GLDF Update

Dipl.-Ing. (FH) Robert Heinze
Relux Informatik AG
Switzerland, 4142 Münchenstein, Kaspar-Pfeiffer-Str. 4

Summary

At the last LICHT 2021 RELUX and DIAL presented the new luminaire format GLDF. My LICHT 2021 paper described the fundamentals of GLDF. In the last 2 years there were progress on the format which this paper will reflect.

GLDF is an open and universal format to transport all data of a luminaire. The invention by RELUX and DIAL ensures the usage in lighting design and lighting specialised applications as well as in all BIM environments. The open-source licence (MIT) allows a wide and simple usage by all luminaire data creators and users. With GLDF the lighting industry gets a digital twin for luminaires.

1 State of the format

After three years of working intensively on the format from both lighting software houses, the version 1.0 of the GLDF format was published by the 1st of April 2022. It was published one year before with a XSD and the zip container description but under constant improvement and development. With version 1.0 the format is in its final state which was not changed since April and will not be touched for the next months.

One of the latest changes early 2022 was the allocation of the LDCs to the light output on the geometry and the early split in absolute (`FixedLightSource`) and relative (`ChangeableLightSource`) light sources.

Also, the new open geometry format L3D was published simultaneously to GLDF. L3D is an XML format in a zip container just like GLDF. The base 3D model format for L3D is OBJ, an open source, simple, and widely used mesh format.

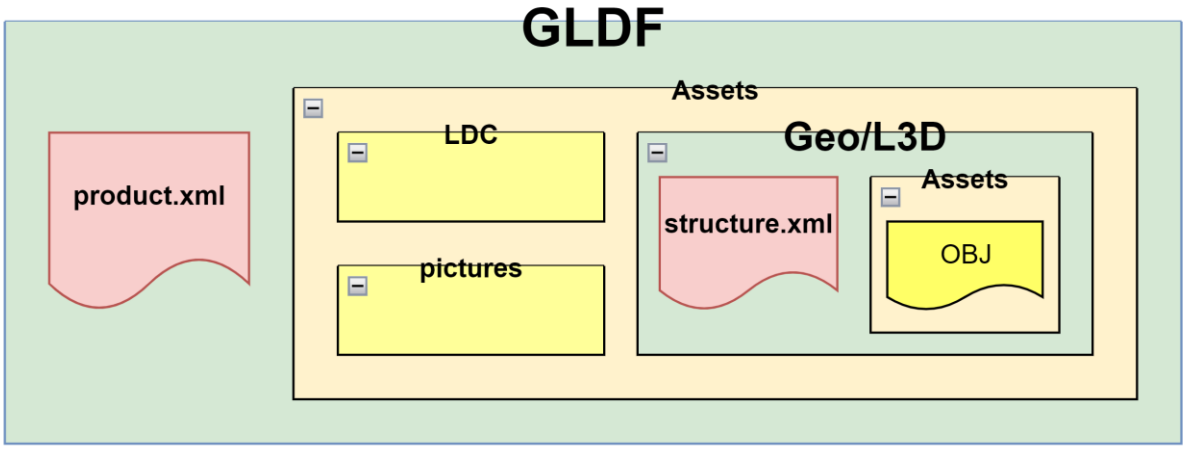

Fig.1: GLDF and L3D container structure

The version 1.0 of GLDF could be accessed and downloaded from *gldf.io*. At the moment the development of the format is paused by RELUX and DIAL. So, the implementation at luminaire manufactures PIM systems will not be disturbed by format changes. Depending on the implementation feedback we will continue the format improvement probably by 2024 and creating a new version.

2 Documentation

Everything related to GLDF could be accessed via *https://gldf.io*. Most important is the well annotated XSD which defined the XML structure for the product.xml file. This is the core of the format. In the product.xml the luminaire or sensor product is described in several variants. Together with the xml all needed assets files, like pictures, geometry, or pdf documents, are zipped in a GLDF container. It is also possible in GLDF to use URLs / links to the asset files instead of hard copied files. The container definition and the L3D geometry definitions could also find on *gldf.io*. There are also samples for GLDF and L3D files and a HTML XSD reference. Besides this technical information also links and references to presentations and videos are available.

