1 Format structure

F4D format is not file based but folder based that contains several datasets. F4D format is composed of mandatory parts(1 header file and 3 sub folders) and optional parts(1 image file and 1 sub folder). Former are HeaderAsimetric.hed, Bricks folder, Models folder and References folder, and later are SimpleBuildingTexture3x3.png and Images_Resized folder respectively. Parent folder name of datasets is usually used as an id.

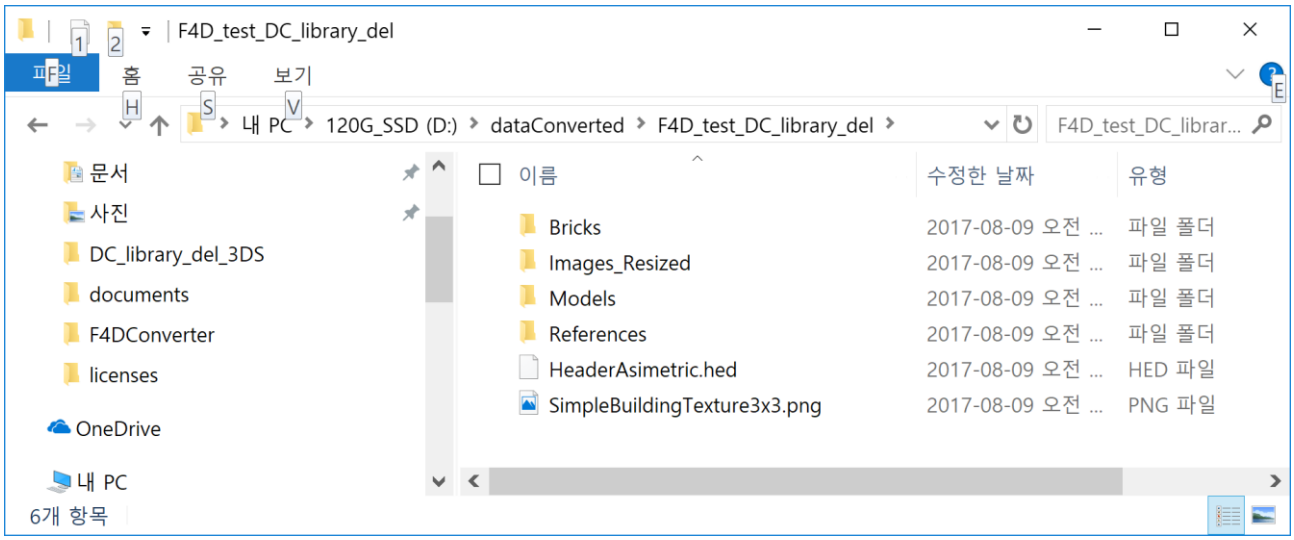

Figure 1. Overall Structure of F4D Format

objectIndexFile.ihe is necessary to publish F4D data through web services. This file is a temporary metadata, which is going to nothing. (The role of this file is being implemented in other parts mago3D and this file is being removed. not removed completely yet.) This file should be in F4D data folder for web service at the same depth as each F4D data folders.

