IFC.JSON

Agenda

- IfcJSON Project Overview
- Technical matters
- IfcJSON tools and resources

Team

	Dennis Shelden	New York, US
	Jan Brouwer	Eindhoven, NL
	Pieter Pauwels	Eindhoven, NL
	Nirvik Saha	New York, US
	Devon Sparks	Oregon, US
	Tim McGinley	Copenhagen, DK

and inputs coming in and out

IfcJson Project Overview

Need for ifcJSON

JSON is used throughout the world for exchanging and using data. Building data needs to be available in JSON. Therefore, IFC needs to be available in JSON format.

IFC.JSON aims primarily at addressing the following problems with IFC:

- Many developers have never seen/used EXPRESS or STP instance files before, which increases the effort required to extract data required from them.
- IFC instance populations are typically exchanged as files, which is at odds with linked, distributed, and rapidly changing data seen on most design and construction projects and products.

ifcJSON Criteria

Main focus:

- Backward compatibility
- Round-trip
- Parallel to EXPRESS schema

ifcJSON V4

To a lesser degree (Due to adhering to the IFC schema):

- Human-readability
- Integration with code
- Clear referencing structure
- Direct usability

ifcJSON V5

Project plan

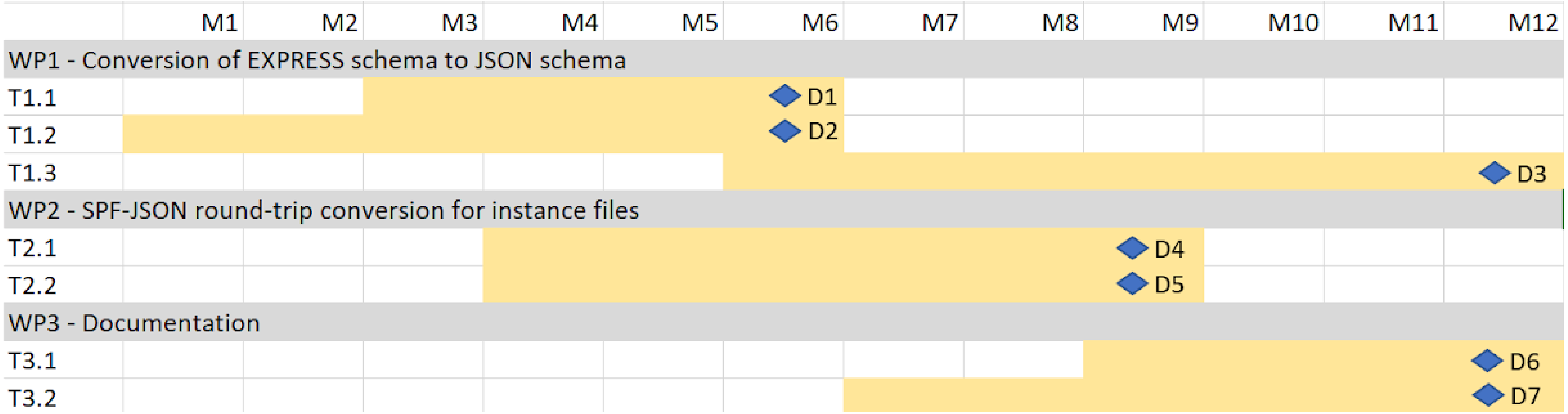
- Started in January 2020
- Activity Proposal submitted & accepted: Spring summit 2020
- Today:
    - Project Proposal submitted
    - Draft deliverables finished:
        - D1: JSON schema: first draft
        - D2: EXPRESS to JSON schema converter (Python): first draft
        - D4: SPF to JSON converter (Python): ready for review
        - D5: JSON to SPF converter (Python): first draft
        - D6: Instance format documentation: ready for review
        - D7: Sample files: ready for review
    - Upcoming deliverables:
        - D3: UML to ifcJSON schema converter

Project timeline

D1: JSON schema
D2: EXPRESS to JSON schema converter
D3: UML to ifcJSON schema converter

D4: SPF to JSON converter
D5: JSON to SPF converter

D6: Instance format documentation
D7: Sample files



	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
WP1 - Conversion of EXPRESS schema to JSON schema												
T1.1			▨	▨	▨	▨ ◆ D1						
T1.2	▨	▨	▨	▨	▨	▨ ◆ D2						
T1.3						▨	▨	▨	▨	▨	▨	▨ ◆ D3
WP2 - SPF-JSON round-trip conversion for instance files												
T2.1				▨	▨	▨	▨	▨	▨ ◆ D4			
T2.2				▨	▨	▨	▨	▨	▨ ◆ D5			
WP3 - Documentation												
T3.1									▨	▨	▨	▨ ◆ D6
T3.2							▨	▨	▨	▨	▨	▨ ◆ D7

WP1 Conversion of EXPRESS schema to JSON schema

D1: JSON schema: a single json schema file that is published on the web (buildingSMART webspace) and can be used for validation.

D2: **EXPRESS to JSON schema converter**: a Python-based converter is developed that generates a JSON schema based on EXPRESS file input.

D3: UML to JSON schema converter: a Python-based converter is developed that generates a JSON schema based on the UML schema for IFC (file input).

WP2 SPF-JSON round-trip conversion for instance files

D4: SPF to JSON converter: a Python-based converter is developed that generates a JSON file based on SPF file input.

D5: JSON to SPF converter: a Python-based converter is developed that generates a SPF file based on JSON file input.

WP3 Documentation

D6: Instance format documentation: The documentation focuses on the instance files. A report is made available via GitHub and PDF that documents the JSON format for IFC. This includes recommendations for the future.

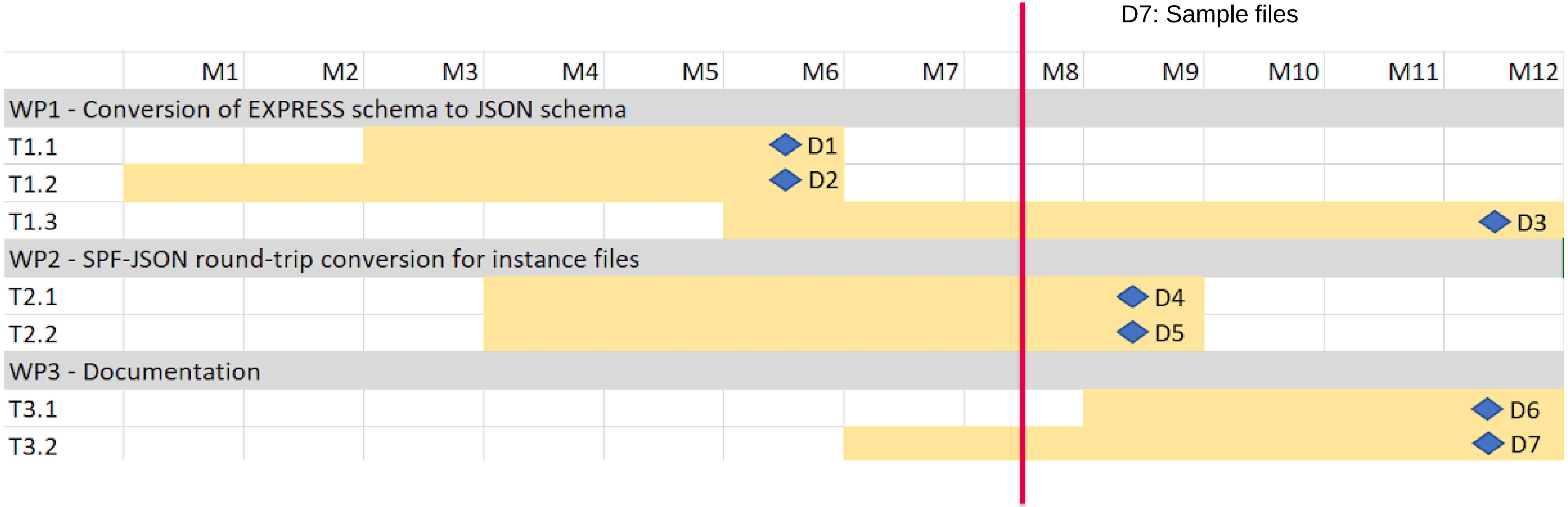
D7: Sample files: As part of the documentation, a number of JSON sample files are included. They will be published under the buildingSMART test file repository in buildingSMART.

Project timeline

D1: JSON schema
D2: EXPRESS to JSON schema converter
D3: UML to ifcJSON schema converter

D4: SPF to JSON converter
D5: JSON to SPF converter

D6: Instance format documentation
D7: Sample files



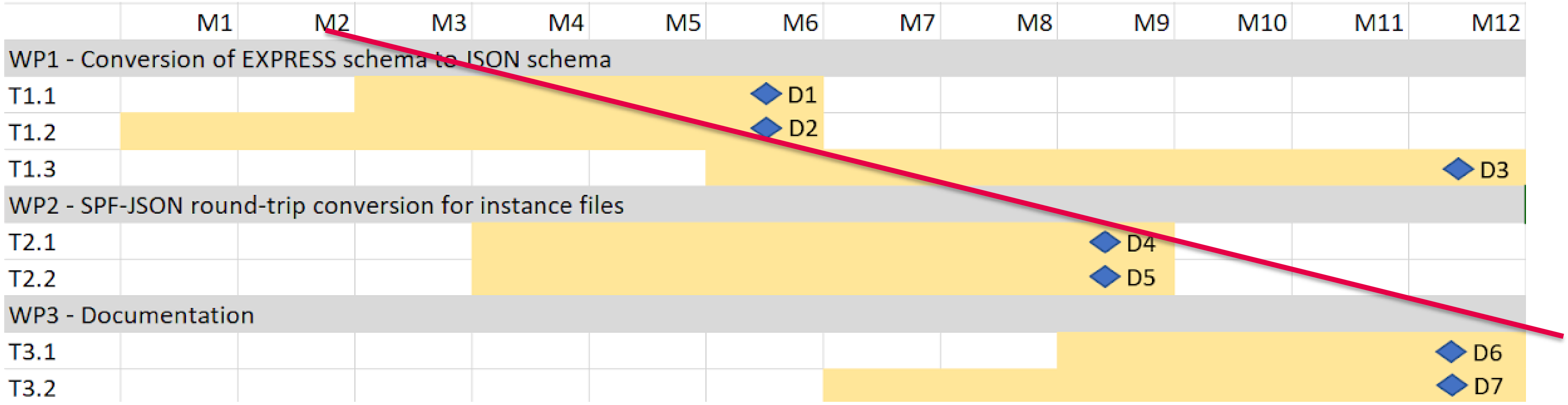
	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
WP1 - Conversion of EXPRESS schema to JSON schema												
T1.1			▨	▨	▨	▨ ◆ D1						
T1.2	▨	▨	▨	▨	▨	▨ ◆ D2						
T1.3						▨	▨	▨	▨	▨	▨	▨ ◆ D3
WP2 - SPF-JSON round-trip conversion for instance files												
T2.1				▨	▨	▨	▨	▨	▨ ◆ D4			
T2.2				▨	▨	▨	▨	▨	▨ ◆ D5			
WP3 - Documentation												
T3.1									▨	▨	▨	▨ ◆ D6
T3.2							▨	▨	▨	▨	▨	▨ ◆ D7

Project timeline

D1: JSON schema
D2: EXPRESS to JSON schema converter
D3: UML to ifcJSON schema converter

D4: SPF to JSON converter
D5: JSON to SPF converter

D6: Instance format documentation
D7: Sample files



	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
WP1 - Conversion of EXPRESS schema to JSON schema												
T1.1			🟨	🟨	🟨	🟨 ◆ D1						
T1.2	🟨	🟨	🟨	🟨	🟨	🟨 ◆ D2						
T1.3						🟨	🟨	🟨	🟨	🟨	🟨	🟨 ◆ D3
WP2 - SPF-JSON round-trip conversion for instance files												
T2.1				🟨	🟨	🟨	🟨	🟨	🟨 ◆ D4			
T2.2				🟨	🟨	🟨	🟨	🟨	🟨 ◆ D5			
WP3 - Documentation												
T3.1									🟨	🟨	🟨	🟨 ◆ D6
T3.2							🟨	🟨	🟨	🟨	🟨	🟨 ◆ D7

buildingSMART

Technical matters

About JavaScript Object Notation (JSON)

```
{ // A dictionary or object
  "Key 1": "Object 1",
  "Key 2": "Value 1"
}
```

```
[   // An Array
  {
    "Key 1": "Object 1",
    "Key 2": "Value 1"
  },
  {
    "Key 1": "Object 2",
    "Key 2": "Value 2"
  },
  {
    "Key 1": "Object 3",
    "Key 2": "Value 3",
    "Key 3": "Value 4" // Flexible
  }
]
```

JSON Serialize / De-serialize

```
‘[    // JSON TEXT
  {
    "Key_1": "Object 1",
    "Key_2": "Value 1"
  },
  {
    "Key_1": "Object 2",
    "Key_2": "Value 2"
  },
  {
    "Key_1": "Object 3",
    "Key_2": "Value 3",
    "Key_3": "Value 4" // Flexible
  }
]’
```