DIAL and RELUX are also working on a PDF-like documentation on *gldf.io/docs*. Half of it is already available. The other half will be available during the next months. The parametric geometry P3D definition is also available in docs today.

3 Properties

Today there are 110 native properties in the GLDF to describe products. Additionally, the main product structure with geometry and variants in the xml hierarchy delivers more information than just plain properties.

With custom properties in the GLDF it is possible to add all kind of different properties of the product or the variant into the GLDF. This is quite efficient if the property source is official and public (e.g., an ISO standard). But it's also possible to transport own properties just for internal usage like a PIM ID number. Custom properties could also be an application-specific property just for a certain tool or environment.

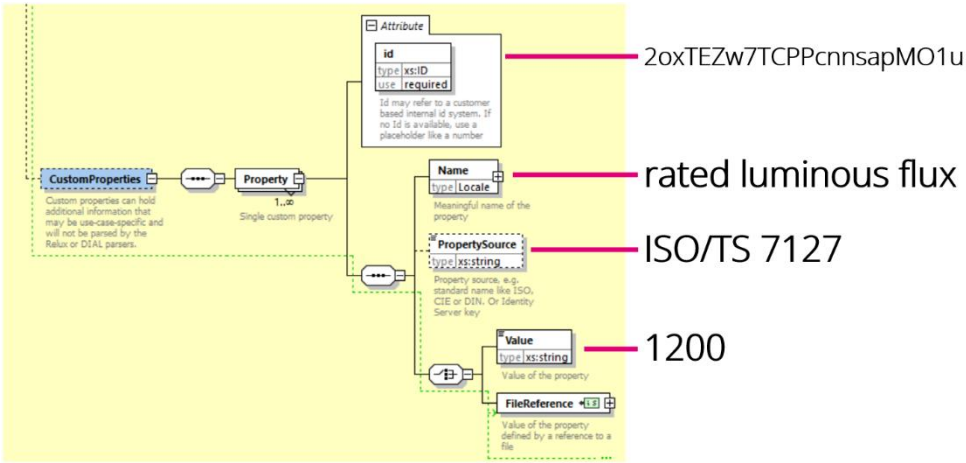

Fig.2: structure of custom properties in GLDF

In spring 2023 the new ISO/TS 7127 will be published. This ISO document is a property source for around 400 properties for lighting systems. The base is the European CEN/TS 17623 from 2021. The ISO technical specification incorporates more international properties and more properties in general. Both standards are compatible cause the GUIDs were not changed, and no property was deleted. All descriptions and references to other documents and standards were reviewed and updated.

4 Tooling

With GLDF Version 1.0 two new and important tools were published on gldf.io. The L3D-Editor allows to view, create, and edit L3D files. The GLDF-Editor does the same for GLDF files. Both editors are free web-based tools and allowing a local file exchange. Before these tools where available all GLDF files were needed to be created manually in XML editors.

The two editors allow a single file treatment with a graphical user interface. Many luminaire manufactures would probably create generators / scripts for the GLDF on their PIM systems. GLDF is perfect for a mass creation and publication automatization processes. Since the format is open source, everybody is allowed to create specialized or public tools.

