.

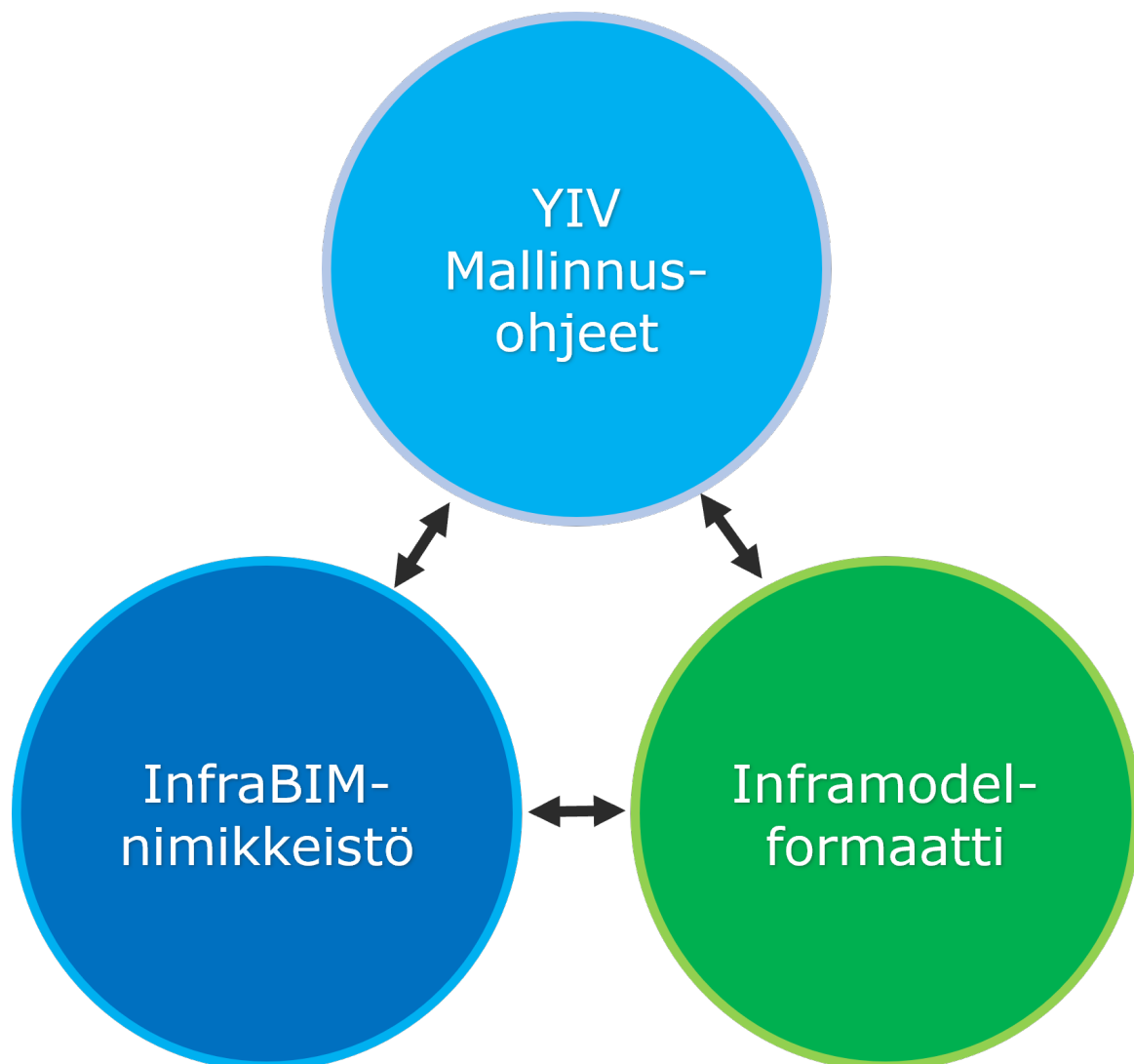


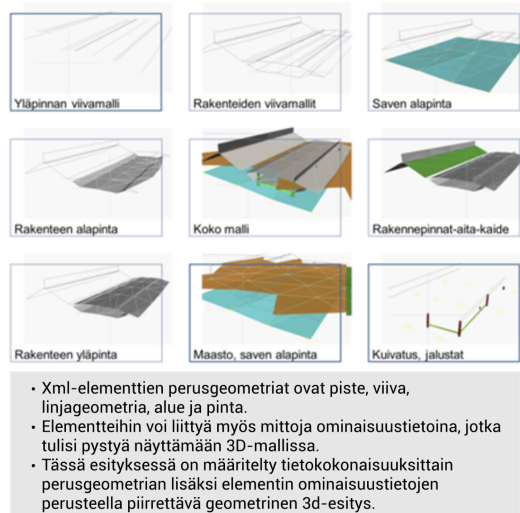
Figure 15.1: Kolminaisuus

- Yleiset inframallivaatimukset (*ohje ja vaatimus: mitä ja miten mallinnetaan, laadunvarmistus, lopputuotteen määrittely, mallinnusprosessi jne.*)

- InfraBIM nimikkeistö (*miten nimetään ja luokitellaan*)
- Inframodel tiedonsiirto (*miten tieto siirretään*)

Inframodel tiedonsiirron vähimmäisvaatimukset on kuvattu hankevaiheittain Yleisten inframallivaatimusten liitteessä: Luovutusaineiston tiedonsiirron vaatimukset.

Inframodel 3



Inframodel 4



Figure 15.2: Aiempien Inframodel versioiden tiedonsiirtomahdollisuudet, joita versio 4.2.0 täydentää

Inframodel 4.2.0 tiedonsiirron uusia ominaisuuksia ovat

- Vapaamuotoiset ominaisuustiedot – Mahdollistavat entistä laajemman kohdekohtaisen ominaisuustiedon siirtämisen standardin avulla
- Metadatan laajempi hyödyntäminen – esim. Tiedostokohtaisesti voidaan osoittaa kuka on tiedon omistaja
- Määrätiedot - Pintojen ja aluerajojen mukana voidaan kuljettaa tietoa pinta-alasta ja tilavuustiedosta
- Kaapelitiedon välittäminen selkeämpää – Kaapelitiedot siirtyvät skeemassa pipenetwork elementin alle
- Putkille ja kaivoille lisätty ominaisuustietoja – Kaivon pohjanpaksuus - Putken lujuusluokka
- Liikennemerkkitietojen siirtäminen – Mahdollista siirtää liikennemerkkien, pylväiden ja jalustan tietoja
- Kaiteet ja aidat – Pylväsväli lisätty ominaisuustietoihin
- Name attribuutit uniikeiksi – Määritellään name-attribuutit tiedostokohtaisesti uniikeiksi
- Turvarajat ja varoalueet - Muutos mahdollistaa tilavaraus- ja turvaetäisyystiedon siirtämisen
- Aluemaiselle tiedolle ominaisuudeksi pinta-ala – Mahdollistaa pinta-ala tiedon myös aluerajauksille

Silta- ja muiden taitorakenteiden tietomallipohjaiseen tiedonsiirtoon käytetään IFC-standardia, jota kehittää buildingSMART organisaatio.