Deserialize **`JSON.Parse()`** →

← *Serialize* **`JSON.Stringify()`**

```
[    //Python data
  {
    K
    K
  },
  {
    K
    K
  },
  {
    K
    K
    K
  }
]
```

```
[    // JavaScript data
  {
    Key_1: "Object 1",
    Key_2: "Value 1"
  },
  {
    Key_1: "Object 2",
    Key_2: "Value 2"
  },
  {
    Key_1: "Object 3",
    Key_2: "Value 3",
    Key_3: "Value 4"
  }
]
```

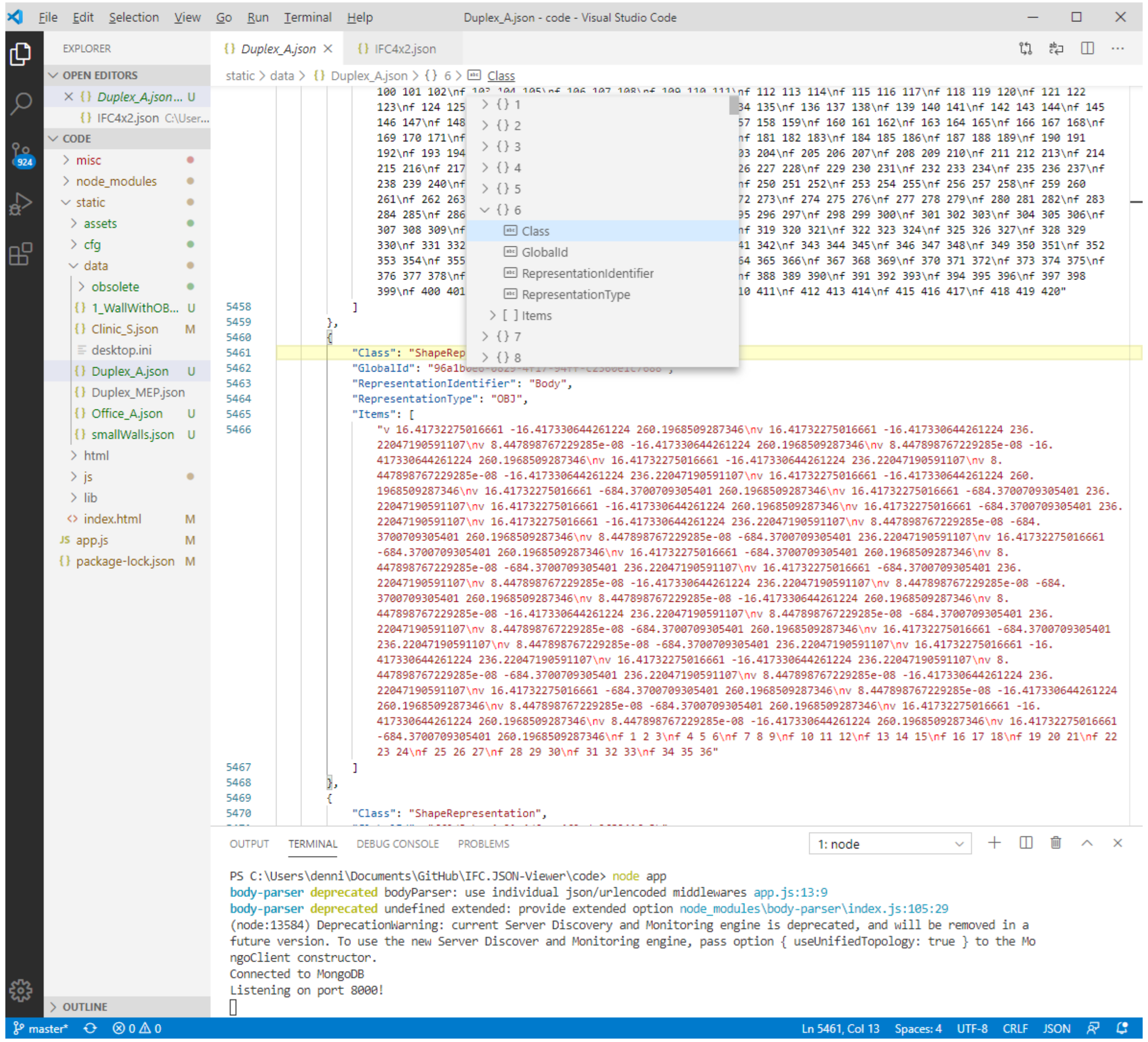
JavaScript & Python natively supported
Compatible with C#, Java, ...

Ifc.JSON

```
[  // A flat array of Ifc.JSON Objects
  {
    "name": "Ifc Object 1",
    "type": "IfcWall",
    "globalId": "68485662-4a08-4f7d-ad9f-379798fee4b2"
  },
  {
    "name": "Ifc Object 2",
    "type": "IfcDoor",
    "globalId": "32cfdee2-71b8-438f-b0b4-0a2a5a05184a"
  },
]
```

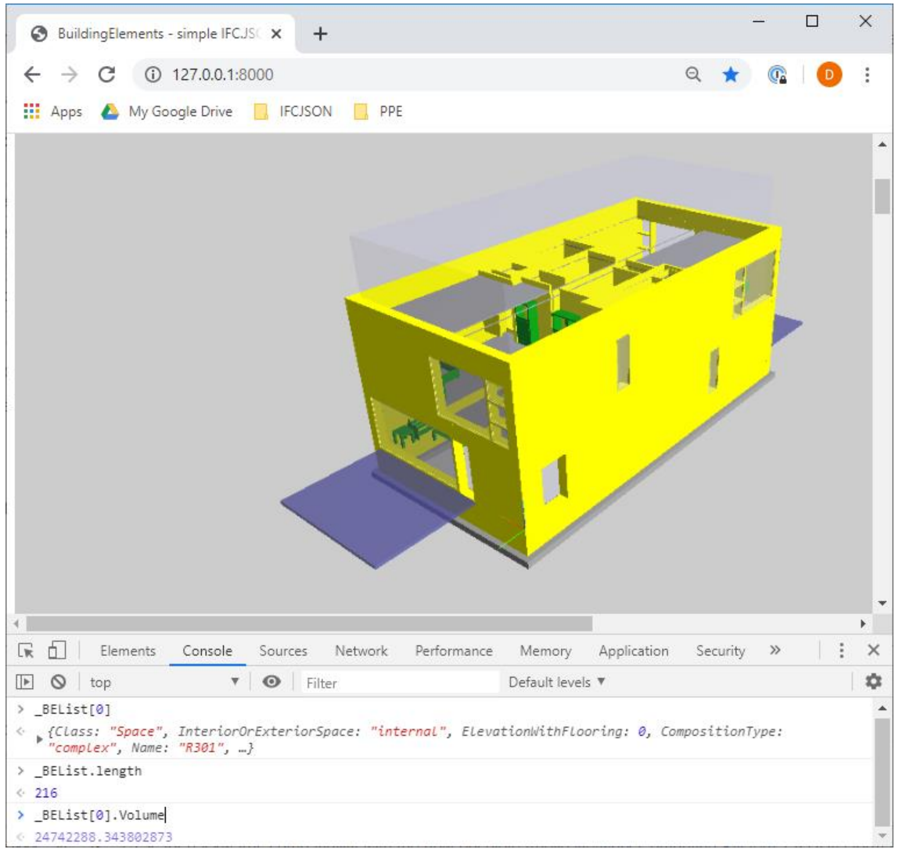
```
[   // An Ifc.JSON Project hierarchy
    {
      "type": "IfcProject",
      "globalId": "cb78a8c2-fb1e-4e12-8f29-6c0d7c39ca0b",
      "name": "Default Project",
      "description": "Description of Default Project",
      "isDecomposedBy": [
        {
          "type": "IfcSite",
          "globalId": "f07e69ce-3709-4ef5-a029-e27de7e95991",
          "name": "TU/e campus",
          "description": "The High Tech campus",
          "compositionType": "ELEMENT",
          "refElevation": 0,
          "isDecomposedBy": [
            {
              "type": "IfcBuilding",
              "globalId": "f3b41796-63ea-4a63-b0aa-
f1d7978a6e47",
              "name": "Vertigo Building",
              "description": "TU/e Department…",
              "compositionType": "ELEMENT",
              "elevationOfRefHeight": 0,
              "elevationOfTerrain": 0,
              "isDecomposedBy": [ …
```

JSON Tool support



```
            100 101 102\nf 103 104 105\nf 106 107 108\nf 109 110 111\nf 112 113 114\nf 115 116 117\nf 118 119 120\nf 121 122
            123\nf 124 125                                              34 135\nf 136 137 138\nf 139 140 141\nf 142 143 144\nf 145
            146 147\nf 148                                              57 158 159\nf 160 161 162\nf 163 164 165\nf 166 167 168\nf
            169 170 171\nf                                              nf 181 182 183\nf 184 185 186\nf 187 188 189\nf 190 191
            192\nf 193 194                                              03 204\nf 205 206 207\nf 208 209 210\nf 211 212 213\nf 214
            215 216\nf 217                                              26 227 228\nf 229 230 231\nf 232 233 234\nf 235 236 237\nf
            238 239 240\nf                                              nf 250 251 252\nf 253 254 255\nf 256 257 258\nf 259 260
            261\nf 262 263                                              72 273\nf 274 275 276\nf 277 278 279\nf 280 281 282\nf 283
            284 285\nf 286                                              95 296 297\nf 298 299 300\nf 301 302 303\nf 304 305 306\nf
            307 308 309\nf                                              nf 319 320 321\nf 322 323 324\nf 325 326 327\nf 328 329
            330\nf 331 332                                              41 342\nf 343 344 345\nf 346 347 348\nf 349 350 351\nf 352
            353 354\nf 355                                              64 365 366\nf 367 368 369\nf 370 371 372\nf 373 374 375\nf
            376 377 378\nf                                              nf 388 389 390\nf 391 392 393\nf 394 395 396\nf 397 398
            399\nf 400 401                                              10 411\nf 412 413 414\nf 415 416 417\nf 418 419 420"
        ]
    },
    {
        "Class": "ShapeRep
        "GlobalId": "96a1b
        "RepresentationIdentifier": "Body",
        "RepresentationType": "OBJ",
        "Items": [
            "v 16.41732275016661 -16.417330644261224 260.1968509287346\nv 16.41732275016661 -16.417330644261224 236.
            22047190591107\nv 8.447898767229285e-08 -16.417330644261224 260.1968509287346\nv 8.447898767229285e-08 -16.
            417330644261224 260.1968509287346\nv 16.41732275016661 -16.417330644261224 236.22047190591107\nv 8.
            447898767229285e-08 -16.417330644261224 236.22047190591107\nv 16.41732275016661 -16.417330644261224 260.
            1968509287346\nv 16.41732275016661 -684.3700709305401 260.1968509287346\nv 16.41732275016661 -684.3700709305401 236.
            22047190591107\nv 16.41732275016661 -16.417330644261224 260.1968509287346\nv 16.41732275016661 -684.3700709305401 236.
            22047190591107\nv 16.41732275016661 -16.417330644261224 236.22047190591107\nv 8.447898767229285e-08 -684.
            3700709305401 260.1968509287346\nv 8.447898767229285e-08 -684.3700709305401 236.22047190591107\nv 16.41732275016661
            -684.3700709305401 260.1968509287346\nv 16.41732275016661 -684.3700709305401 260.1968509287346\nv 8.
            447898767229285e-08 -684.3700709305401 236.22047190591107\nv 16.41732275016661 -684.3700709305401 236.
            22047190591107\nv 8.447898767229285e-08 -16.417330644261224 236.22047190591107\nv 8.447898767229285e-08 -684.
            3700709305401 260.1968509287346\nv 8.447898767229285e-08 -16.417330644261224 260.1968509287346\nv 8.
            447898767229285e-08 -16.417330644261224 236.22047190591107\nv 8.447898767229285e-08 -684.3700709305401 236.
            22047190591107\nv 8.447898767229285e-08 -684.3700709305401 260.1968509287346\nv 16.41732275016661 -684.3700709305401
            236.22047190591107\nv 8.447898767229285e-08 -684.3700709305401 236.22047190591107\nv 16.41732275016661 -16.
            417330644261224 236.22047190591107\nv 16.41732275016661 -16.417330644261224 236.22047190591107\nv 8.
            447898767229285e-08 -684.3700709305401 236.22047190591107\nv 8.447898767229285e-08 -16.417330644261224 236.
            22047190591107\nv 16.41732275016661 -684.3700709305401 260.1968509287346\nv 8.447898767229285e-08 -16.417330644261224
            260.1968509287346\nv 8.447898767229285e-08 -684.3700709305401 260.1968509287346\nv 16.41732275016661 -16.
            417330644261224 260.1968509287346\nv 8.447898767229285e-08 -16.417330644261224 260.1968509287346\nv 16.41732275016661
            -684.3700709305401 260.1968509287346\nf 1 2 3\nf 4 5 6\nf 7 8 9\nf 10 11 12\nf 13 14 15\nf 16 17 18\nf 19 20 21\nf 22
            23 24\nf 25 26 27\nf 28 29 30\nf 31 32 33\nf 34 35 36"
        ]
    },
    {
        "Class": "ShapeRepresentation",
```

```
PS C:\Users\denni\Documents\GitHub\IFC.JSON-Viewer\code> node app
body-parser deprecated bodyParser: use individual json/urlencoded middlewares app.js:13:9
body-parser deprecated undefined extended: provide extended option node_modules\body-parser\index.js:105:29
(node:13584) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a
future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the Mo
ngoClient constructor.
Connected to MongoDB
Listening on port 8000!
```



```
> _BEList[0]
<  {Class: "Space", InteriorOrExteriorSpace: "internal", ElevationWithFlooring: 0, CompositionType:
   "complex", Name: "R301", …}
> _BEList.length
< 216
> _BEList[0].Volume
< 24742288.343802873
```