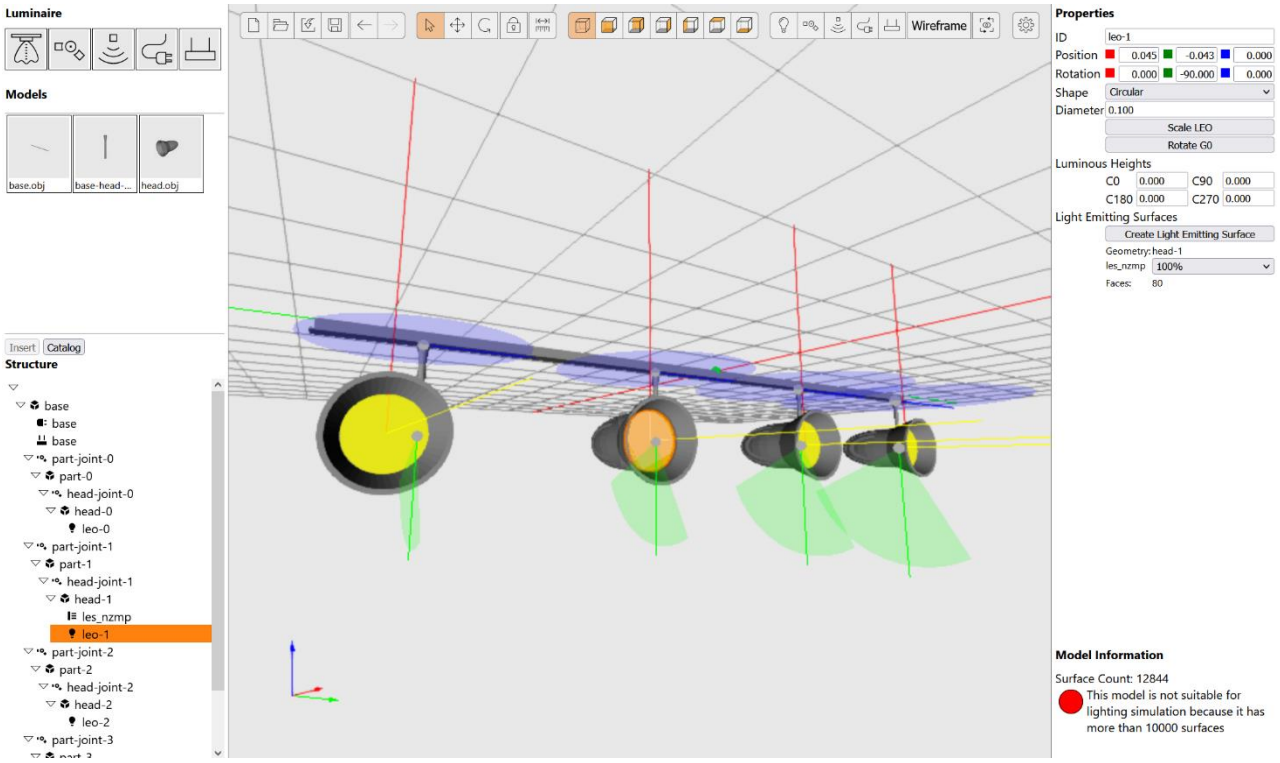

Fig.3: L3D-Editor – l3d-editor.gldf.io

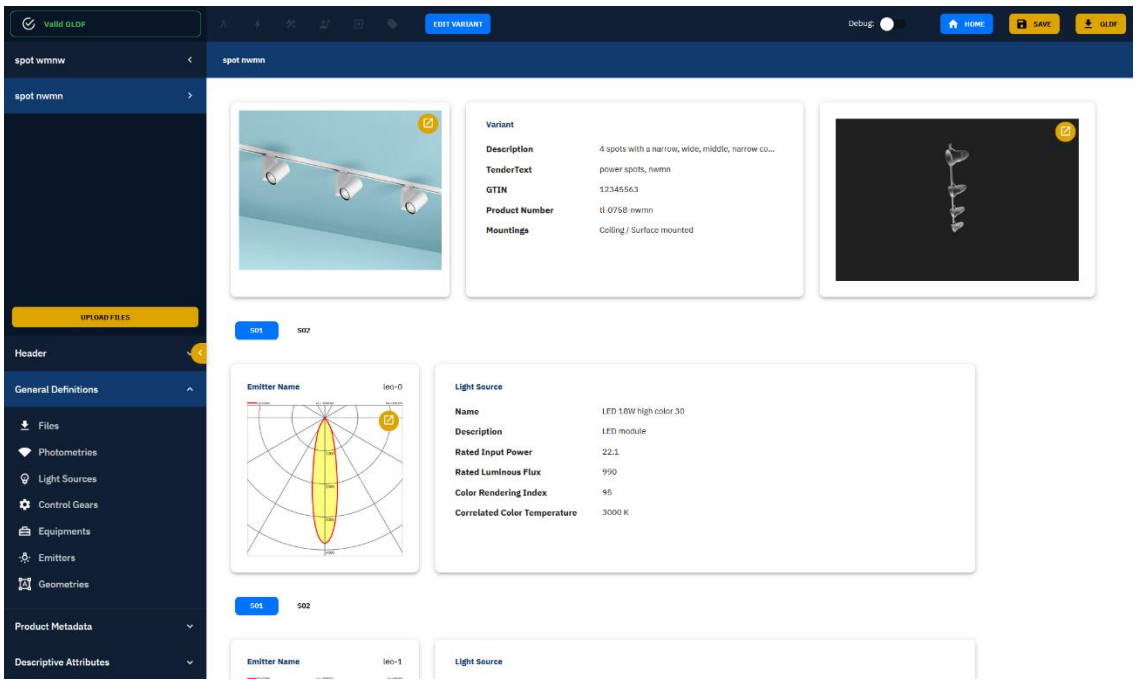

Fig.4: GLDF-Editor – gldf-editor.gldf.io

The GLDF checker is in a beta state now. An online free to use tool to check the filling level of an GLDF for everyone. The GLDF checker shows the amount of filled structure and properties in an easy