Pohjatutkimusten siirtoon käytetään SGY:n (Suomen Geoteknillinen Yhdistys) Infra-pohjatutkimusformaattia.

Inframodel on eräänlainen tietokantamuoto, joka sisältää geometriaa, objektikuvauksia sekä metatietoja. Metatiedoille saattaa olla rajoitettuja arvojoukkoja, jotka ovat sallittuja ko. objekteille.

Inframodel- formaatti on kehittyvä ja sen tietosisältöä laajennetaan tulevissa versioissa. Inframodel on suomalainen lokalisatio, miten infran tietoa siirretään koneluettavassa muodossa. Tulevaisuuden tavoite on, että Inframodel aineistoa voidaan kirjoittaa ja lukea LandXML sekä IFC muodossa.

15.2 Ohjeen sisältö

Tämä ohje täydentää kohdassa 1 mainittua dokumentaatiota ja ohje on kohdennettu loppukäyttäjille.

Ohjeen jäsentely noudattaa Inframodel dokumentaation sisällysluetteloa. Ohjeen ulkoasu ja sisältö koostuu ohjeista sekä Q&A artikkeleista.

Ohjeistus on jatkuvasti päivittyvä ja täydentyvä kysymyksiin ja vastauksiin perustuva jatkuvasti kehittyvä dokumentti.

Ohjeistukseen voi itse vaikuttaa antamalla palautetta tai kysymällä buildingSMART Finlandin palautekanavien kautta tai tekemällä suoraan tiketin kehitysympäristöön. Saatu palaute tai tarvittavat lisäykset / tarkennukset täydennetään käyttöohjeeseen ylläpitoprosessin mukaisesti.

15.3 Käyttöohje + Q&A

15.3.1 Yleistä

Tiedonsiirrossa tulee noudattaa seuraavia yleisiä periaatteita:

- Tiedostojen ja suunnitelman eri osien nimeäminen noudattaa hankkeen yleistä nimeämiskäytäntöä
- Tiedostojen ja suunnitelman osien nimet ovat kuvaavia, mutta lyhyitä ja ytimekkäitä
- Kaikki saman hankkeen tiedot ovat samassa koordinaatti- ja korkeusjärjestelmässä
- Siirrettävistä tiedoista laaditaan hankkeelle aineistoluettelo ja aineistoselostus YIV ohjeiden mukaisesti

15.3.1.1 Aineiston jakaminen osiin

Hankkeissa tulee noudattaa selkeää jaottelua loogisiin osakokonaisuuksiin tai sopiviin tietomääriin. Hankkeissa on suositeltua noudattaa esim. seuraavanlaista jakoa:

- Maastomalli
- Maaperämallin pinnat
- Väyläkohtainen: geometriat, rakennekerrokset
- Vesihuolto / kuivatus kokonaisuutena tai verkostolajeittain
- Varusteet ja laitteet kokonaisuutena tai tekniikkalajeittain

Näiden lisäksi voi tulla tarpeeseen jakaa aineistoa myös paaluväleittäin selkeyttämään suunnittelurajoja tai mallinnusositusta. Vaikka tietomalli sisältää tiedon aineiston käyttötarkoituksesta, nimeämisessä olisi hyvä muistaa käyttää loogisia jaotteluita. Aineiston jakamisesta sopiviin osiin on hyvä sopia ennen hankkeen alkua, koska hankevaihe tai -muoto vaikuttaa myös aineiston jakamisen tarpeisiin. Esim. varhaisen suunnitteluvaiheen aineistoa voi olla helpompi jakaa isompina kokonaisuuksina, jopa yhtenä projektitietona.



Figure 15.3: Tiedostolista

Käytännössä tiedonsiirron tarpeet ovat monenlaisia ja kokonaisuudet on sovittava hankkeen alussa selkeästi. Esimerkiksi kun suunnittelija toimittaa malleja koneohjaukseen, voidaan toimia väylärakenteiden toteutusmalliohjeen mukaisesti (Väylärakenteen toteutusmallin vaatimukset ja -ohjeet) Tai kun maastomallin eri mittausaineistot (likimalli, tarkka maastomalli, täydennysmittaukset) halutaan selkeästi erottaa omina tiedostoina toisistaan.

Aineiston luomisessa, dokumentoinnissa, laadunvarmistuksessa, objekti- ja tiedostonimeämisessä sekä kansioinnissa noudatetaan yleisiä inframallivaatimuksia.

15.3.1.2 Q&A

15.3.2 Headers

Linkki dokumentaatioon section 1 *File headers*

15.3.2.1 Yleistiedot

Inframodel tukee LandXML-versiota 1.2 Inframodel- ja LandXML-versio näkyy aina tiedoston alussa. **Aineiston toimittaja vastaa siitä, että tiedosto on näiden suhteen validi, eli täyttää sisällön ja ominaisuustietojen (attribuuttien) rajausten ja eheyden osalta ko. versioiden vaatimukset.**

Mittayksikkösuositukset:

section **1.3 Units**

- sijainnin ja etäisyyden mittayksikkönä käytetään metriä (m)
- pinta-alalle neliometri (m2)
- tilavuuden yksikkö kuutiometri (m3)

Yleistietona on annettava seuraavat:

- Projektin
 - lyhyt nimi tai tunnus
 - pitkä koko nimi
- Koordinaatti- ja korkeusjärjestelmän nimi yksiselitteisesti
 - Suosituksena globaali EPSG- koodi
- Käytetyt lajiluokitusjärjestelmät
 - lähtötietoaineisto, eli maaastomallin ja maaperämallin kohteet ja pinnat
 - suunniteltujen kohteiden luokitus
 - lisäksi muut mahdolliset hankkeessa sovitut luokitukset
- Lähdeohjelmistosta ohjelman nimi ja versio
- Toimittajatiedoista
 - yritys
 - toimittajan nimi
 - sähköpostiosoite
 - tulostuksen aikaleima

Tiedoston ja objektin nimi ei saa sisältää erikois- tai skandinaavisia merkkejä.