IFC objects -> Ifc.JSON

```
myIfcJSONText = JSON.stringify( myIfcObjects );
```

Ifc.JSON -> IFC objects

```
myIfcObjects = JSON.parse(myIfcJSONText );
```

Built into Visual Studio Code

Built into Chrome and Node.js

Querying

Multiple tools and libraries, often 1 line of code

JSONPath

"walks" a JSON hierarchy and returns an array of objects with match the criteria

```
OBJARR = jsonPath(JSONARR, "$..[?(@.representationType=='OBJ')]");
```

Mongo query

queries a collection for objects

```
myDatabase.collection(myModelName).find(myQuery);
```

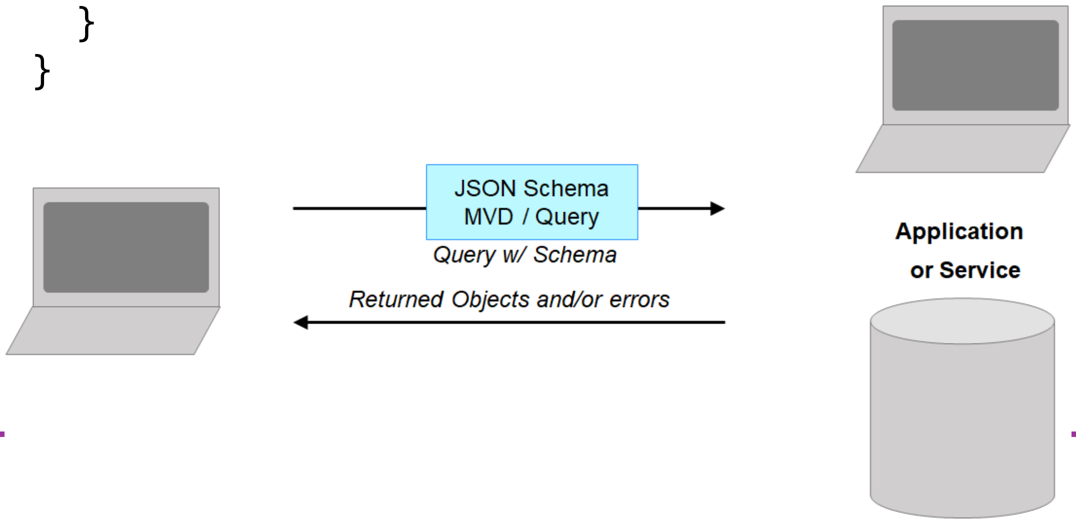
JSON Schema query

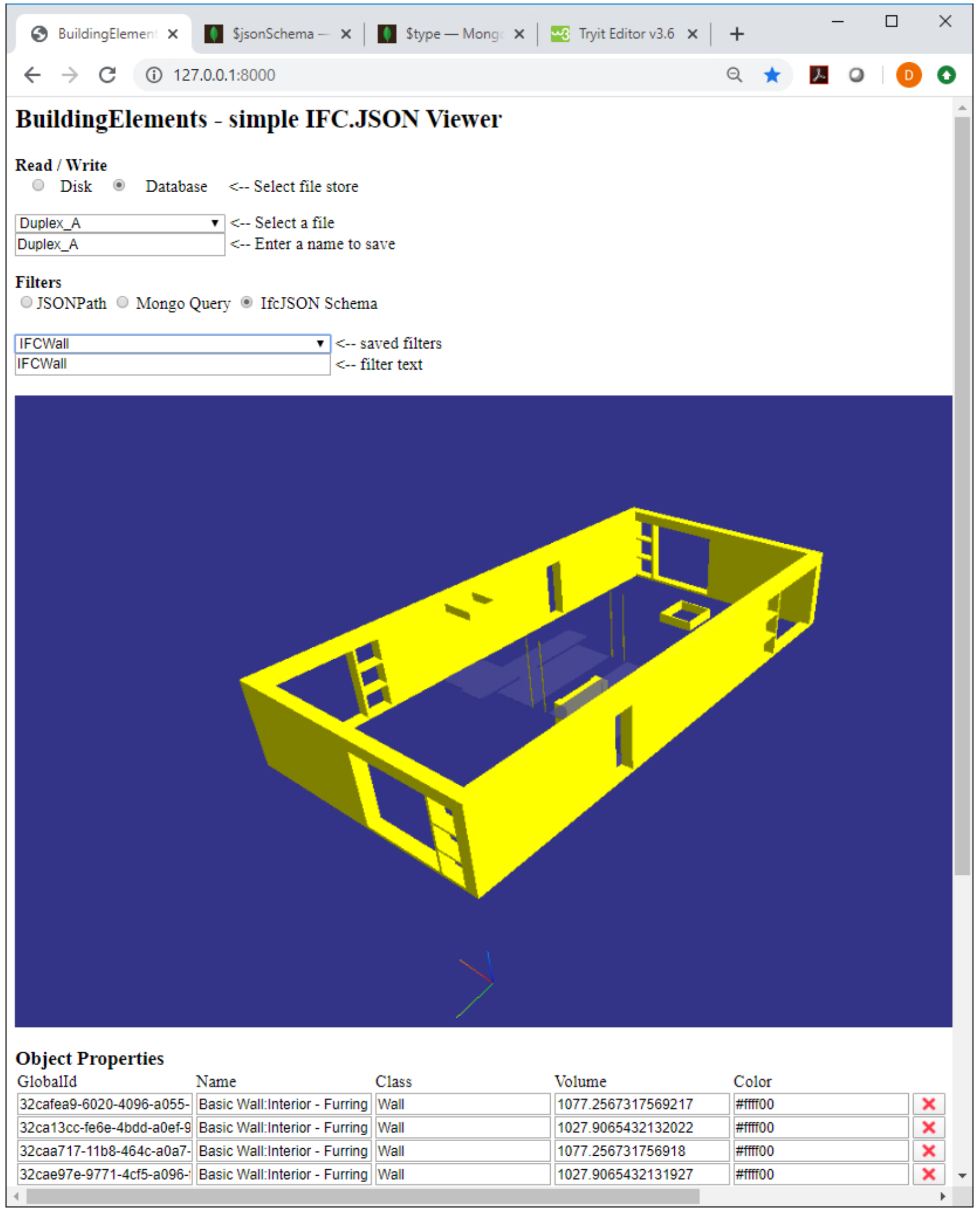
queries or validates a collection of objects against a schema

```
myDatabase.collection(myModelName).find({ $jsonSchema:mySchema);
```

JSON Schema

```
{
    "bsonType": "object",
    "required": ["type"],
    "properties": {
      "type": {
        "type": "string",
         "enum": ["IfcWall", "IfcSlab", "IfcShapeRepresentatio
         n"],
      },
      "Volume": {
        "type": "double",
        "maximum": 10000,
      }
    }
  }
```





BuildingElements - simple IFC.JSON Viewer

Read / Write
Disk [✓ Database] <-- Select file store

Duplex_A <-- Select a file
Duplex_A <-- Enter a name to save

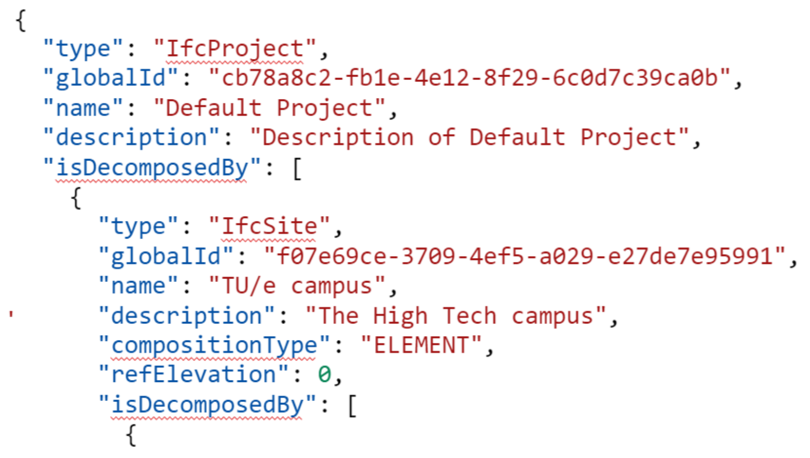
Filters
JSONPath Mongo Query [✓ IfcJSON Schema]

IFCWall <-- saved filters
IFCWall <-- filter text

Object Properties

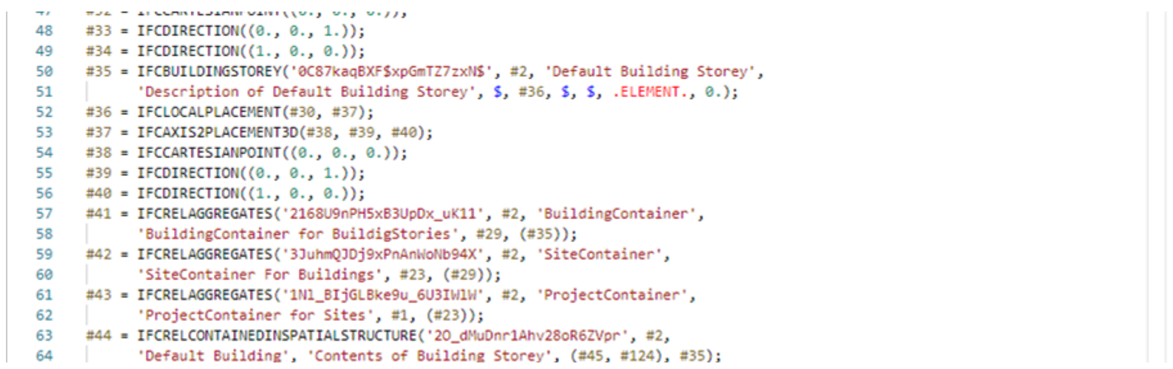
GlobalId	Name	Class	Volume	Color	
32cafea9-6020-4096-a055-	Basic Wall:Interior - Furring	Wall	1077.2567317569217	#ffff00	✕
32ca13cc-fe6e-4bdd-a0ef-9	Basic Wall:Interior - Furring	Wall	1027.9065432132022	#ffff00	✕
32caa717-11b8-464c-a0a7-	Basic Wall:Interior - Furring	Wall	1077.256731756918	#ffff00	✕
32cae97e-9771-4cf5-a096-	Basic Wall:Interior - Furring	Wall	1027.9065432131927	#ffff00	✕

IfcJSON



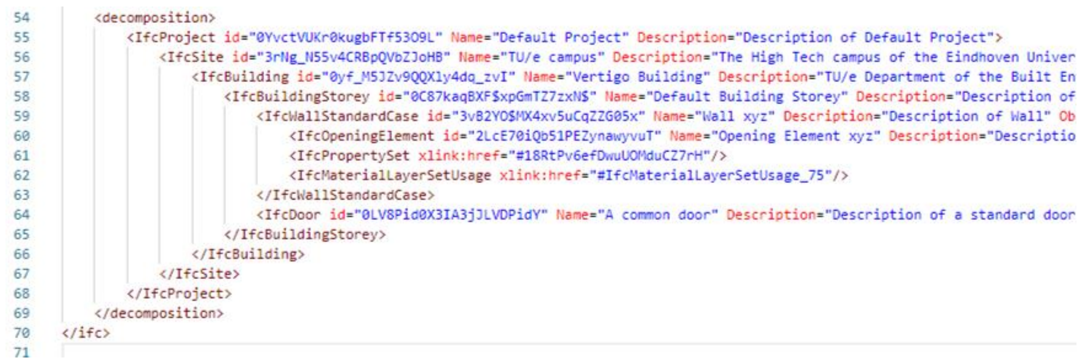
```json
{
  "type": "IfcProject",
  "globalId": "cb78a8c2-fb1e-4e12-8f29-6c0d7c39ca0b",
  "name": "Default Project",
  "description": "Description of Default Project",
  "isDecomposedBy": [
    {
      "type": "IfcSite",
      "globalId": "f07e69ce-3709-4ef5-a029-e27de7e95991",
      "name": "TU/e campus",
      "description": "The High Tech campus",
      "compositionType": "ELEMENT",
      "refElevation": 0,
      "isDecomposedBy": [
        {
```