and graphical way. A good option for quality checks on GLDF files. The specific applications and groups help to identify the potential usage.

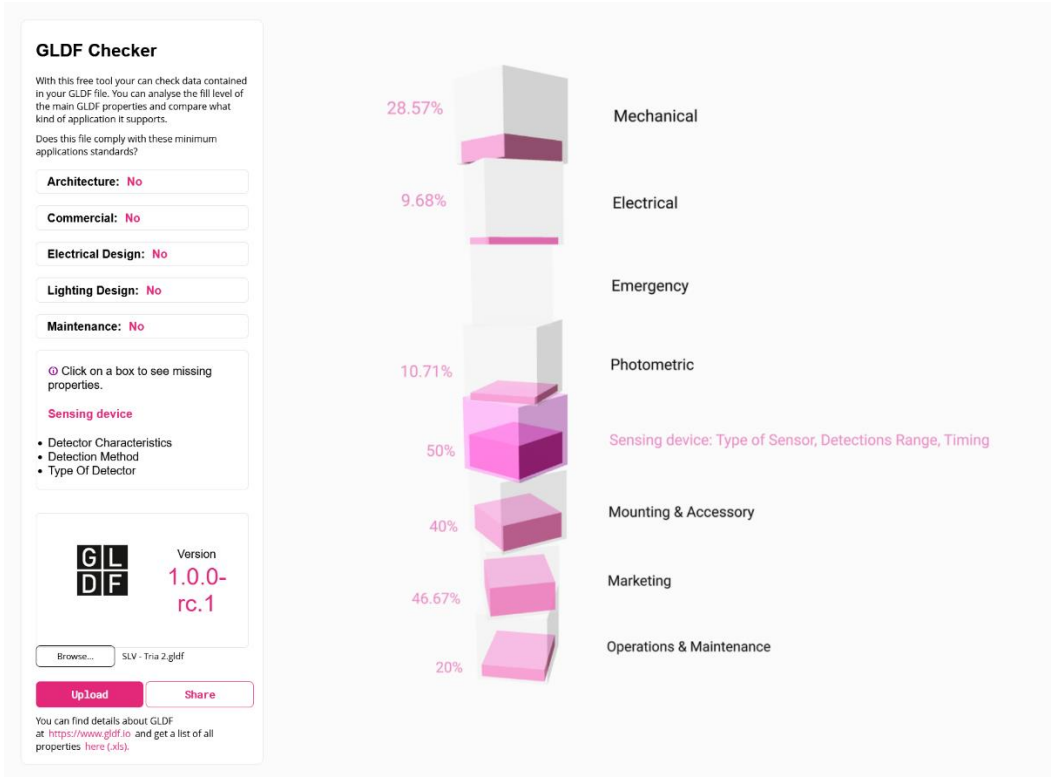

Fig.5: GLDF-Checker

5 Parametric Geometry

We are seeing progress on the parametric geometry. RELUX and DIAL created the xml based P3D format and defined 22 basic luminaire shapes.