15.3.2.2 Codings

Linkki dokumentaatioon section **1.6 Type coding systems** ja section **14.2 Type coding systems**

15.3.2.2.1 Lajiluokitukset ja nimikkeistöt

Inframodel tiedonsiirrossa aineiston laatimisessa käytetty luokittelu ja nimikkeistö esitetään Codings elementillä.

Yleisesti koodaukseen on sovittu seuraavat arvot ja käyttötarkoitukset:

Lähtötietoaineisto: maastomallin ja maaperämallin kohteet

- **<terrainCoding>** = Infra

Lähtötietoaineisto: maastomallin ja maaperämallin pinnat

- **<surfaceCoding>** = Infra

Suunniteltujen kohteiden (pinnat, alueet, pisteet, verkostot jne.) luokitus

- **<infraCoding>** = InfraBIM**

* *Tie ja ratahankkeiden maastotiedot, mittausohje (liikennevirasto 18/2017), koodiluettelo*

** *InfraBIM-nimikkeistö*

Lisäksi on mahdollista sisällyttää yksi tai useampi muu tarkentava koodaus. Esimerkiksi ohjelmiston oma lajiluokitus, organisaation oma luokitus, vesilaitoksen luokitus jne.

- **<proprietaryCoding>** = käyttäjän asettama arvo***

- Lähdejärjestelmän lajiluokitus (vapaaehtoinen)

- * esim. **<Proprietary label = "proprietaryInfraCoding" value="Tekla">**

*** *Suosittelaaan käytettäväksi tarvittaessa lisäinformaationa esimerkiksi silloin, kun em. luokitukset eivät ole riittäviä kyseisessä kohteessa. Tämän soveltamisesta pitää sopia erikseen hankekohtaisesti.*

15.3.2.3 Q&A

15.3.3 Base data

Linkki dokumentaatioon section **2 Base data**

15.3.3.1 Lähtötietoaineisto ja maaperämallit

Maastomalli/kartoitusaineistot jaetaan eri ryhmiin tai tiedostoihin (Surface) sekä alkuperän mukaisesti, esim. ilmakuvilta mitattu, laserkeilattu, maastossa mitattu. Lähtötietoaineiston tarkkuusvaatimuksia voivat olla esim. pintamallin yhtenäinen kolmiointi ja pisteiden tai viivojen noudattaminen maastomittausohjetta.

Kalliopinta-aineisto jaetaan avokalliopisteisiin ja tulkittuihin pisteisiin. Erottelu tehdään tulkintapisteiden luokittelulla. Lisäksi tulkintapisteet voidaan projektikohtaisesti sopia jaoteltavan tarkemmin: varmistettu porakoneella tai koekuopasta, tulkinta kairauksen kohdalla, tulkinta maatutkaluotauksesta, muu tulkinta.

Maakerrosrajapinnat tms. maaperämallin pinnat tulostetaan joko kaikki samaan tiedostoon omina pintoina tai erikseen kukin pinta omaan tiedostoon.

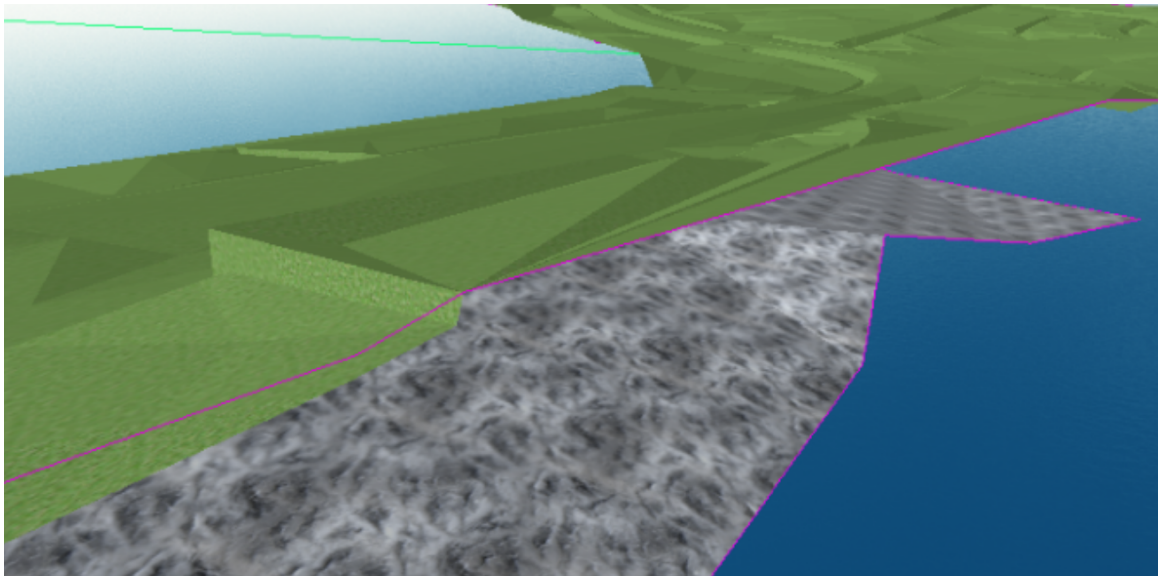


Figure 15.4: Maalajipinnat esimerkki 1

Lähtitietoaineistot sisältävät ko. pintojen ns. lähtiedot eli pisteet ja viivat. Näistä muodostettuja kolmiopintoja voidaan siirtää omina tiedostoinaan, jos tiedoston koko muuten kasvaa liian isoksi.

Maaperämallin pintojen lajikoodausta on tarkennettu InfraBIM-nimikkeistössä. Suosituksen mukaisesti maaperämallin tulkinta pitää luokitella tulkintatarkkuuden mukaan ja luokituksesta on tehty valmiit arvotjoukot eli luettelo sallituista tietoarvoista.