Ifc-SPF



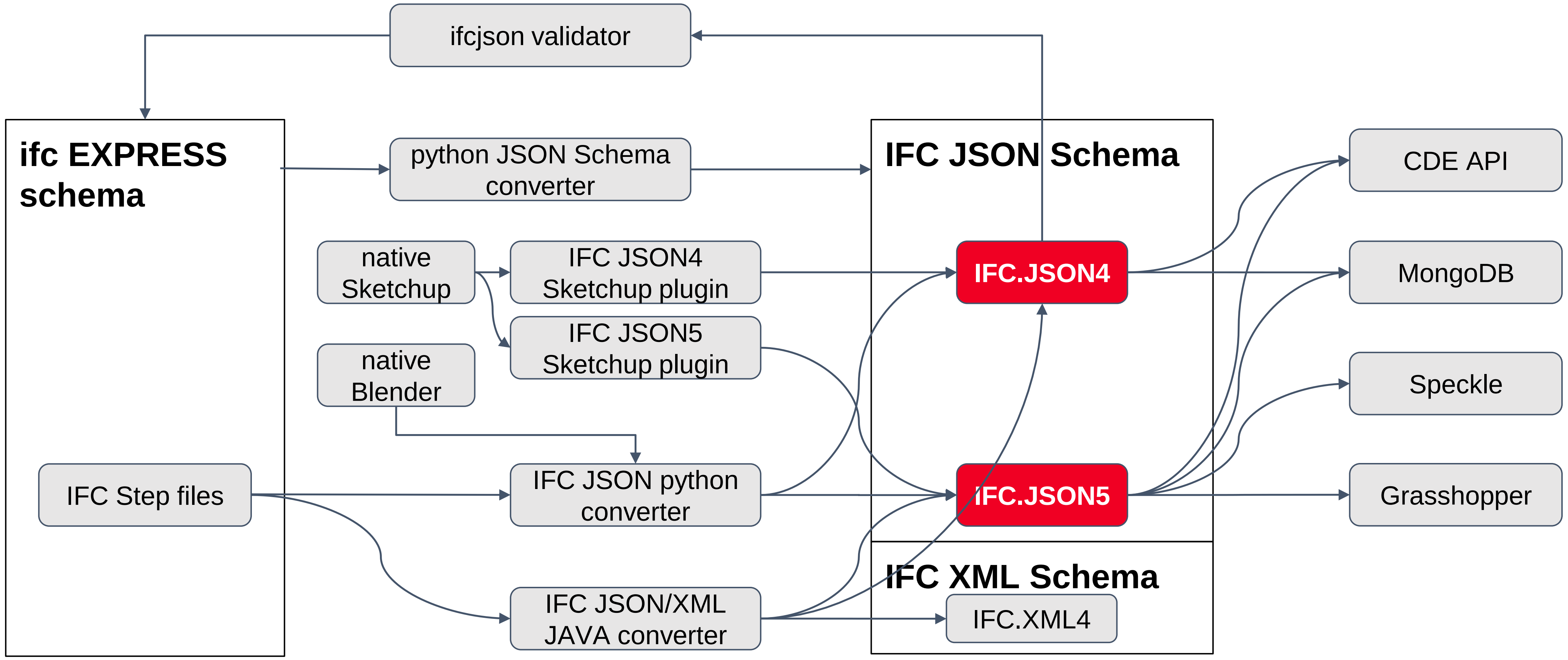
```
#33 = IFCDIRECTION((0., 0., 1.));
#34 = IFCDIRECTION((1., 0., 0.));
#35 = IFCBUILDINGSTOREY('0C87kaqBXF$xpGmTZ7zxN$', #2, 'Default Building Storey',
       'Description of Default Building Storey', $, #36, $, $, .ELEMENT., 0.);
#36 = IFCLOCALPLACEMENT(#30, #37);
#37 = IFCAXIS2PLACEMENT3D(#38, #39, #40);
#38 = IFCCARTESIANPOINT((0., 0., 0.));
#39 = IFCDIRECTION((0., 0., 1.));
#40 = IFCDIRECTION((1., 0., 0.));
#41 = IFCRELAGGREGATES('2168U9nPH5xB3UpDx_uK11', #2, 'BuildingContainer',
       'BuildingContainer for BuildigStories', #29, (#35));
#42 = IFCRELAGGREGATES('3JuhmQJDj9xPnAnWoNb94X', #2, 'SiteContainer',
       'SiteContainer For Buildings', #23, (#29));
#43 = IFCRELAGGREGATES('1N1_BIjGLBke9u_6U3IW1W', #2, 'ProjectContainer',
       'ProjectContainer for Sites', #1, (#23));
#44 = IFCRELCONTAINEDINSPATIALSTRUCTURE('2O_dMuDnr1Ahv28oR6ZVpr', #2,
       'Default Building', 'Contents of Building Storey', (#45, #124), #35);
```

IfcXML



```xml
    <decomposition>
        <IfcProject id="0YvctVUKr0kugbFTf53O9L" Name="Default Project" Description="Description of Default Project">
            <IfcSite id="3rNg_N55v4CRBpQVbZJoHB" Name="TU/e campus" Description="The High Tech campus of the Eindhoven Univer
                <IfcBuilding id="0yf_M5JZv9QQXly4dq_zvI" Name="Vertigo Building" Description="TU/e Department of the Built En
                    <IfcBuildingStorey id="0C87kaqBXF$xpGmTZ7zxN$" Name="Default Building Storey" Description="Description of
                        <IfcWallStandardCase id="3vB2YO$MX4xv5uCqZZG05x" Name="Wall xyz" Description="Description of Wall" Ob
                            <IfcOpeningElement id="2LcE70iQb51PEZynawyvuT" Name="Opening Element xyz" Description="Descriptio
                            <IfcPropertySet xlink:href="#18RtPv6efDwuUOMduCZ7rH"/>
                            <IfcMaterialLayerSetUsage xlink:href="#IfcMaterialLayerSetUsage_75"/>
                        </IfcWallStandardCase>
                        <IfcDoor id="0LV8Pid0X3IA3jJLVDPidY" Name="A common door" Description="Description of a standard door
                    </IfcBuildingStorey>
                </IfcBuilding>
            </IfcSite>
        </IfcProject>
    </decomposition>
</ifc>
```

IfcJson Ecosystem

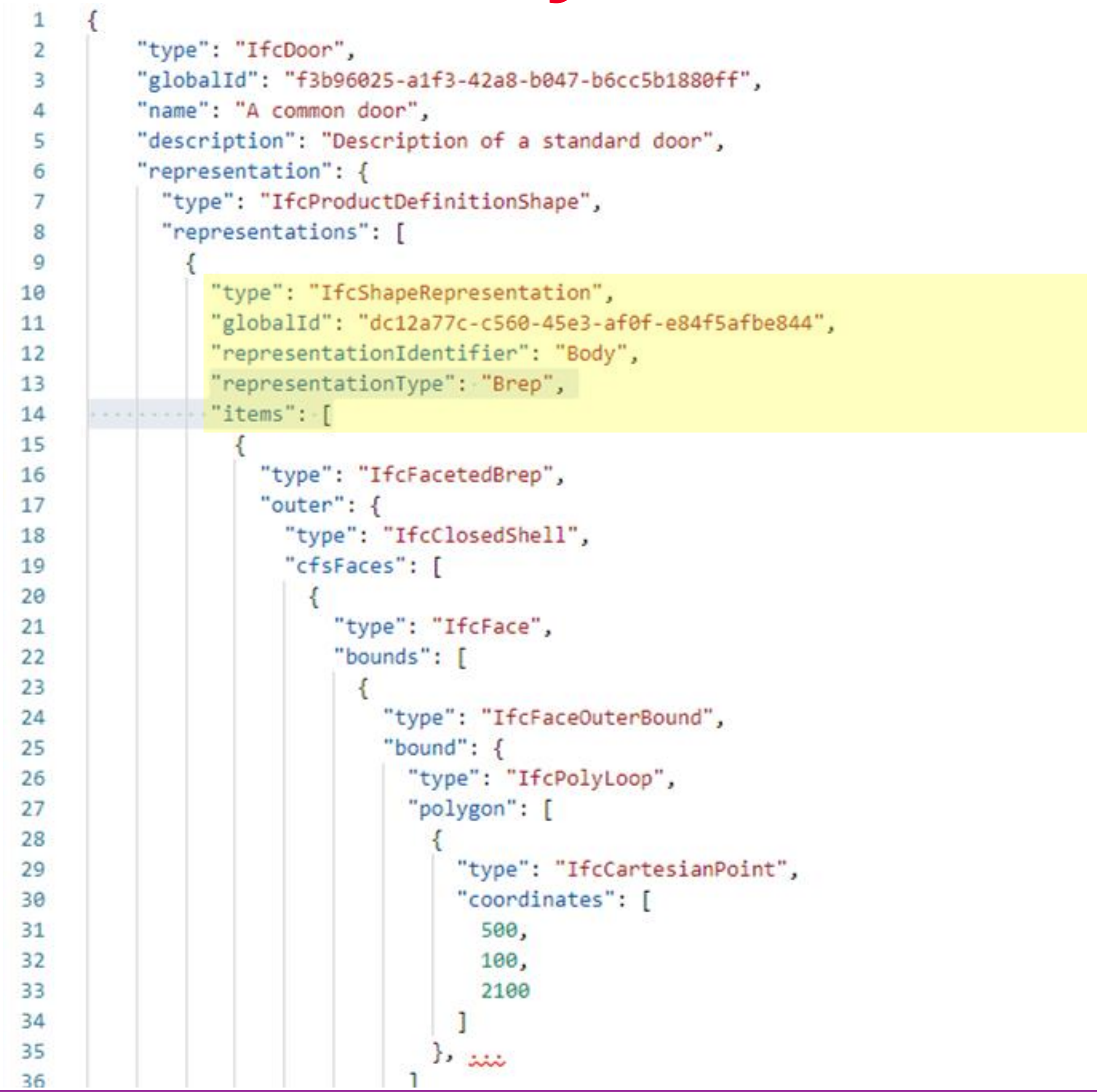


Recommendations for IFC5

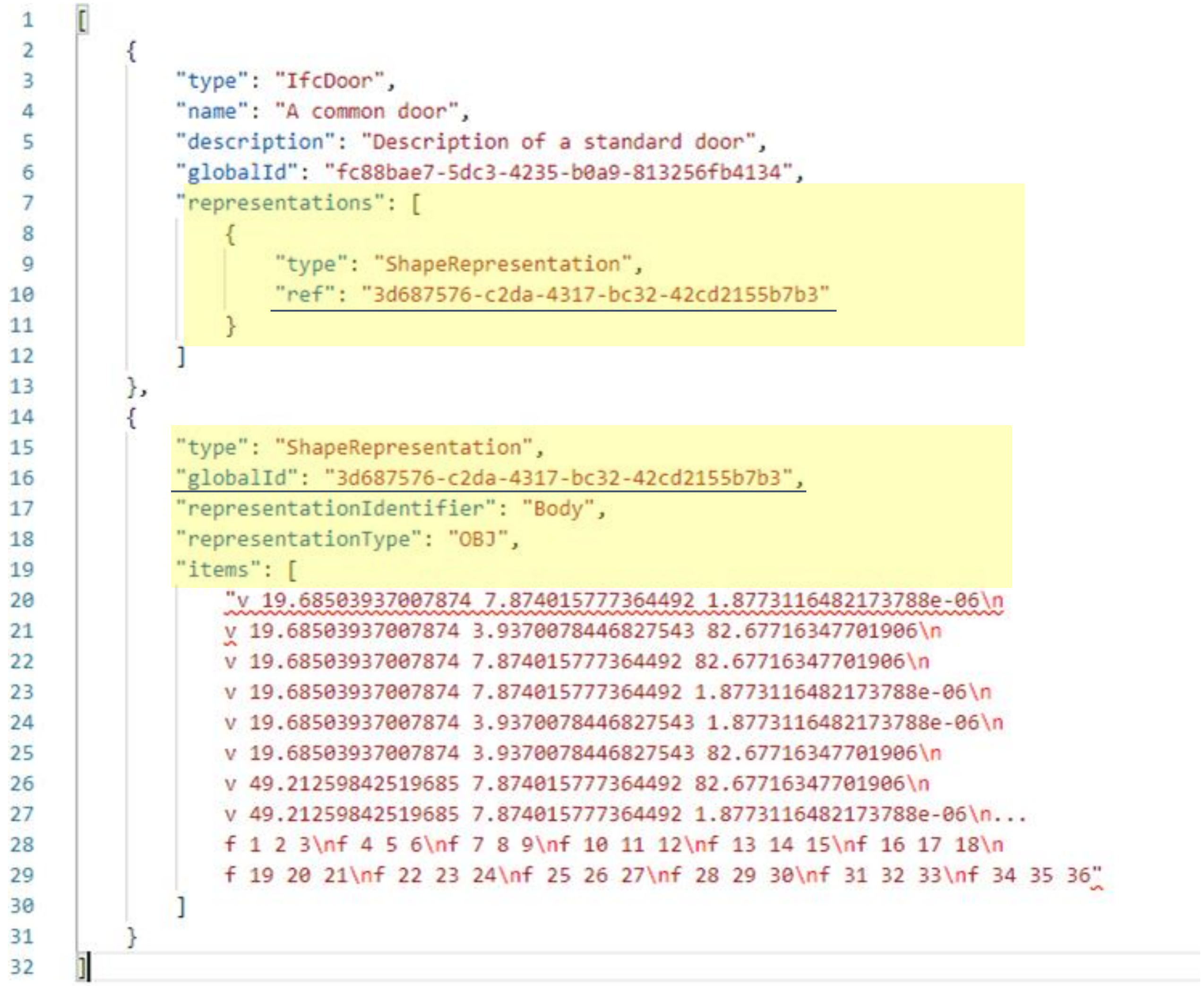
- Use JSON hierarchy
- Limit requirements and restrictions at the base schema
- Keep out null, unknown and empty values
- Support OBJ, FBX, … geometry
- Parameters at the root object
- Unique identifiers

```
[
  {
    "type": "Building",
    "globalId": "f3b41796-63ea-4a63-b0aa-f1d7978a6e47",
    "name": "Vertigo Building",
    "description": "TU/e Department of the Built Environment",
    "compositionType": "ELEMENT",
    "isDecomposedBy": [
      {
        "type": "BuildingStorey",
        "globalId": "f3b837ed-c73a-422c-80f1-621164f4d99f",
        "name": "Default Building Storey",
        "description": "Description of Default Building Storey",
        "compositionType": "ELEMENT",
        "elevation": 0,
        "containsElements": [
          {
            "type": "Wall",
            "globalId": "f3b7a52f-4eb5-44a8-80e0-87592507aed9",
            "name": "Wall xyz",
            "description": "Description of Wall",
            "representation": {
              "type": "ProductDefinitionShape",
              "representations": [
                {
                  "type": "OBJ",
                  "ref": "9b76f770-b9ea-4c50-ae00-97b5105644d5"
                },
                {
                  "type": "Brep",
                  "ref": "dc12a77c-c560-45e3-af0f-e84f5afbe844"
                }
                 ...
```