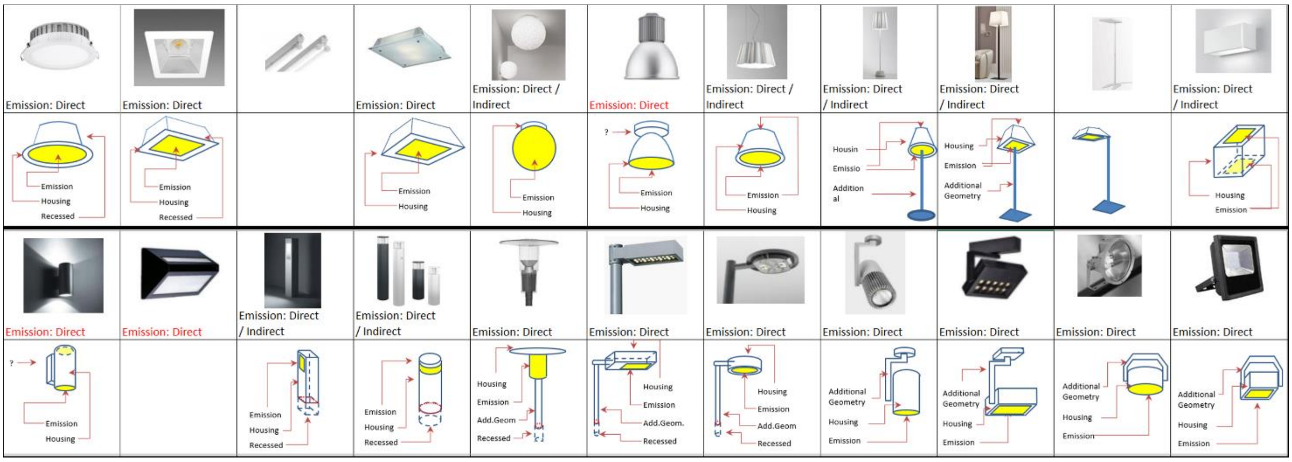

Fig.6: Parametric basic shapes

These basic shapes together with a minimum of dimensions are enough for a full geometry description or rather a L3D. For the most shapes a simple length, width, hight information is enough. These dimensions could be found in a Eulumdat file.

It is not possible to use the P3D directly in a GLDF as geometry. It’s necessary to convert P3D to L3D beforehand. The pure backend functionally for this conversion was completed in 2022. We are looking forward to providing this functionality this year.