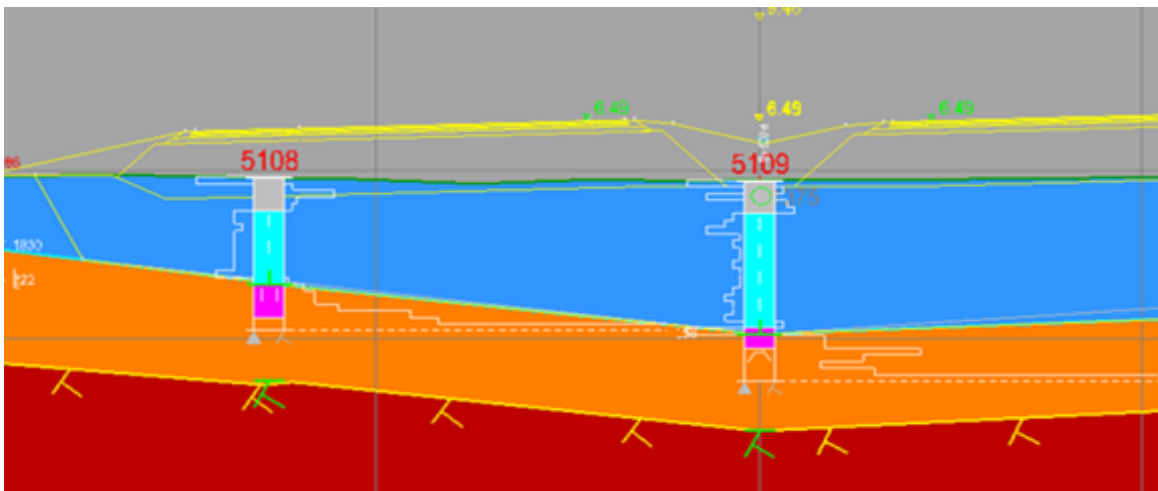


Figure 15.5: Maalajipinnat esimerkki 2

15.3.3.2 Q&A

15.3.4 Route planning

[Linkki dokumentaatioon section 3 Route planning](#)

15.3.4.1 Väylätiedot

Inframodel-tiedonsiirrossa väylällä on yksi jatkuva mittalinja ja tasaus. Tien mittalinjan InfraBIM-koodi tulee olla nimikkeistön mukaisesti (tie= 101 tai rata=111). Ratamallissa kilometripaalaus liitetään mittaraiteeseen. Muut geometrialinjat ja taiteviivat sekä pinnat nimetään InfraBIM-luokituksen mukaisesti. Taiteviivamallin viivoilla tulee olla tiedostokohtainen yksilöllinen nimi (@name).

Aineistojen nimeämisessä noudatetaan Yleisten inframallivaatimusten sekä InfraBIM-nimikkeistön mukaisia numerointi- ja nimeämiskäytäntöjä. Aineistossa siirretään mallin objekteja sekä niihin liitettyä tietoa. Objektit voivat olla esim. pisteitä, viivoja, geometrioita sekä kolmiointeja. Kolmioinnin vaatimuksena on käyttää kolmioverkkoa, jonka *sourcedata* osiosta löytyy myös BreakLine-tieto (kolmioinnin laskentaa ohjaavat taiteviivat).

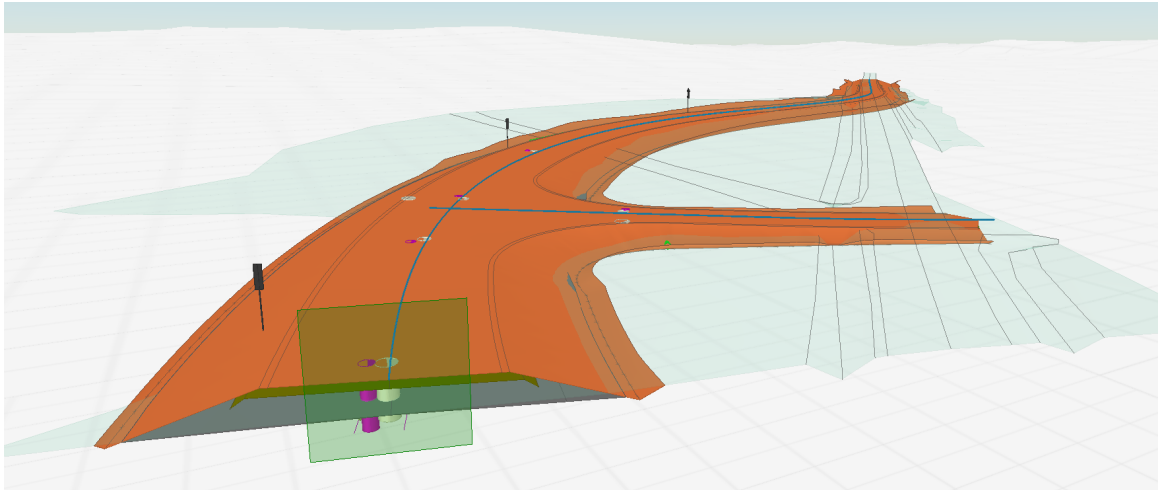


Figure 15.6: Esimerkki yhdistelmämalli

Aineistossa siirretään mallin objekteja sekä niihin liitettyä tietoa. Objektit voivat olla esim. pisteitä, viivoja, alueita, geometrioita sekä kolmiointeja.

15.3.4.2 Q&A

15.3.5 Roads and street

Linkki dokumentaatioon section **4** *Road and street design*

15.3.5.1 Q&A

15.3.6 Railways

Linkki dokumentaatioon section **5** *Railway design*

15.3.6.1 Q&A

15.3.7 Waterways

Linkki dokumentaatioon section **7** *Waterway design*

15.3.7.1 Q&A

15.3.8 Areas

Linkki dokumentaatioon section **9** *Area structures*

15.3.8.1 Q&A

15.3.9 Water supply and sewerage

Linkki dokumentaatioon section **10** *Pipenetworks*

15.3.9.1 Kuivatus

Vesihuoltoverkostot siirretään koko hankkeen kattavana tiedostona. Suurissa hankkeissa tiedostot voidaan jakaa esim. suunnitteluosuuksiin tai verkostolajeittain.