Geometry



```
{
    "type": "IfcDoor",
    "globalId": "f3b96025-a1f3-42a8-b047-b6cc5b1880ff",
    "name": "A common door",
    "description": "Description of a standard door",
    "representation": {
      "type": "IfcProductDefinitionShape",
      "representations": [
        {
          "type": "IfcShapeRepresentation",
          "globalId": "dc12a77c-c560-45e3-af0f-e84f5afbe844",
          "representationIdentifier": "Body",
          "representationType": "Brep",
          "items": [
            {
              "type": "IfcFacetedBrep",
              "outer": {
                "type": "IfcClosedShell",
                "cfsFaces": [
                  {
                    "type": "IfcFace",
                    "bounds": [
                      {
                        "type": "IfcFaceOuterBound",
                        "bound": {
                          "type": "IfcPolyLoop",
                          "polygon": [
                            {
                              "type": "IfcCartesianPoint",
                              "coordinates": [
                                500,
                                100,
                                2100
                              ]
                            }, ...
                          ]
```



```
[
    {
        "type": "IfcDoor",
        "name": "A common door",
        "description": "Description of a standard door",
        "globalId": "fc88bae7-5dc3-4235-b0a9-813256fb4134",
        "representations": [
            {
                "type": "ShapeRepresentation",
                "ref": "3d687576-c2da-4317-bc32-42cd2155b7b3"
            }
        ]
    },
    {
        "type": "ShapeRepresentation",
        "globalId": "3d687576-c2da-4317-bc32-42cd2155b7b3",
        "representationIdentifier": "Body",
        "representationType": "OBJ",
        "items": [
            "v 19.68503937007874 7.874015777364492 1.8773116482173788e-06\n
            v 19.68503937007874 3.9370078446827543 82.67716347701906\n
            v 19.68503937007874 7.874015777364492 82.67716347701906\n
            v 19.68503937007874 7.874015777364492 1.8773116482173788e-06\n
            v 19.68503937007874 3.9370078446827543 1.8773116482173788e-06\n
            v 19.68503937007874 3.9370078446827543 82.67716347701906\n
            v 49.21259842519685 7.874015777364492 82.67716347701906\n
            v 49.21259842519685 7.874015777364492 1.8773116482173788e-06\n...
            f 1 2 3\nf 4 5 6\nf 7 8 9\nf 10 11 12\nf 13 14 15\nf 16 17 18\n
            f 19 20 21\nf 22 23 24\nf 25 26 27\nf 28 29 30\nf 31 32 33\nf 34 35 36"
        ]
    }
]
```

Brep Geometry

OBJ Geometry

IfcJSON Schema

IfcJSON Schema is a specification for JSON-based IFC data.

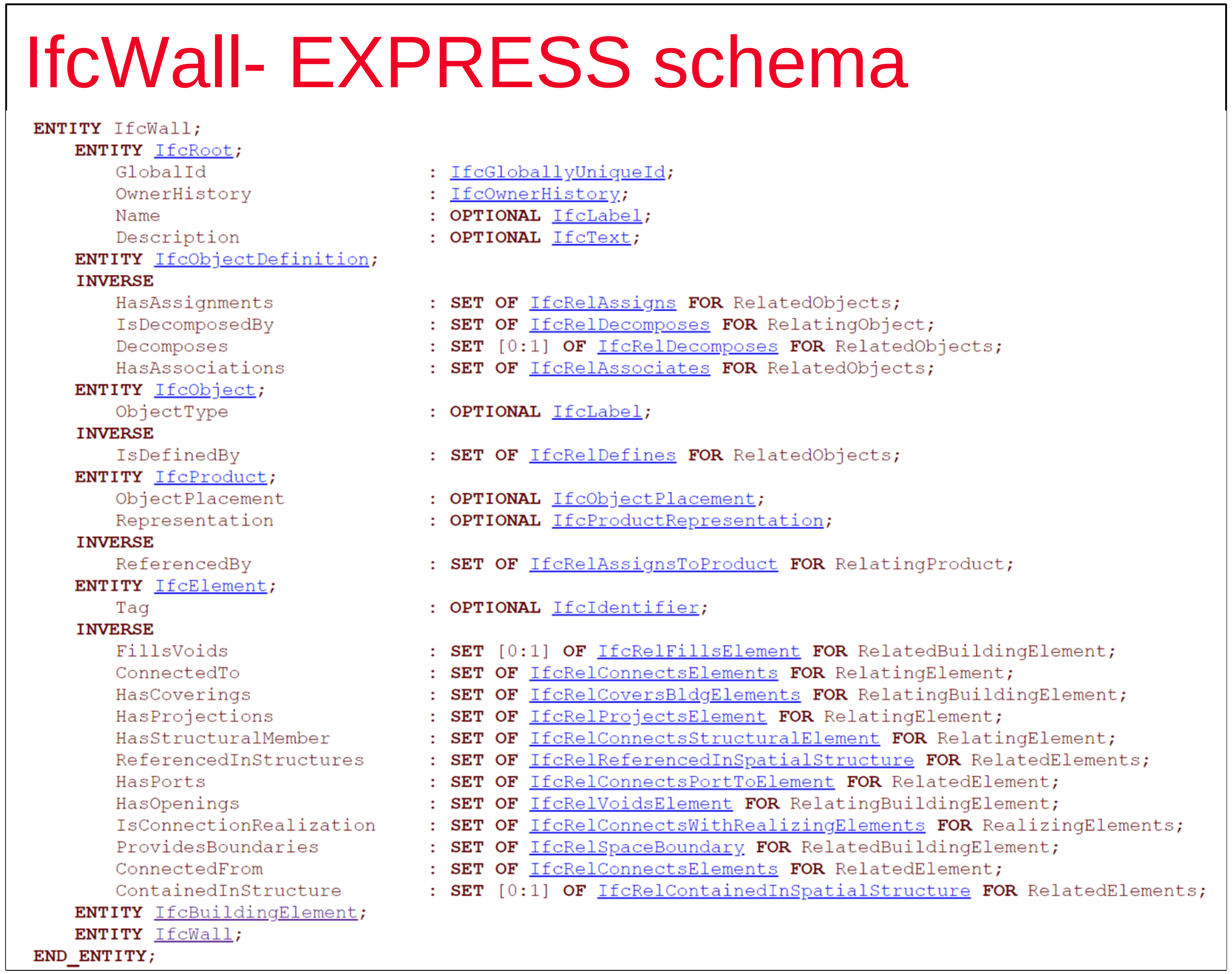
Our approach:

- ❖ Compliance with IFC specification
- ❖ Using JSON syntactic framework
- ❖ Adopting JSON schema specification proposed by IETF

Main objectives:

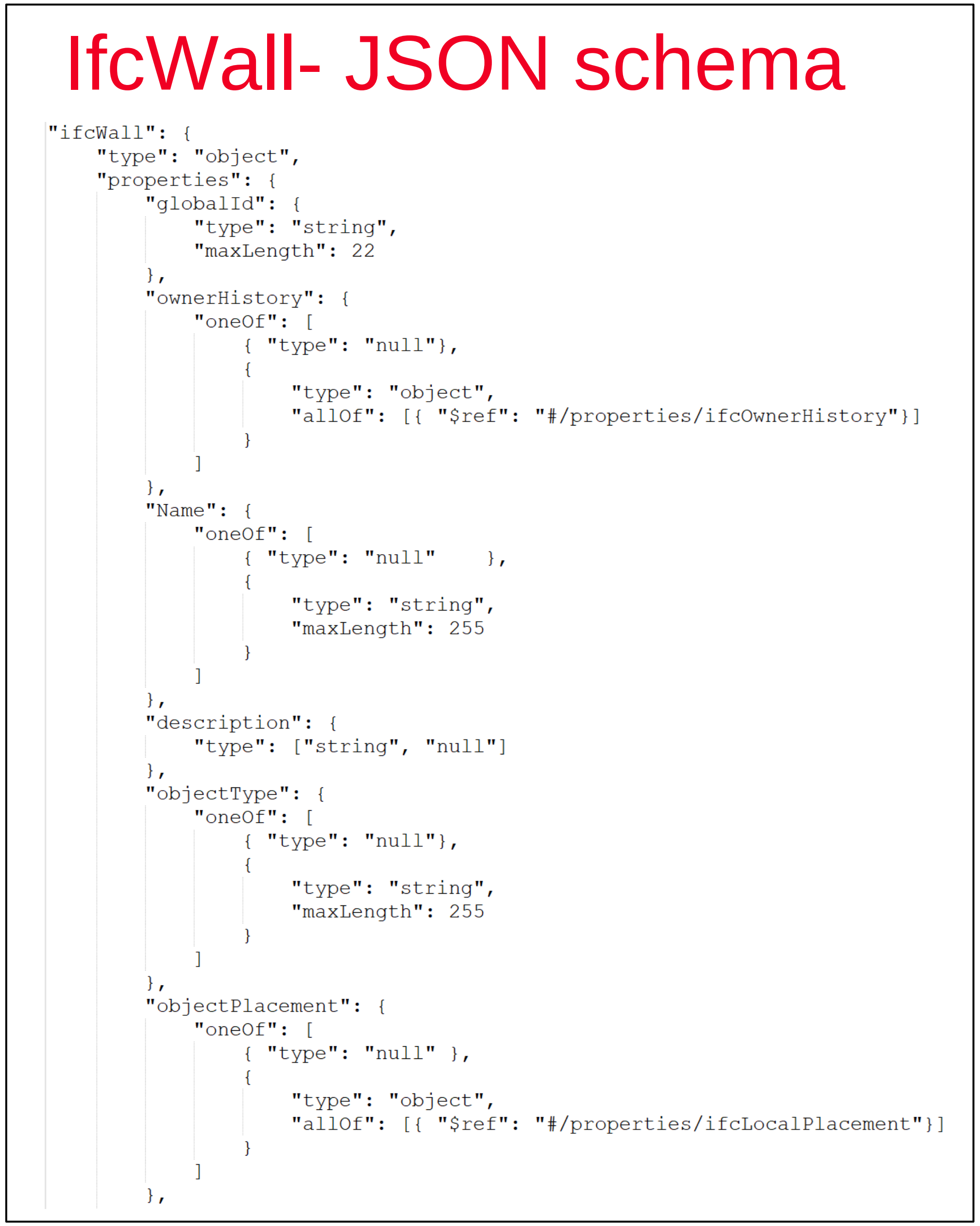
- ❖ Improving consistency
- ❖ Removing redundancies
- ❖ Simplifying IFC complexities

IFC Schema



IfcWall- EXPRESS schema

```
ENTITY IfcWall;
    ENTITY IfcRoot;
        GlobalId                : IfcGloballyUniqueId;
        OwnerHistory            : IfcOwnerHistory;
        Name                    : OPTIONAL IfcLabel;
        Description             : OPTIONAL IfcText;
    ENTITY IfcObjectDefinition;
    INVERSE
        HasAssignments          : SET OF IfcRelAssigns FOR RelatedObjects;
        IsDecomposedBy          : SET OF IfcRelDecomposes FOR RelatingObject;
        Decomposes              : SET [0:1] OF IfcRelDecomposes FOR RelatedObjects;
        HasAssociations         : SET OF IfcRelAssociates FOR RelatedObjects;
    ENTITY IfcObject;
        ObjectType              : OPTIONAL IfcLabel;
    INVERSE
        IsDefinedBy             : SET OF IfcRelDefines FOR RelatedObjects;
    ENTITY IfcProduct;
        ObjectPlacement         : OPTIONAL IfcObjectPlacement;
        Representation          : OPTIONAL IfcProductRepresentation;
    INVERSE
        ReferencedBy            : SET OF IfcRelAssignsToProduct FOR RelatingProduct;
    ENTITY IfcElement;
        Tag                     : OPTIONAL IfcIdentifier;
    INVERSE
        FillsVoids              : SET [0:1] OF IfcRelFillsElement FOR RelatedBuildingElement;
        ConnectedTo             : SET OF IfcRelConnectsElements FOR RelatingElement;
        HasCoverings            : SET OF IfcRelCoversBldgElements FOR RelatingBuildingElement;
        HasProjections          : SET OF IfcRelProjectsElement FOR RelatingElement;
        HasStructuralMember     : SET OF IfcRelConnectsStructuralElement FOR RelatingElement;
        ReferencedInStructures  : SET OF IfcRelReferencedInSpatialStructure FOR RelatedElements;
        HasPorts                : SET OF IfcRelConnectsPortToElement FOR RelatedElement;
        HasOpenings             : SET OF IfcRelVoidsElement FOR RelatingBuildingElement;
        IsConnectionRealization : SET OF IfcRelConnectsWithRealizingElements FOR RealizingElements;
        ProvidesBoundaries      : SET OF IfcRelSpaceBoundary FOR RelatedBuildingElement;
        ConnectedFrom           : SET OF IfcRelConnectsElements FOR RelatedElement;
        ContainedInStructure    : SET [0:1] OF IfcRelContainedInSpatialStructure FOR RelatedElements;
    ENTITY IfcBuildingElement;
    ENTITY IfcWall;
END_ENTITY;
```