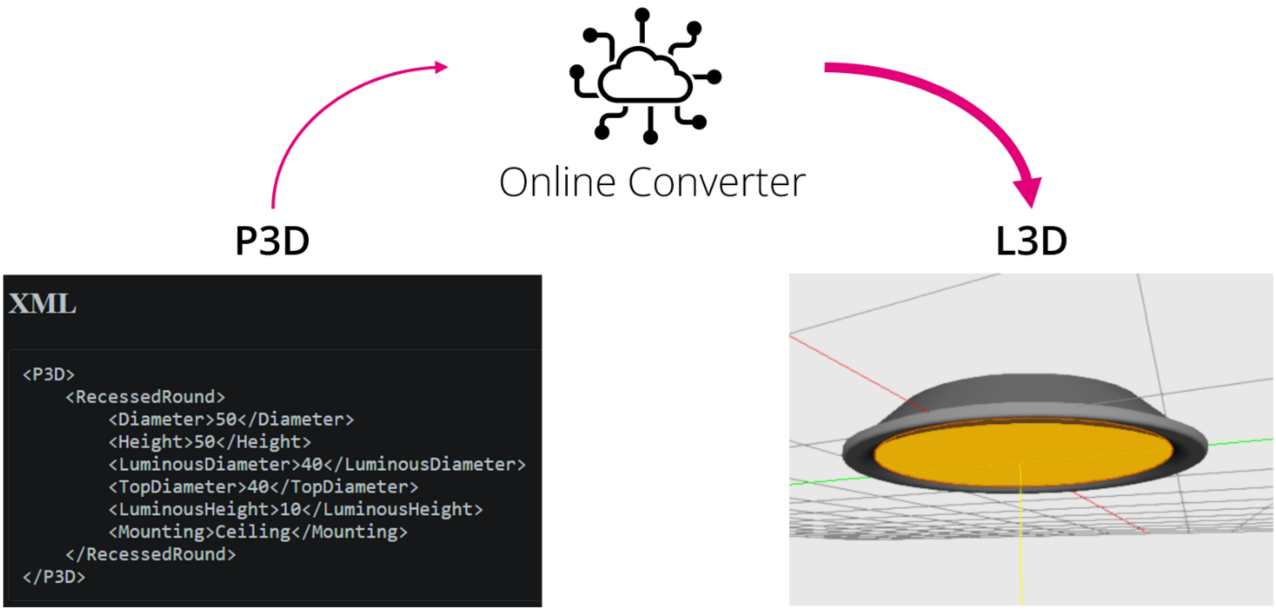

Fig.7: P3D example with conversion result

6 Applications

RELUX and DIAL are going to allow a direct import of GLDF files in ReluxDesktop and DIALux evo. Since November 2022 RELUX runs a closed Beta Test with an GLDF Drag and Drop Feature. The test group consists of luminaire manufactures which are preparing GLDF generation processes on their product portfolio. Soon RELUX will also test the mass data transport of GLDF files from manufactures via ReluxNet to ReluxDesktop and ReluxCAD for Revit including the automized RFA generation.

DIAL is planning to publish a GLDF import in DIALux evo in the middle of this year.

There is a discussion at Autodesk Revit about an implementation of GLDF.