Kaivoilla ja laitteilla tulee olla koko aineiston kattava yksilöllinen nimi (@name). Vaatimuksena on myös sisällyttää putken korkeustaso uloskirjoitukseen.

15.3.9.2 Verkostolajit: Kaukolämpö, kaukokylmä, kaasu, jäte yms.

Inframodel sisältää omat määritteet verkostolajeina mm. vesijohdolle, kaukolämmölle, kaasulle sekä sähkö ja tietoliikennekaapelisuoja-putkille. Verkkolajit sisällytetään metatietolajiin "IM_pipeNetworkType".

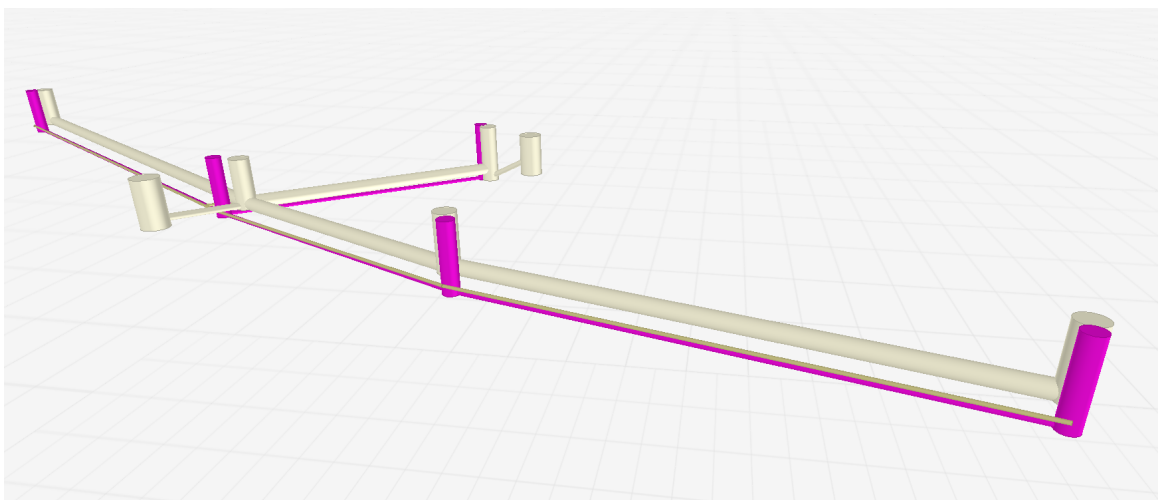


Figure 15.7: Esimerkki verkostoista

15.3.9.3 Q&A

15.3.10 Planimetric features

Linkki dokumentaatioon section **11** *Planimetric features*

15.3.10.1 Pintarakenteet ja niiden materiaaliominaisuudet

Linkki dokumentaatioon section **4.4** *Terrain model and structural model of road or street*

Pintarakenteiden materiaali esitetään aluerajauksina. Rajausta annetaan geometrialinjoina tai 2D- tai 3D-taiteviivaketjuina. Materiaalin päänimike tulee InfraBIM-luokituksen mukaisesti. Lisäksi voidaan antaa kerrospaksuus ja tarkempi kuvaus materiaalista tekstinä kuvauskentässä.

15.3.10.2 Rakennekerrosten materiaaliominaisuudet

Linkki dokumentaatioon subsection **11.1.1.5** *Surface structure*

Päällyste- ja pintarakennekerroksille luokitellaan materiaaliominaisuudet. Perusperiaatteena, että ominaisuus liitetään kerroksen yläpintaan. Tiedolla kuvataan kerroksen materiaali ja/tai materiaalin ominaisuuksia. Inframodel sisältää materiaaliominaisuuksista valmiit arvojoukkoluettelot.

15.3.10.3 Jalustojen ominaisuudet

Linkki dokumentaatioon subsection **11.1.1.2** *Footing*

Jalusta on pistemäinen PlanFeatures- kohde, jolla on InfraBIM-luokitus. Lisäksi jalustalle voidaan antaa lisätietoja kuten korkeus, materiaali, perustustapa, numero ja tilavaraus. Korkeus esitetään tietomallin *Units* asetusten mukaisesti. Jalustan tilavaraus määritetään joko säteenä tai suorakulmion nurkkapisteinä suhteessa jalustan keskipisteeseen.