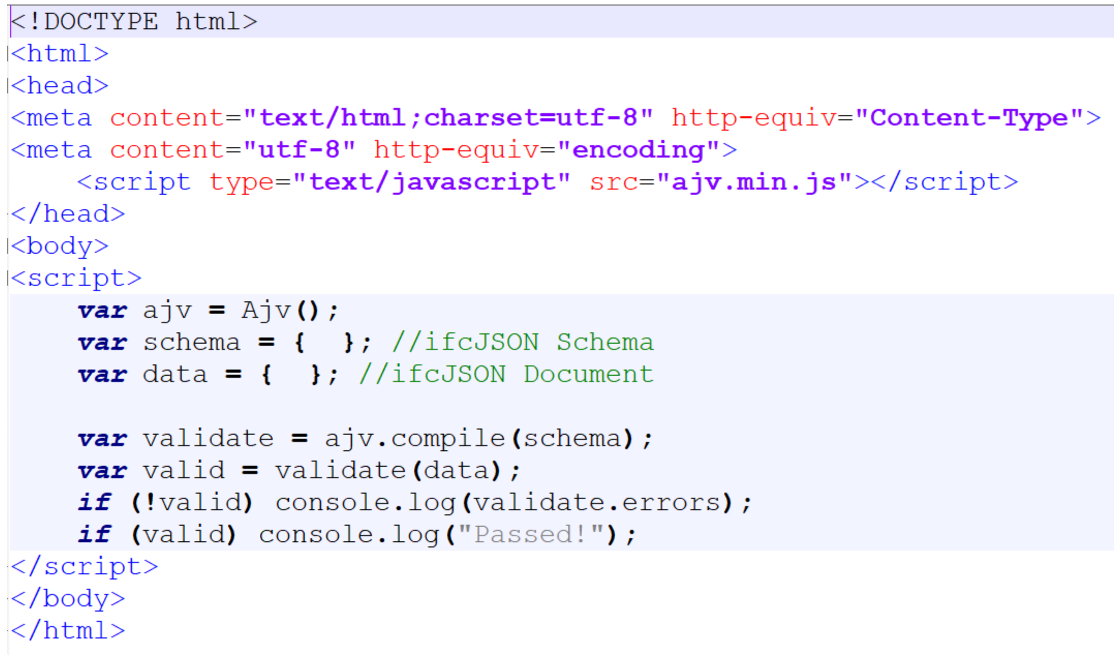
IfcWall- JSON schema

```
"ifcWall": {
    "type": "object",
    "properties": {
        "globalId": {
            "type": "string",
            "maxLength": 22
        },
        "ownerHistory": {
            "oneOf": [
                { "type": "null"},
                {
                    "type": "object",
                    "allOf": [{ "$ref": "#/properties/ifcOwnerHistory"}]
                }
            ]
        },
        "Name": {
            "oneOf": [
                { "type": "null"    },
                {
                    "type": "string",
                    "maxLength": 255
                }
            ]
        },
        "description": {
            "type": ["string", "null"]
        },
        "objectType": {
            "oneOf": [
                { "type": "null"},
                {
                    "type": "string",
                    "maxLength": 255
                }
            ]
        },
        "objectPlacement": {
            "oneOf": [
                { "type": "null" },
                {
                    "type": "object",
                    "allOf": [{ "$ref": "#/properties/ifcLocalPlacement"}]
                }
            ]
        },
```

IfcJSON Validation

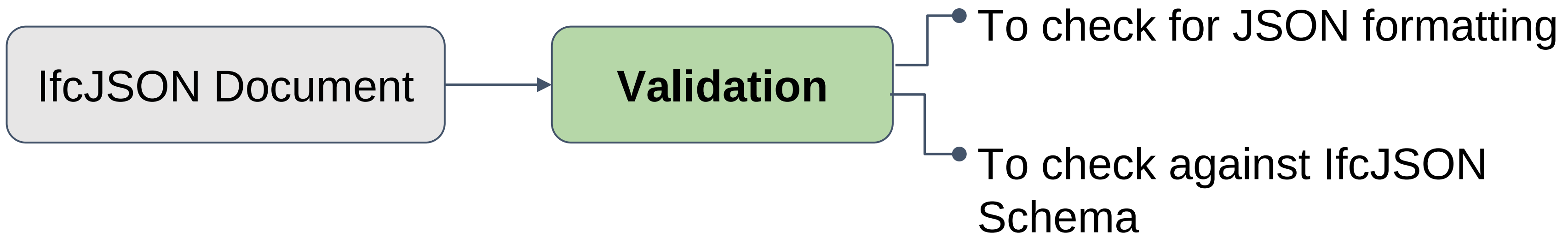
- Validating IfcJSON document
  - against IfcJSON schema
  - for syntactic validation

Example validator →



```
<!DOCTYPE html>
<html>
<head>
<meta content="text/html;charset=utf-8" http-equiv="Content-Type">
<meta content="utf-8" http-equiv="encoding">
    <script type="text/javascript" src="ajv.min.js"></script>
</head>
<body>
<script>
    var ajv = Ajv();
    var schema = {   }; //ifcJSON Schema
    var data = {   }; //ifcJSON Document

    var validate = ajv.compile(schema);
    var valid = validate(data);
    if (!valid) console.log(validate.errors);
    if (valid) console.log("Passed!");
</script>
</body>
</html>
```

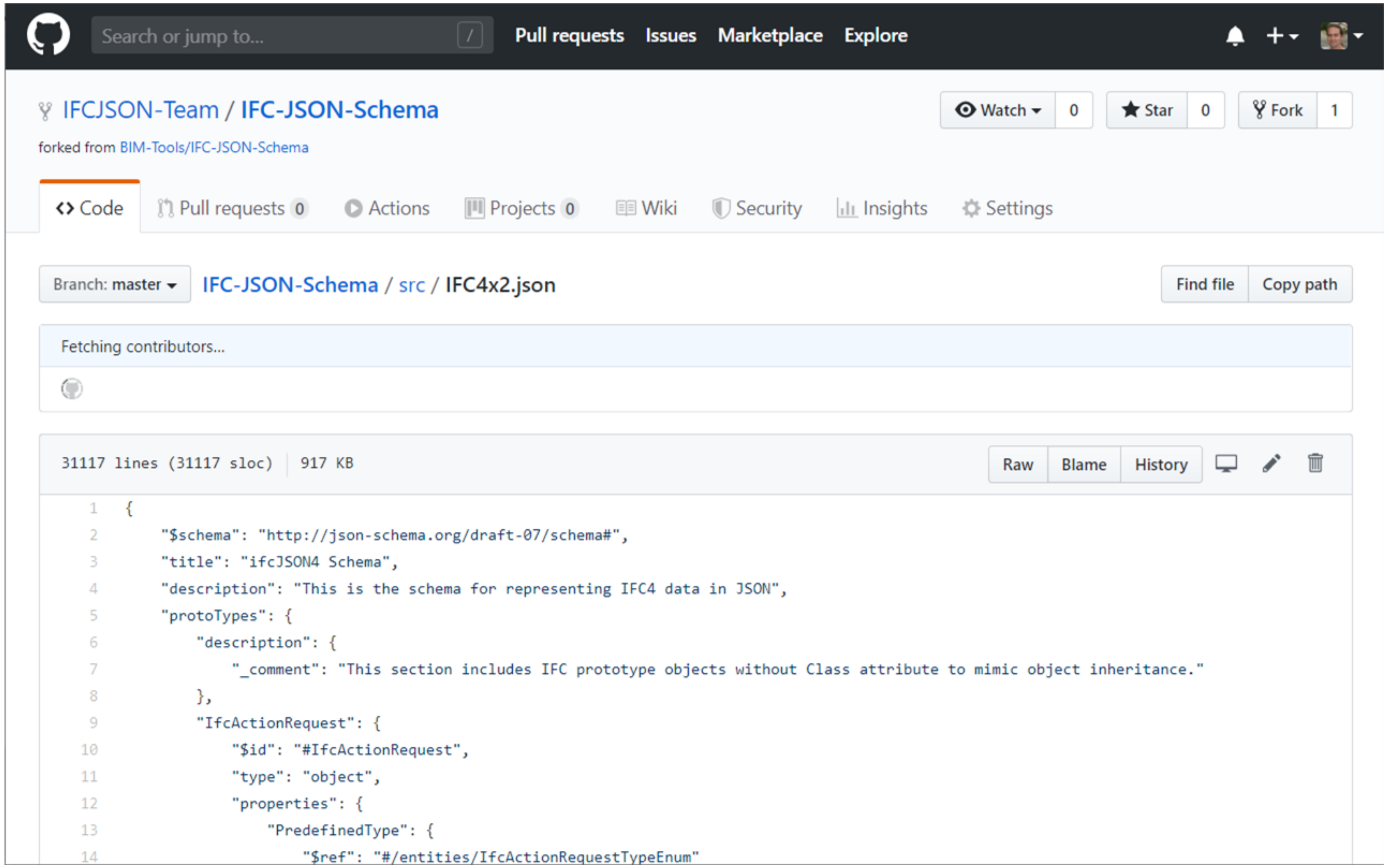


IfcJson tools & resources

everything is on Github [https://github.com/**IFCJSON-Team**](https://github.com/IFCJSON-Team)

IfcJSON

IfcJSON Schema generator



IFCJSON-Team / IFC-JSON-Schema

forked from BIM-Tools/IFC-JSON-Schema

IFC-JSON-Schema / src / IFC4x2.json

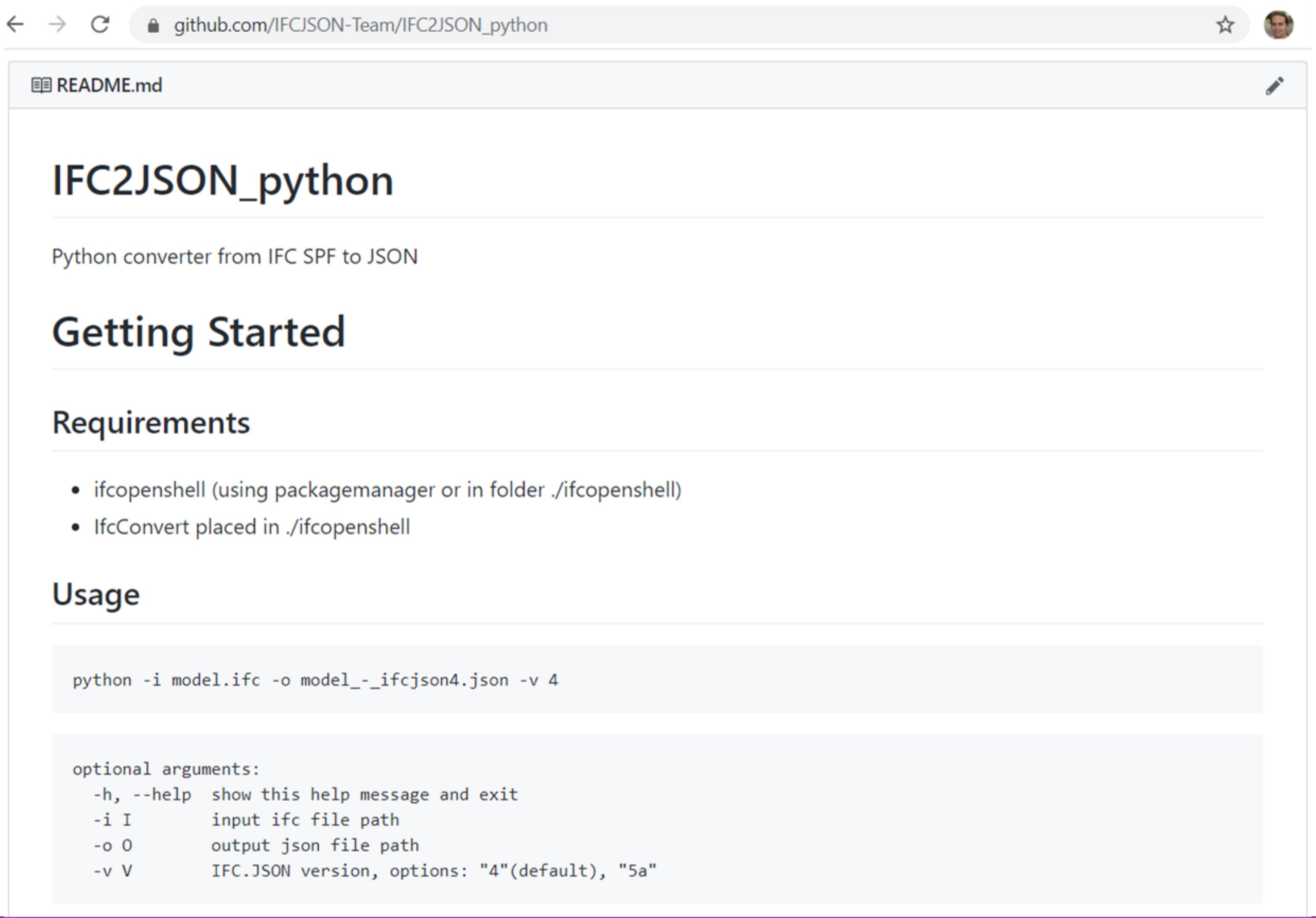
Fetching contributors...