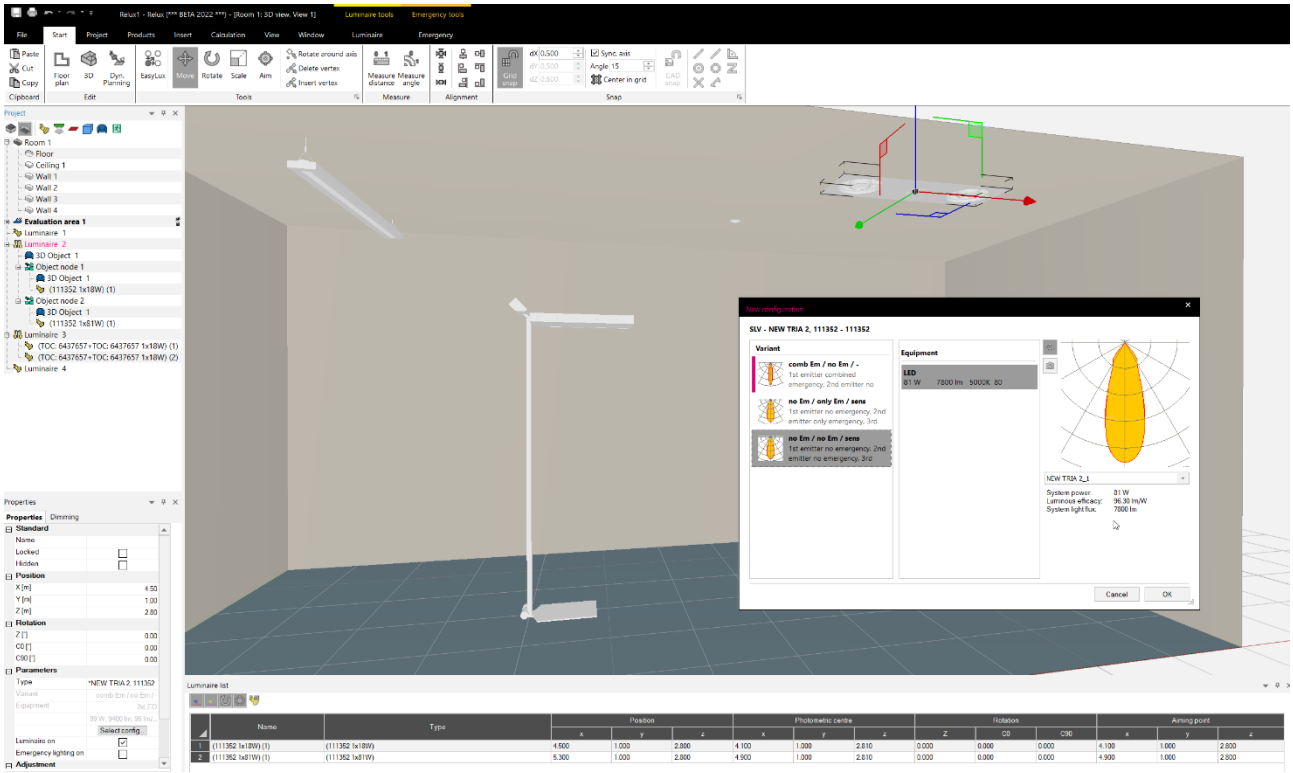

Fig.8: ReluxDesktop BETA with GLDF luminaires; open window for variant selection

7 Technical GLDF sample

With time more and more samples in different complexities from more and more creators are going to be circulate. A good starting point is a minimal content type GLDF just to test the core functionality. Here is a sample of a minimum GLDF with L3D geometry.

The geometry and even the light distribution are optional. But the real minimal GLDF according to mandatory elements is too boring. Of cause, the real-world samples are much

longer and more complex. Kind of standard is, like in ROLF, is a product picture and descriptions. We are curious which of the new options of GLDF will be used in practice.

```
<?xml version="1.0" encoding="UTF-8"?>
<Root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://gldf.io/xsd/gldf/1.0.0-rc.1/gldf.xsd">
    <Header>
        <Manufacturer>Philips Lighting</Manufacturer>
        <CreationTimeCode>2021-01-25T09:30:47Z</CreationTimeCode>
        <CreatedWithApplication>N/A</CreatedWithApplication>
        <FormatVersion>1.0.0-rc.1</FormatVersion>
    </Header>
    <GeneralDefinitions>
        <Files>
            <File id="ldtFile" contentType="ldc/eulumdat" type="localFileName">SP542P SRD L1480.ldt</File>
            <File id="geometryFile" contentType="geo/l3d" type="localFileName">SP542P_SRD_L1480.l3d</File>
        </Files>
        <Photometries>
            <Photometry id="photometry01">
                <PhotometryFileReference fileId="ldtFile"></PhotometryFileReference>
            </Photometry>
        </Photometries>
        <LightSources>
            <FixedLightSource id="lightSource01">
                <Name><Locale language="de">LED</Locale></Name>
                <RatedInputPower>47.5</RatedInputPower>
            </FixedLightSource>
        </LightSources>
        <Emitters>
            <Emitter id="emitter01">
                <FixedLightEmitter>
                    <PhotometryReference photometryId="photometry01"/>
                    <LightSourceReference fixedLightSourceId="lightSource01"/>
                    <RatedLuminousFlux>6600</RatedLuminousFlux>
                </FixedLightEmitter>
            </Emitter>
        </Emitters>
        <Geometries>
            <ModelGeometry id="geometry01">
                <GeometryFileReference fileId="geometryFile" levelOfDetail="Medium"/>
            </ModelGeometry>
        </Geometries>
    </GeneralDefinitions>
    <ProductDefinitions>
        <ProductMetaData>
            <ProductNumber>
                <Locale language="de">SP542P</Locale>
            </ProductNumber>
            <Name><Locale language="de">Pendelleuchten SP542P LED66S/940</Locale></Name>
        </ProductMetaData>
        <Variants>
            <Variant id="variant1">
                <Name>
                    <Locale language="en">Variant 1</Locale>
                </Name>
                <Mountings>
                    <Ceiling><SurfaceMounted/></Ceiling>
                </Mountings>
                <Geometry>
                    <ModelGeometryReference geometryId="geometry01">
                    <EmitterReference emitterId="emitter01">
                        <EmitterObjectExternalName>LEO</EmitterObjectExternalName>
                    </EmitterReference>
                </ModelGeometryReference>
                </Geometry>
            </Variant>
        </Variants>
    </ProductDefinitions>
</Root>
```

Fig.9: Miminal GLDF product.xml with one Eulumdat and one L3D Geometry

This product.xml zipped with the Eulumdat file and the L3D file looks in ReluxDesktop BETA like this:

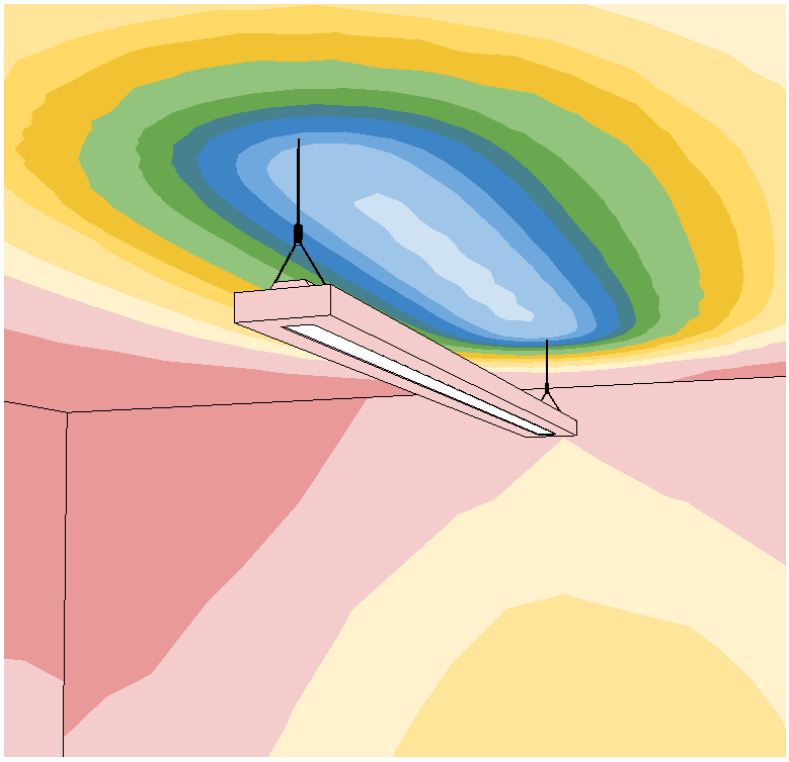

Fig.10: Miminal GLDF in ReluxDesktop BETA

And here the L3D structure.xml of the above sample for the completeness:

```
<Luminaire a0:noNamespaceSchemaLocation="https://gldf.io/xsd/l3d/0.9.2/l3d.xsd"
  xmlns:a0="http://www.w3.org/2001/XMLSchema-instance">
    <Header>
        <Name/>
        <CreatedWithApplication>Online L3D-Editor</CreatedWithApplication>
        <CreationTimeCode>2023-01-29T09:46:10.229Z</CreationTimeCode>
    </Header>
    <GeometryDefinitions>
        <GeometryFileDefinition id="geom_1" filename="luminaire.obj" units="m"/>
    </GeometryDefinitions>
    <Structure>
        <Geometry partName="luminaire">
            <Position x="0" y="0" z="0"/>
            <Rotation x="0" y="0" z="0"/>
            <GeometryReference geometryId="geom_1"/>
            <LightEmittingObjects>
                <LightEmittingObject partName="LEO">
                    <Position x="0" y="0" z="-0.21"/>
                    <Rotation x="0" y="0" z="0"/>
                    <Rectangle sizeX="0.052" sizeY="1.404"/>
                </LightEmittingObject>
            </LightEmittingObjects>
            <LightEmittingSurfaces>
                <LightEmittingSurface partName="les_tolo">
                    <LightEmittingObjectReference lightEmittingPartName="LEO" intensity="1"/>
                    <FaceAssignments>
                        <FaceRangeAssignment groupIndex="0" faceIndexBegin="1396" faceIndexEnd="1397"/>
                        <FaceRangeAssignment groupIndex="0" faceIndexBegin="1404" faceIndexEnd="1405"/>
                    </FaceAssignments>
                </LightEmittingSurface>
            </LightEmittingSurfaces>
        </Geometry>
    </Structure>
</Luminaire>
```

Fig.11: L3D structure.xml; the luminaire 3D model “luminaire.obj” is required in the L3D zip container