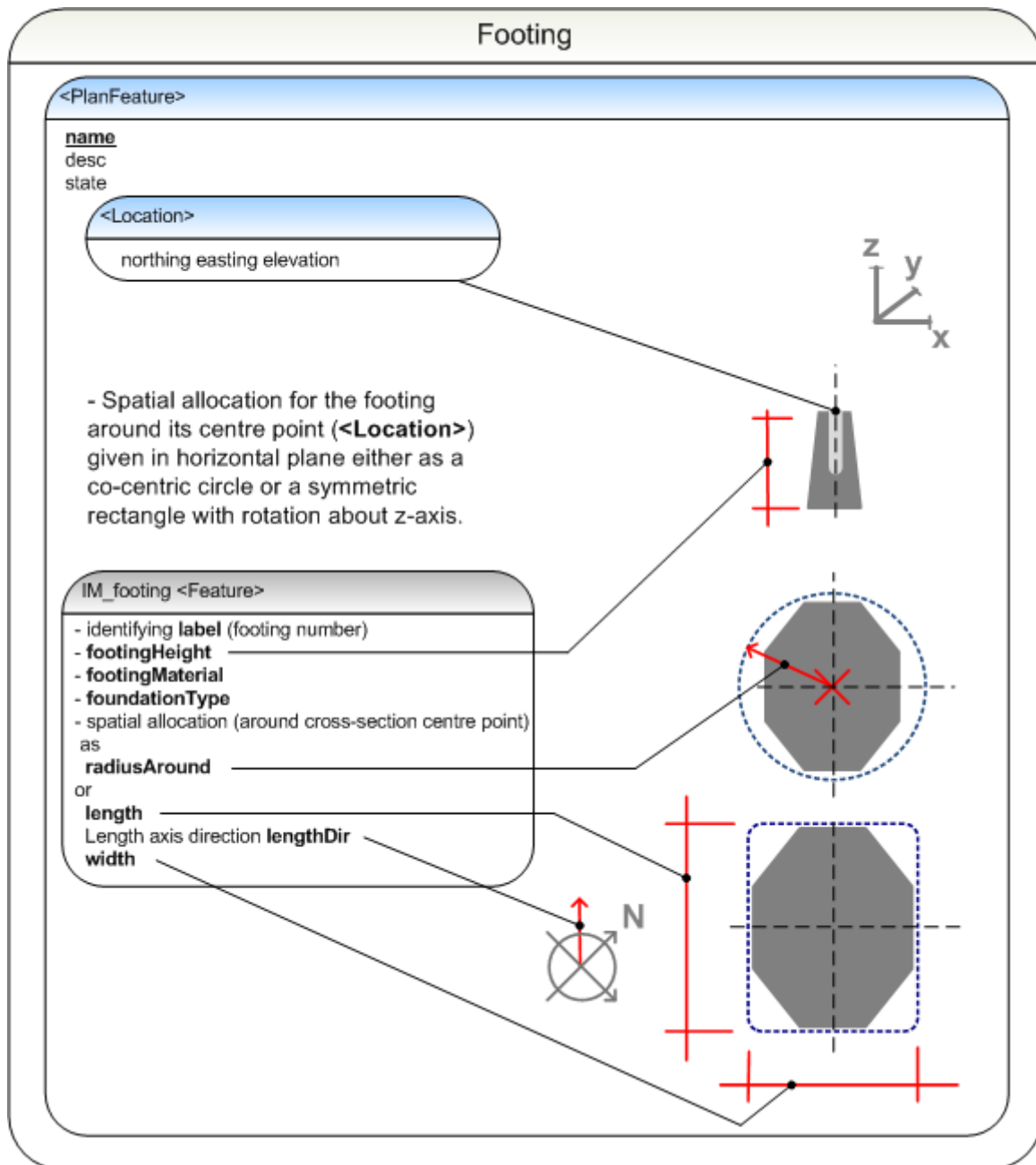


Figure 15.8: Esimerkkikuva jalustat

15.3.10.4 Johto- ja kaapelireitit

Linkki dokumentaatioon subsection **11.1.1.1 Cable**

Kaapelirakenteet on kuvattu Inframodelissa PlanFeature objekteina. Kaapeli on murtoviivaa, jolla on InfraBIM-luokitus. Kaapelin muita ominaisuuksia ovat omistaja, kaapelin tyyppi, tunnus ja tilavaraus. Tilavaraus kuvataan metatietona säteenä tai suorakulmion nurkkapisteinä suhteessa kaapelin keskipisteeseen.

Inframodel 4.2.0 versiossa johdot ja kaapelit voidaan kuvata myös putkiverkkona. section **10.5 Pipes**

15.3.10.5 Kaiteiden ja aitojen ominaisuudet

Linkki dokumentaatioon subsection **11.1.1.3 Railing** ja subsection **11.1.1.4 Fence**

Kaide ja aitarakenteet on kuvattu Inframodelissa PlanFeature-objekteina. Kohteet ovat murtoviivaa, jolla on InfraBIM-luokitus. Objektin asennuskohdan XYZ ilmaistaan ylimmän yhdistelmäpinnan (yyp) koordinaatteina. Lisäinformaatio voi sisältää esim. kaidetyyppi, käyttötarkoitus, törmäyskestävyys, joustovara, aerauskestävyys, toimintaleveys, korkeus, kaiteen aloitus ja lopetus