31117 lines (31117 sloc) | 917 KB

```
{
    "$schema": "http://json-schema.org/draft-07/schema#",
    "title": "ifcJSON4 Schema",
    "description": "This is the schema for representing IFC4 data in JSON",
    "protoTypes": {
        "description": {
            "_comment": "This section includes IFC prototype objects without Class attribute to mimic object inheritance."
        },
        "IfcActionRequest": {
            "$id": "#IfcActionRequest",
            "type": "object",
            "properties": {
                "PredefinedType": {
                    "$ref": "#/entities/IfcActionRequestTypeEnum"
```

IfcJSO

Python converter



github.com/IFCJSON-Team/IFC2JSON_python

README.md

IFC2JSON_python

Python converter from IFC SPF to JSON

Getting Started

Requirements

- ifcopenshell (using packagemanager or in folder ./ifcopenshell)
- IfcConvert placed in ./ifcopenshell

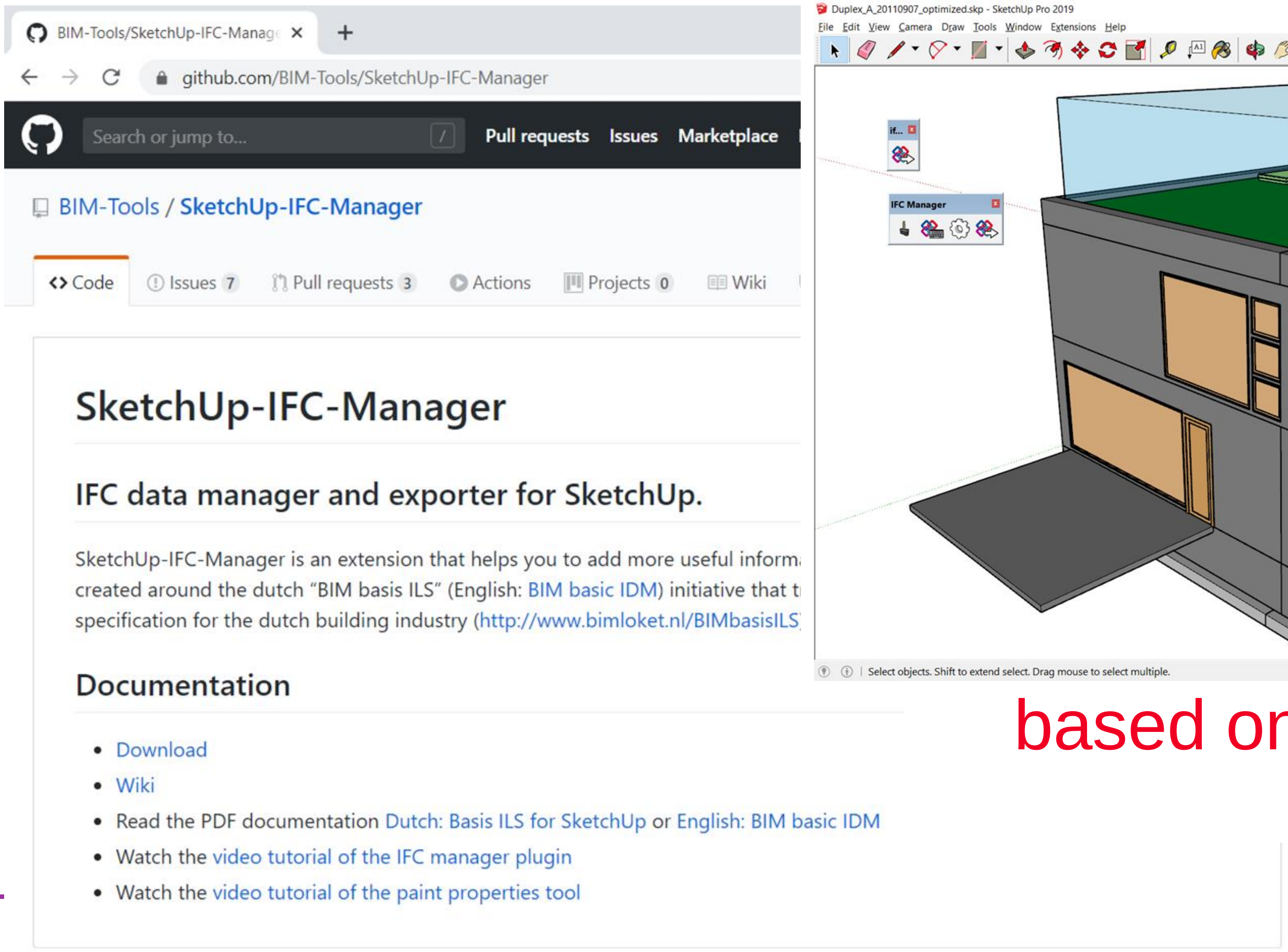
Usage

```
python -i model.ifc -o model_-_ifcjson4.json -v 4
```

```
optional arguments:
  -h, --help  show this help message and exit
  -i I        input ifc file path
  -o O        output json file path
  -v V        IFC.JSON version, options: "4"(default), "5a"
```

IfcJSON

Sketchup exporter



BIM-Tools / SketchUp-IFC-Manager

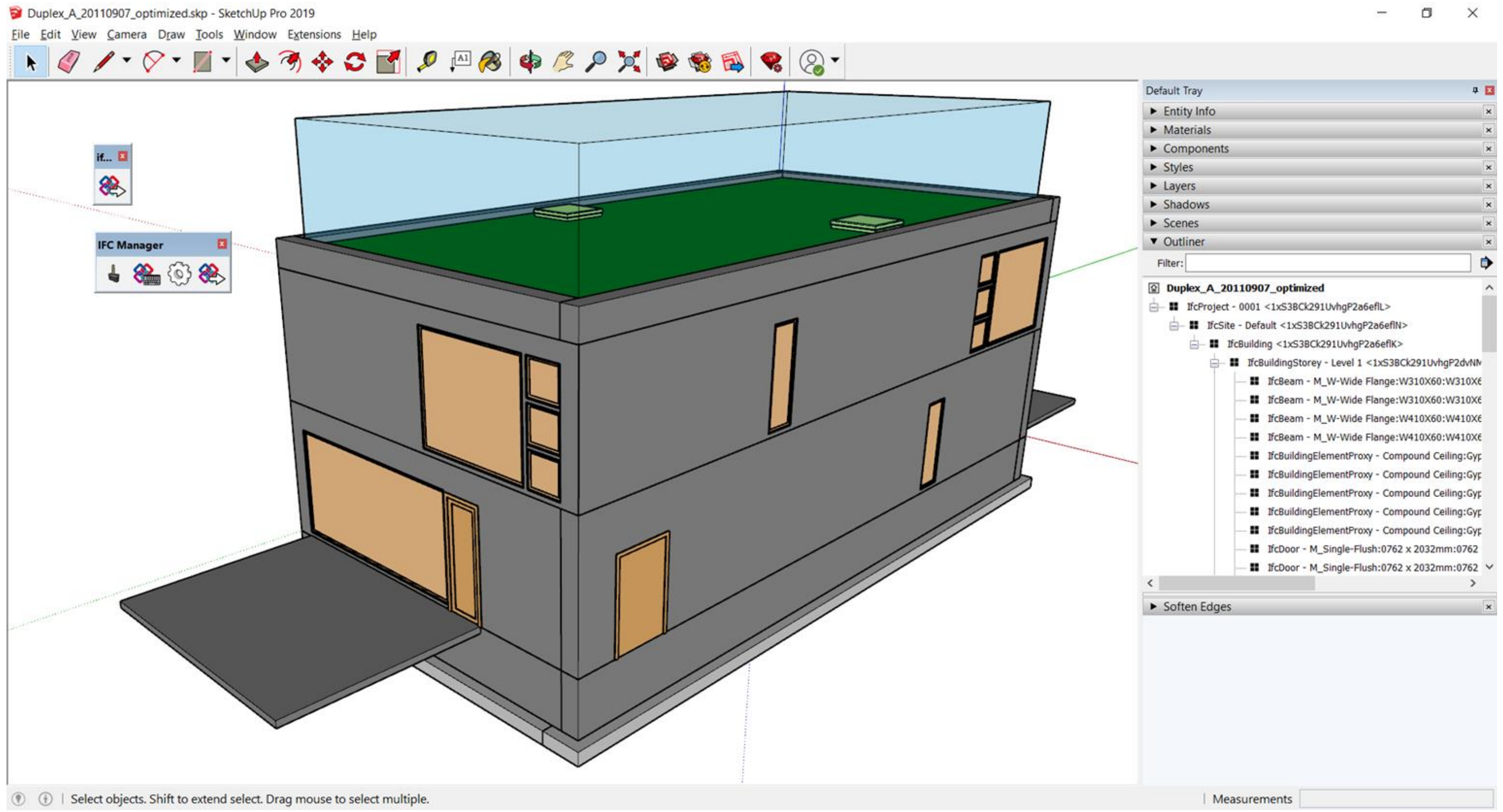
SketchUp-IFC-Manager

IFC data manager and exporter for SketchUp.

SketchUp-IFC-Manager is an extension that helps you to add more useful inform
created around the dutch “BIM basis ILS” (English: BIM basic IDM) initiative that t
specification for the dutch building industry (http://www.bimloket.nl/BIMbasisILS

Documentation

- Download
- Wiki
- Read the PDF documentation Dutch: Basis ILS for SketchUp or English: BIM basic IDM
- Watch the video tutorial of the IFC manager plugin
- Watch the video tutorial of the paint properties tool



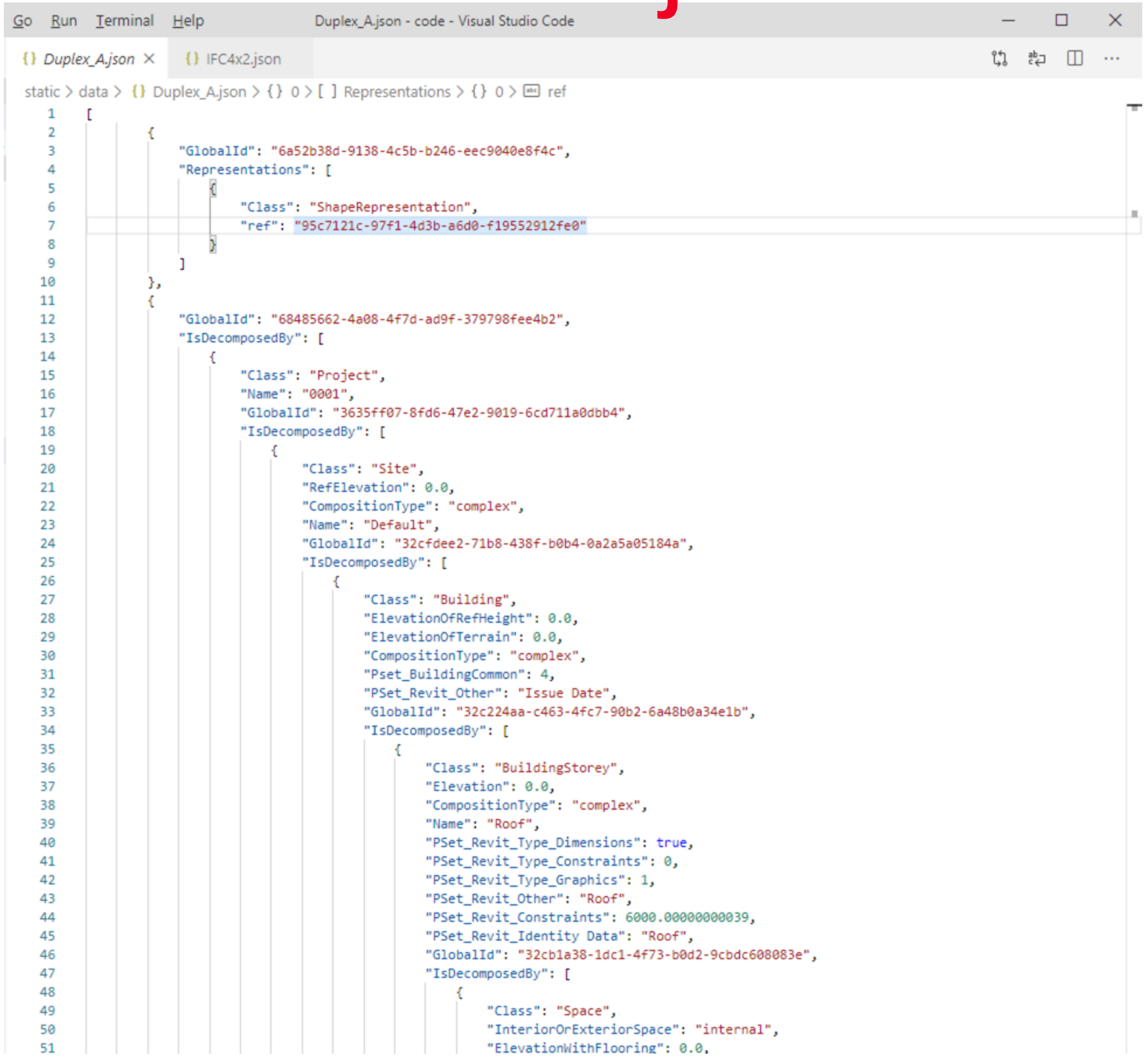
Duplex_A_20110907_optimized
IfcProject - 0001 <1xS3BCk291UvhgP2a6eflL>
IfcSite - Default <1xS3BCk291UvhgP2a6eflN>
IfcBuilding <1xS3BCk291UvhgP2a6eflK>
IfcBuildingStorey - Level 1 <1xS3BCk291UvhgP2dvNM
IfcBeam - M_W-Wide Flange:W310X60:W310X6
IfcBeam - M_W-Wide Flange:W310X60:W310X6
IfcBeam - M_W-Wide Flange:W410X60:W410X6
IfcBeam - M_W-Wide Flange:W410X60:W410X6
IfcBuildingElementProxy - Compound Ceiling:Gyp
IfcBuildingElementProxy - Compound Ceiling:Gyp
IfcBuildingElementProxy - Compound Ceiling:Gyp
IfcBuildingElementProxy - Compound Ceiling:Gyp
IfcBuildingElementProxy - Compound Ceiling:Gyp
IfcDoor - M_Single-Flush:0762 x 2032mm:0762
IfcDoor - M_Single-Flush:0762 x 2032mm:0762