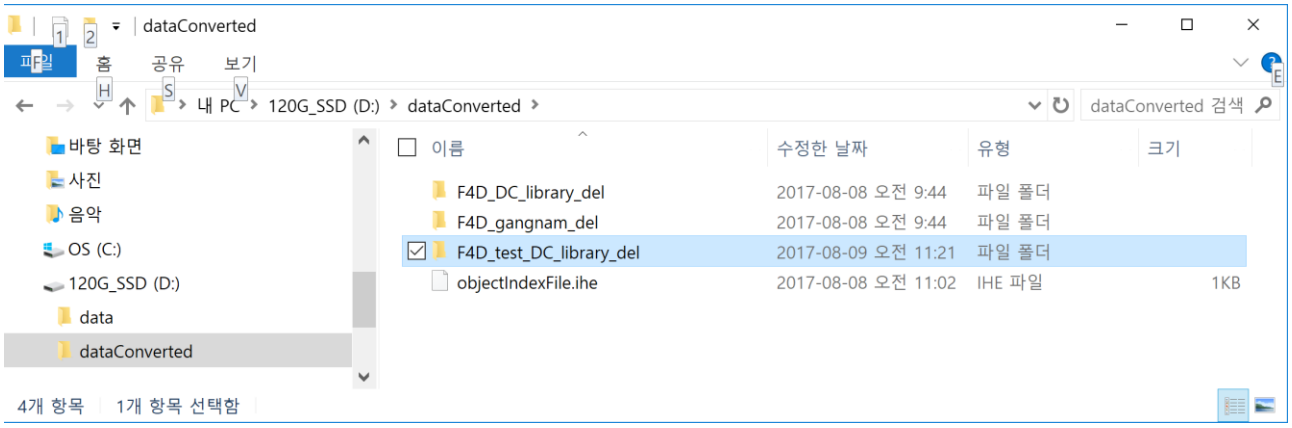

Figure 2. objectIndexFile.ihe

* “Length of Bytes” in all following table contents are described by C++ style.

1.1 HeaderAsimetric.hed

It contains the metadata for 3D GIS of datasets in the F4D format folder and serves as a header file of datasets. Overall structure of HeaderAsimetric.hed is explained in Table 1.

Table 1 Structure of HeaderAsimetric.hed

Name of the Field	Length of Bytes	Remarks
version	5 × sizeof(char) bytes	F4D version info
guidLength	sizeof(int) bytes	Length of guid
guid	[guidLength] × sizeof(char) bytes	guid
longitude latitude	2 × sizeof(double) bytes	Longitude, latitude(degree) of datasets
altitude	sizeof(float) bytes	Altitude(unit: m) of datasets
bbox	6 × sizeof(float) bytes	Bounding box of minX, minY, minZ, maxX, maxY, MaxZ in meter
octree dimension	flexible	Hierarchy info of octree structure

Table 2 Detailed Structure of "octree dimension" Specified in above table

Name of the Field	Length of Bytes	Remarks
depth	sizeof(unsigned int) bytes	Depth of this octree
bbox	6 × sizeof(float) bytes	Bounding box of minX, minY, minZ, maxX, maxY, MaxZ. Only record when depth == 0
childCount	sizeof(unsigned char) bytes	No. of child octree
triangleCount	sizeof(unsigned int) bytes	No. of triangle in this octree
chidrenDimensions	flexible	Recursively record the child octree dimensions

1.2 Bricks Folder

This folder contains low resolution rough geometry information of the objects. F4D converter creates the Lego block style data to increase the rendering speed of large scale 3D objects. This trick could lower the usage of memory of the client and at the same time could increase the rendering speed of large size 3D objects by showing only rough geometry when viewer sees the objects from the long distance.

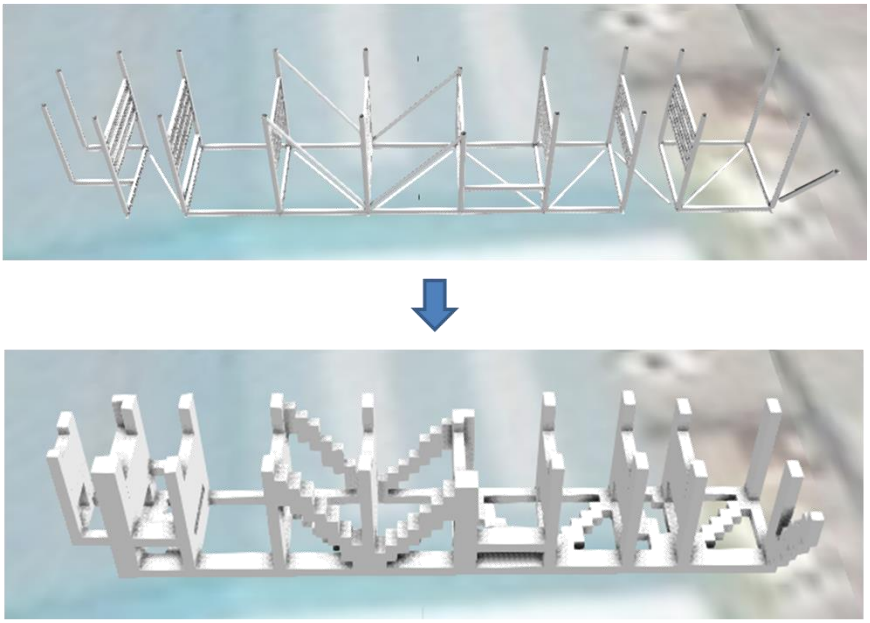

Figure 3 Roughness of objects can be controlled by adjusting block size

Overall structure of Brick file is explained in Table 3.