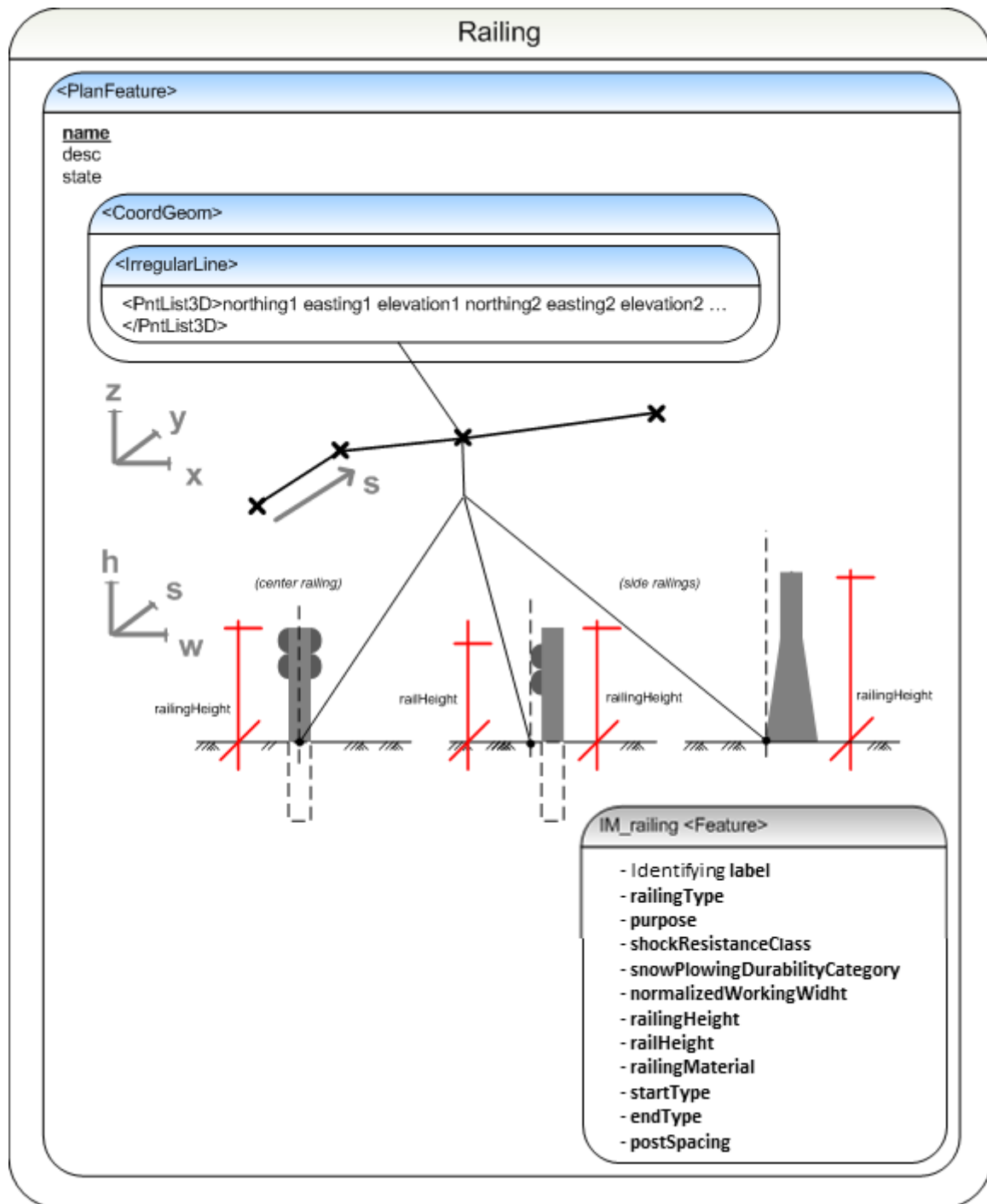


Figure 15.9: Esimerkkikuva kaiteet

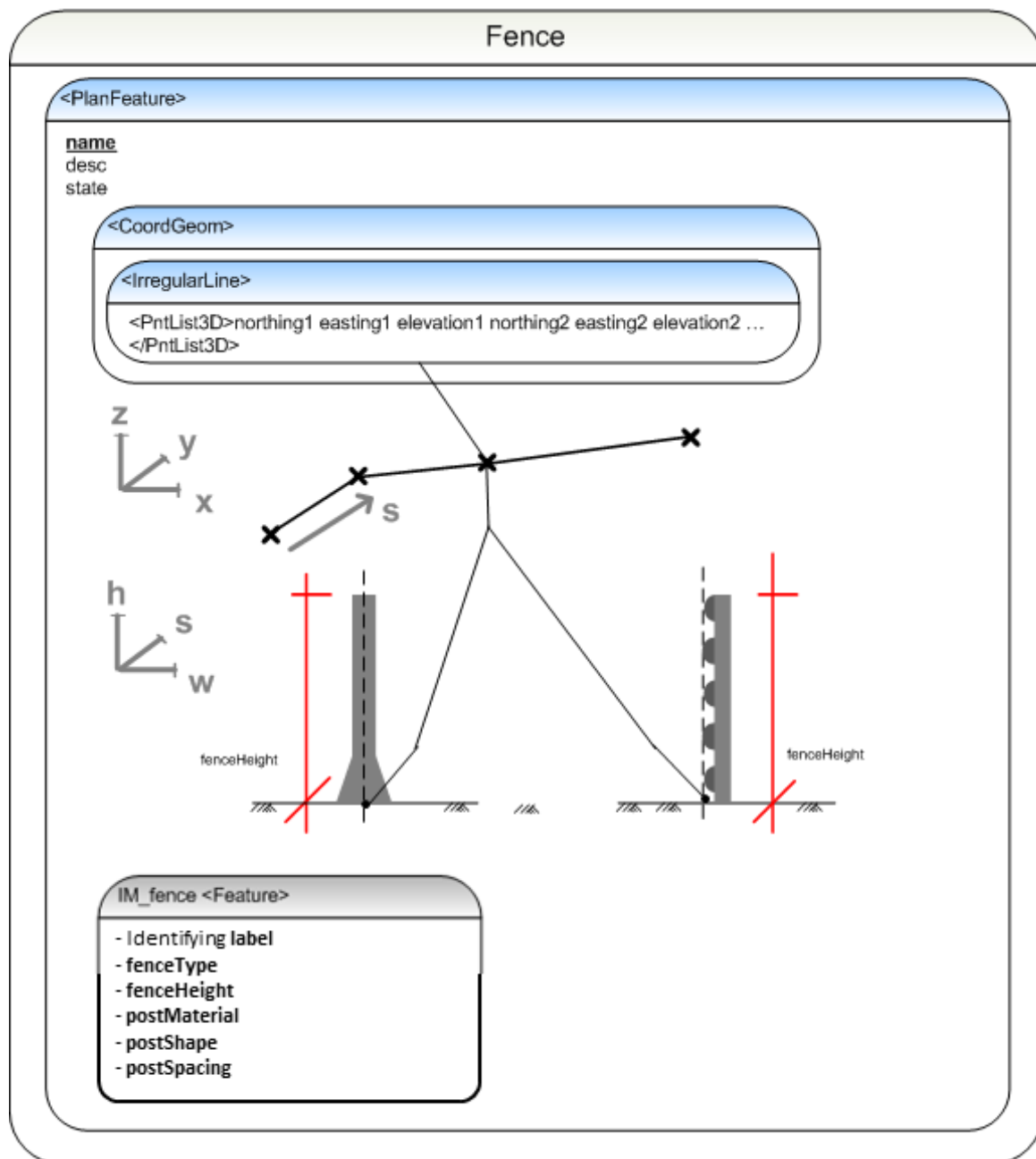


Figure 15.10: Esimerkkikuva aidat

15.3.10.6 Q&A

15.3.11 As-built

Linkki dokumentaatioon section **12 AsBuilt data**

15.3.11.1 Toteumatiedot

Toteutuneen tilanteen tallentamista kutsutaan toteumatiedoksi. Tietoihin sisällytetään selkeästi yksilöitävät pistetiedot, toleranssit, erovektorit sekä mittausmenetelmän tarkkuusluokka.

Toteumamallin käyttötarkoitus on rakenteen geometrisen laadun ja vaatimusten mukaisen toteutuksen todentaminen tilaajalle ja omaisuudenhallinnan lähtötietona toimiminen tilaajan ylläpitoprosessissa. Suosituksena on vähentää laadunvarmistukseen liittyvän mittaustiedon paperidokumentaation laatimiseen käytettävää työmäärää ja dokumentteja.

Väylärakenteen toteumamalli koostuu seuraavista INFRA 2015 rakennusosa- ja hankenimikkeistön mukaisista rakennusosista:

15.3.11.2 Q&A

15.3.12 Deep foundations

Linkki dokumentaatioon section **13** *Deep foundations*

15.3.12.1 Pilari- ja massastabilointi

Pilari- ja massastabilointi on pääelementti Inframodelissa. Pilariryhmä voidaan kuvata pintojen välisenä tilavarauksena (ulkoreuna ylä- ja alapinnassa) tai yksittäisen pilarin tarkkuudella. Molemmissa tavoissa ryhmään voi liittää useita ominaisuuksia, kuten esim. halkaisija, sideainetyyppi, sideainemäärä.