Table 3 Overall Structure of Brick File

Name of the Field	Length of Bytes	Remarks
bbox	6 × sizeof(float) bytes	Bounding box of minX, minY, minZ, maxX, maxY, MaxZ
vertexCount	sizeof(unsigned int) bytes	No. of vertex in a Brick

vertexPositions	[vertexCount] × 3 × sizeof(float) bytes	x, y, z coordinates of each vertex(unit: m)
bNormal	sizeof(bool) bytes	Whether normal vector existing or not
normalCount	sizeof(unsigned int) bytes	No. of normal vectors(same as No. of vertex count) Only record when bNormal == true
normalVectors	[normalCount] × 3 × sizeof(char) bytes	Normal vectors of x, y, z. Only record when bNormal == true
bColor	sizeof(bool) bytes	Record whether colour array is existing or not
colorCount	sizeof(unsigned int) bytes	Length of colour array(same as No. of vertex count) Only record when bColour == true
vertexColors	[colorCount]×4×sizeof(unsigned char) bytes	Colour values (red, green, blue and alpha) of each vertex. Only record when bColour == true
bTextureCoord	sizeof(bool) bytes	Record whether texture coordinate array is existing or not
texgtureCoordType	sizeof(unsigned short) bytes	GL enumeration value for type of texture coordinate description. Now only 5126(float) type available. Only record when bTextureCoord == true
vertexCount	sizeof(unsigned int) bytes	No. of vertex count. Only record when bTextureCoord == true
textureCoordinates	[vertexCount] × 2 × sizeof(float)	u, v coordinate of each vertex. Only record when bTextureCoord == true

1.3 Models Folder

This folder contains original 3D objects information called Models that constitutes F4D itself. Models will be made to be an instance physically in the 3D space by pairing Reference information in Reference folder. Structure of F4D's Model information is shown in Table 4, 5, 6.

Table 4 Overall Structure of Models File

Name of the Field	Length of Bytes	Remarks
modelCount	sizeof(unsigned int) bytes	No. of Models that References in octree refers
modelData	$\sum_{i=1}^{modelCount} modelSize_i$ bytes	Data of each Model

Table 5 Detailed Structure of Model Data in above table

Name of the Field	Length of Bytes	Remarks
modelIndex	sizeof(unsigned int) bytes	Unique ID of Model. This ID is unique number throughout the datasets
bbox	3 × sizeof(float) bytes	Bounding box of minX, minY, minZ, maxX, maxY, MaxZ
vboCount	sizeof(unsigned int) bytes	No. of VBO(Vertex Buffer Object) that constitutes Model
vboData	$\sum_{i=1}^{vboCount} vboDataSize_i$ bytes	Data of each VBO

Table 6 Detailed Structure of VBO Data in above talbe

Name of the Field	Length of Bytes	Remarks
vertexCount	sizeof(unsigned int) bytes	No. of vertex that constitutes VBO
vertexPositions	[vertexCount] × 3 × sizeof(float) bytes	x, y, z coordinates of each vertex(unit: m)
normalCount	sizeof(unsigned int) bytes	No. of normal vectors(same as vertexCount)
normalVectors	[normalCount] × 3 × sizeof(char) bytes	Normal vectors of x, y, z

indexCount	sizeof(unsigned int) bytes	No. of index of vertex that constitutes triangle
sizeLevels	sizeof(unsigned char) bytes	No. of levels to be used when storing index of triangles in order of size
sizeThresholds	[sizeLevels] × sizeof(float) bytes	Delimiter of each segment
indexMarkers	[sizeLevels] × sizeof(unsigned int) bytes	The positions of the indexes whose size is first changed in the sorted triangle index array.
indexArray	[indexCount] × sizeof(unsigned short) bytes	Array that records indexes of vertex of each triangle

1.4 Reference Folder

This folder contains reference information such as position, ID, indexes of 3D objects that is constituting F4D. By combining Reference information and Models information, real 3D objects will be physically rendered on the described place with attributes (- called as "3D objects are instantiated."). Structure of F4D's Reference information is shown in Table 7, 8, and 9.

Table 7 Overall Structure of Reference File

Name of the Field	Length of Bytes	Remarks
referenceCount	sizeof(unsigned int) bytes	No. of reference in the octree
referenceData	$\sum_{i=0}^{referenceCount} referenceDataSize_i$ bytes	Data of each reference

Table 8 Detailed Structure of Reference Data in above table

Name of the Field	Length of Bytes	Remarks
referenceIndex	sizeof(unsigned int) bytes	Unique ID of reference. This ID is unique number throughout the datasets
objectIdLength	sizeof(unsigned char) bytes	Length of object ID
objectId	[objectIdLength] ×sizeof(char) bytes	Object ID
modelIndex	sizeof(unsigned int) bytes	Unique ID of original model referred by reference information
transformMatrix	16 × sizeof(float) bytes	4 x 4 transformation matrix for placing and rotating the models
bSingleColor	sizeof(bool) bytes	Record whether it uses a default colour
singleColorValueType	sizeof(unsigned short) bytes	Default colour value type in GL enumeration for color value. Now only 5121(unsigned byte) supported. Only record when bSingleColor == true
singleColorDimension	sizeof(unsigned char) bytes	Channel count of default color. Now fixed as 4(RGBA). Only record when bSingleColor == true
singleColor	[colorDimension1] × sizeof(unsigned char) bytes	default colour value of this reference
bVertexColor	sizeof(bool) bytes	Record whether vertex colours are available
bTextureCoord	sizeof(bool) bytes	Record whether texture coordinates are available
vboCount	sizeof(unsigned int) bytes	
vertexColorAndTextureCoordinateInfo	∑ sizeOfvertexColor&textureCoord	vertex colours and texture coordinates of this reference on each vbo

Table 9 Detailed Structure of vertexColorAndTextureCoordinateInfo in above table

Name of the Field	Length of Bytes	Remarks
vertexColorType	sizeof(unsigned short) bytes	GL enumeration value for color. Now only 5121 supported. Only record when bVertexColor in above table == true

vertexColorDimension	sizeof(unsigned char) bytes	Channel count of vertex colour. Now fixed as 3. Only record when bVertexColor in above table == true
vertexCount	sizeof(unsigned int) bytes	Vertex count in this vbo. Only record when bVertexColor in above table == true
vertexColors	[vertexCount] × vertexColorDimension × sizeof(unsigned char) bytes	Vertex color of each vertex. Only record when bVertexColor in above table == true
textureCoordinateValueType	sizeof(unsigned short) bytes	GL enumeration value for texture coordinate. Now firxed as 5126. Only record when bTextureCoord in above table == true
vertexCount	sizeof(unsigned int) bytes	Vertex count in this vbo. Only record when bTextureCoord in above table == true
textureCoordinates	[vertexCount] × 2 × sizeof(float) bytes	u, v coordinate of each vertex. Only record when bTextureCoord in above table == true

1.5 Images_Resized folder

This folder is for image files for texture mapping. Files in this folder are created only when raw data has images for texture. Because WebGL accepts texture images only with width and height of power of 2 pixels, resized texture images are created if necessary.

1.6 SimpleBuildingTexture3x3.png

This file is for texture mapping of meshes in Bricks folder(now 512 by 512). As same as files in Image_resized, this file is created only when raw data has images for texture. In Future, regardless of whether original data has textures or not, texture mapping for lego structures will be supported.

1.7 objectIndexFile.ihe

As described in “1. Format structure”, this file is metadata of full set of all F4D data on web service. Structure of objectIndexFile.ihe is shown in Table 10 and 11.

Table 10 Overall Structure of Reference File

Name of the Field	Length of Bytes	Remarks
dataFolderCount	sizeof(unsigned int) bytes	Count of F4D folders on service
eachDataFolderInfo	∑ dataFolderInfoSize	Simple information of each F4D data folder.

Table 11 Detailed Structure of eachDataFolderInfo in above table

Name of the Field	Length of Bytes	Remarks
dataFolderNameLength	sizeof(unsigned int) bytes	Length of this F4D data folder name
dataFolderName	[dataFolderNameLength] × sizeof(char) bytes	Name of this F4D data folder.
lonLatAlt	sizeof(double) + sizeof(double) + sizeof(float)	Representative longitude, latitude, and altitude of this F4D(Deprecated)
boundingBox	sizeof(float) × 6	minX, minY, minZ, maxX, maxY, maxZ of bounding box of this F4D data