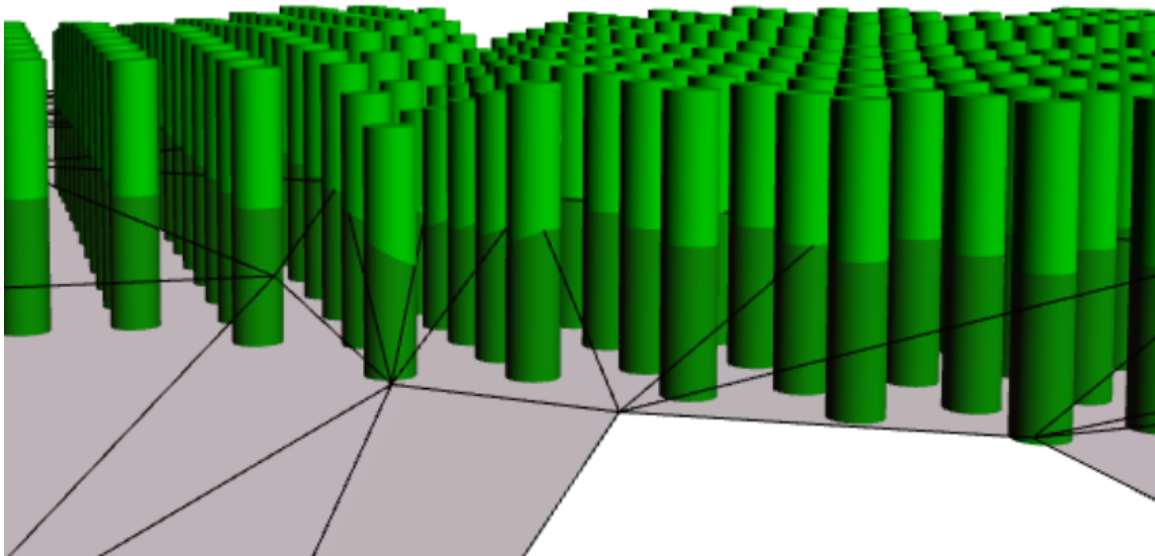


Figure 15.11: Esimerkki pilaristabilointi

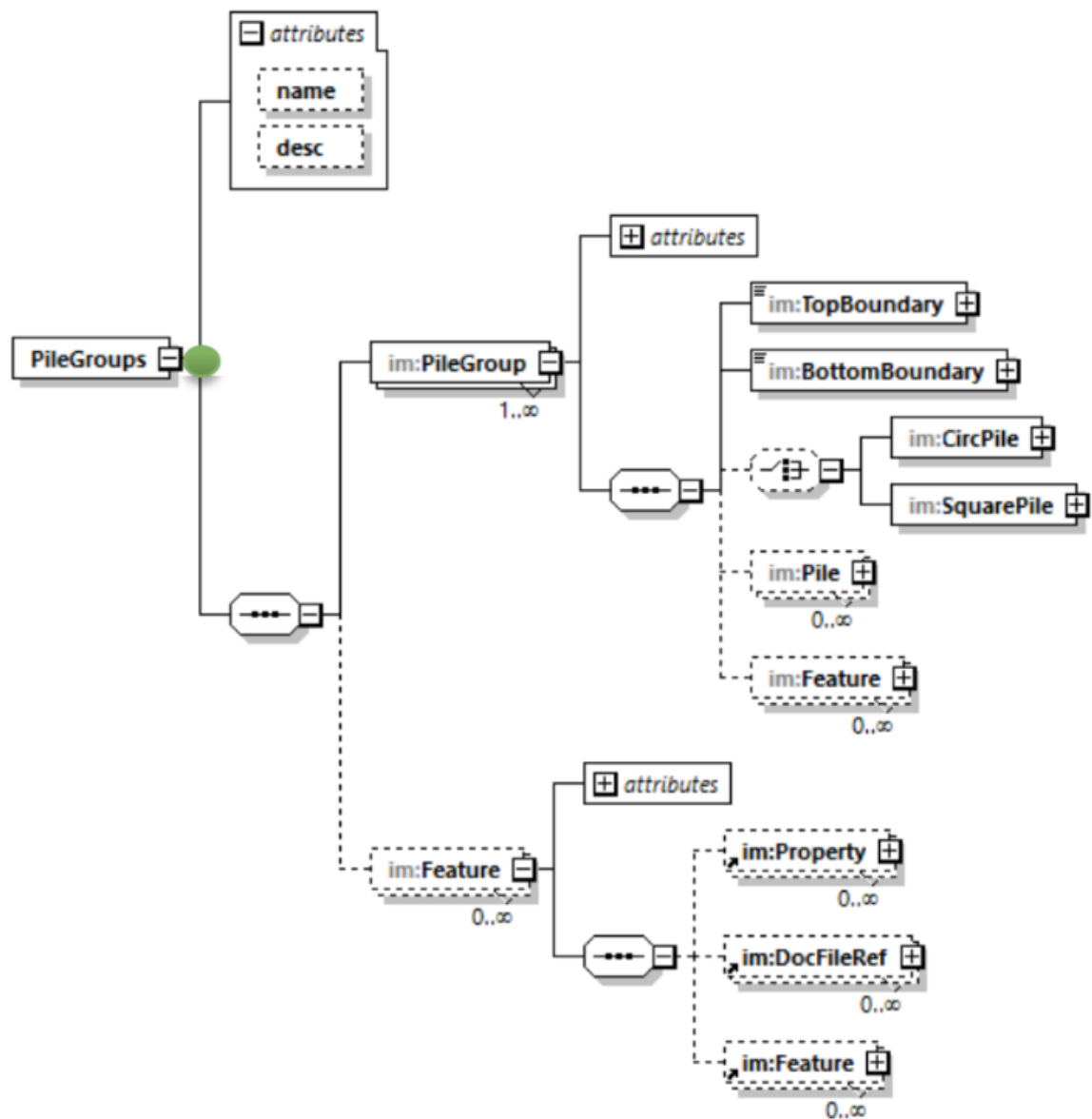


Figure 15.12: Esimerkki skeemakuvaus

15.3.12.2 Q&A

15.3.13 Extensions

Linkki dokumentaatioon section **14** *Inframodel <Feature> extensions*

15.3.13.1 Q&A