Select objects. Shift to extend select. Drag mouse to select multiple.

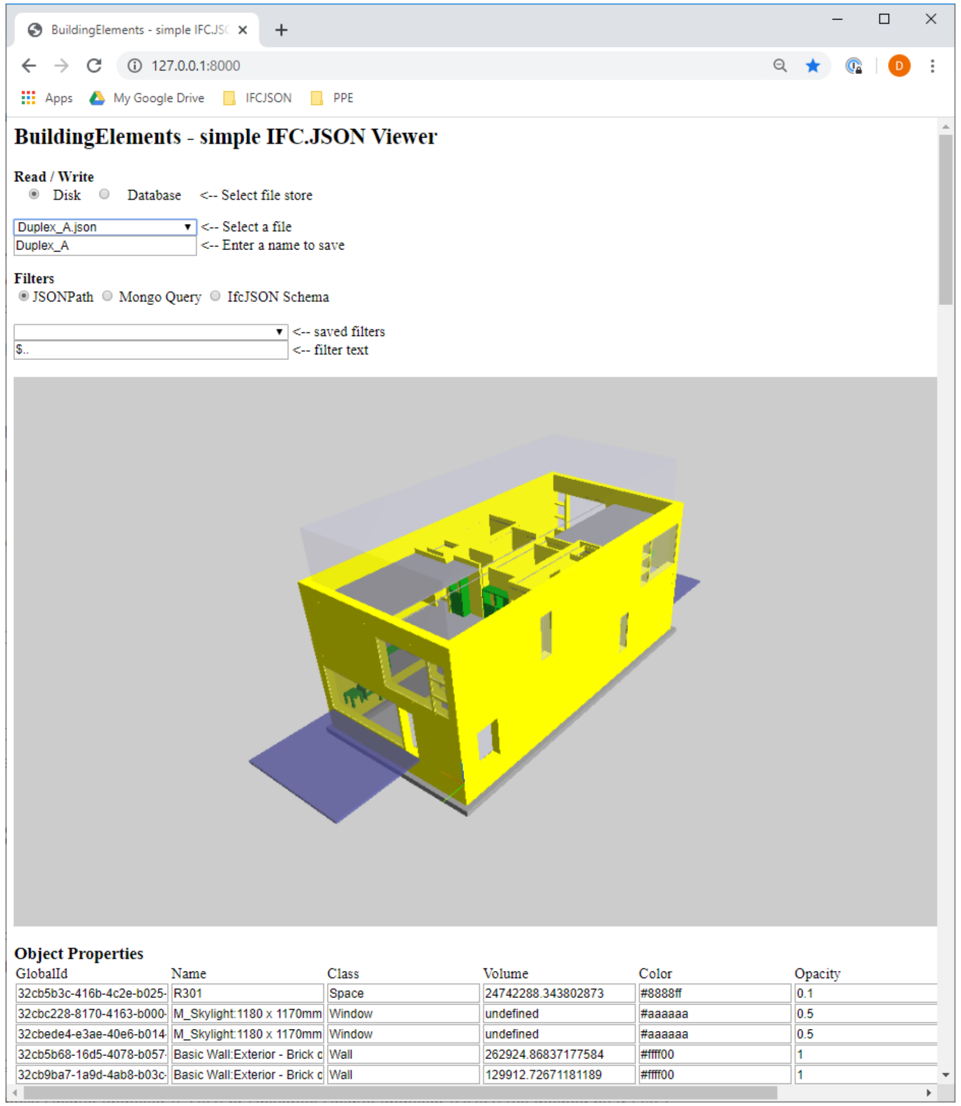
based on experimental version of Sketchup-IFC-Manager

buildingSMART®

IfcJSON node.js web framework


Go Run Terminal Help
Duplex_A.json - code - Visual Studio Code
Duplex_A.json
IFC4x2.json
static > data > Duplex_A.json > {} 0 > [ ] Representations > {} 0 > ref

```
[
    {
        "GlobalId": "6a52b38d-9138-4c5b-b246-eec9040e8f4c",
        "Representations": [
            {
                "Class": "ShapeRepresentation",
                "ref": "95c7121c-97f1-4d3b-a6d0-f19552912fe0"
            }
        ]
    },
    {
        "GlobalId": "68485662-4a08-4f7d-ad9f-379798fee4b2",
        "IsDecomposedBy": [
            {
                "Class": "Project",
                "Name": "0001",
                "GlobalId": "3635ff07-8fd6-47e2-9019-6cd711a0dbb4",
                "IsDecomposedBy": [
                    {
                        "Class": "Site",
                        "RefElevation": 0.0,
                        "CompositionType": "complex",
                        "Name": "Default",
                        "GlobalId": "32cfdee2-71b8-438f-b0b4-0a2a5a05184a",
                        "IsDecomposedBy": [
                            {
                                "Class": "Building",
                                "ElevationOfRefHeight": 0.0,
                                "ElevationOfTerrain": 0.0,
                                "CompositionType": "complex",
                                "Pset_BuildingCommon": 4,
                                "PSet_Revit_Other": "Issue Date",
                                "GlobalId": "32c224aa-c463-4fc7-90b2-6a48b0a34e1b",
                                "IsDecomposedBy": [
                                    {
                                        "Class": "BuildingStorey",
                                        "Elevation": 0.0,
                                        "CompositionType": "complex",
                                        "Name": "Roof",
                                        "PSet_Revit_Type_Dimensions": true,
                                        "PSet_Revit_Type_Constraints": 0,
                                        "PSet_Revit_Type_Graphics": 1,
                                        "PSet_Revit_Other": "Roof",
                                        "PSet_Revit_Constraints": 6000.00000000039,
                                        "PSet_Revit_Identity Data": "Roof",
                                        "GlobalId": "32cb1a38-1dc1-4f73-b0d2-9cbdc608083e",
                                        "IsDecomposedBy": [
                                            {
                                                "Class": "Space",
                                                "InteriorOrExteriorSpace": "internal",
                                                "ElevationWithFlooring": 0.0,
```



BuildingElements - simple IFC.JSON Viewer

Read / Write
[✓ Disk] Database <-- Select file store

Duplex_A.json <-- Select a file
Duplex_A <-- Enter a name to save

Filters
[✓ JSONPath] Mongo Query IfcJSON Schema

<-- saved filters
$.. <-- filter text

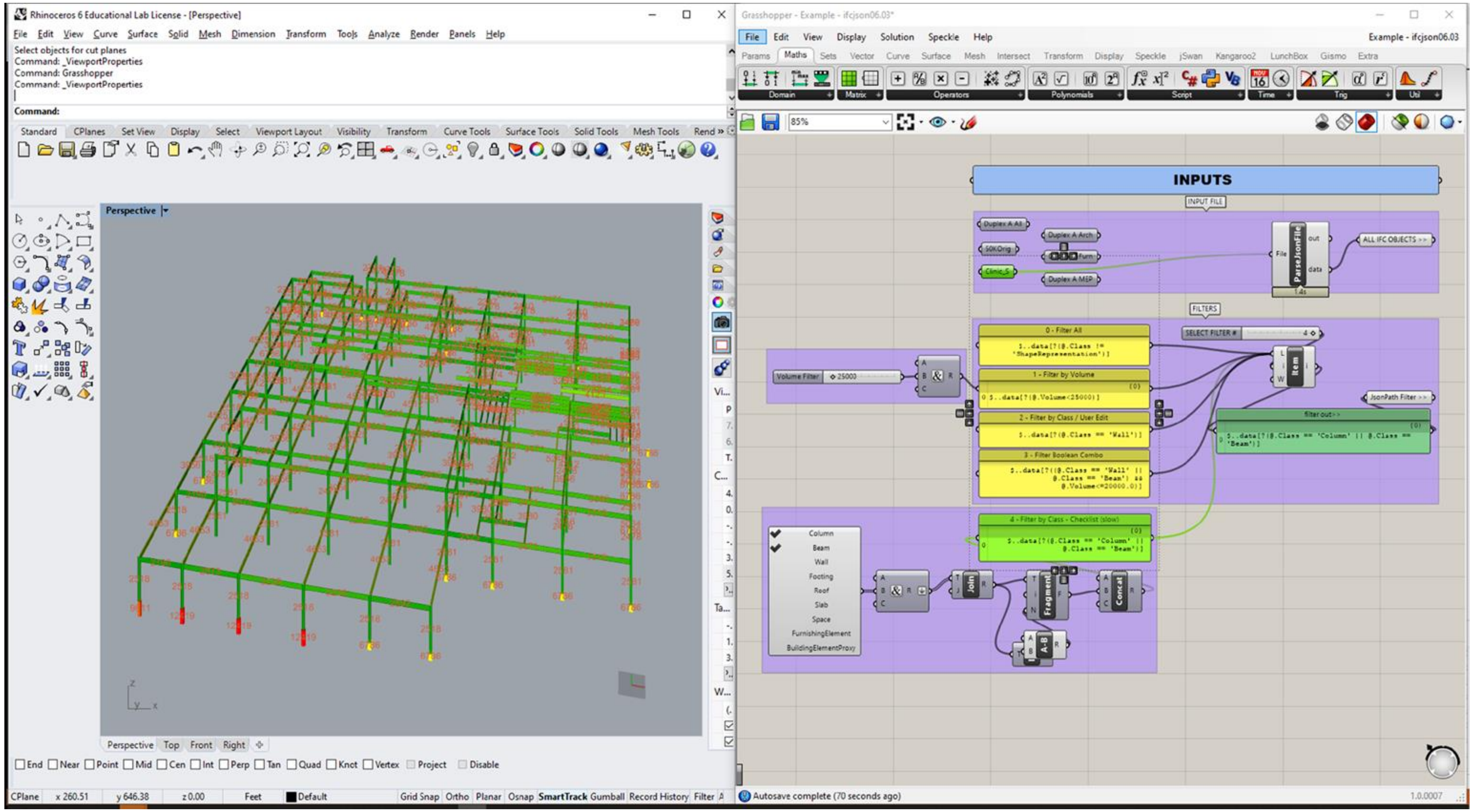
Object Properties

GlobalId	Name	Class	Volume	Color	Opacity
32cb5b3c-416b-4c2e-b025-	R301	Space	24742288.343802873	#8888ff	0.1
32cbc228-8170-4163-b000-	M_Skylight:1180 x 1170mm	Window	undefined	#aaaaaa	0.5
32cbede4-e3ae-40e6-b014-	M_Skylight:1180 x 1170mm	Window	undefined	#aaaaaa	0.5
32cb5b68-16d5-4078-b057-	Basic Wall:Exterior - Brick c	Wall	262924.86837177584	#ffff00	1
32cb9ba7-1a9d-4ab8-b03c-	Basic Wall:Exterior - Brick c	Wall	129912.72671181189	#ffff00	1

buildingSMART®

IfcJSON

Grasshopper importer


Rhinoceros 6 Educational Lab License - [Perspective]
File Edit View Curve Surface Solid Mesh Dimension Transform Tools Analyze Render Panels Help
Standard CPlanes Set View Display Select Viewport Layout Visibility Transform Curve Tools Surface Tools Solid Tools Mesh Tools Rend
Perspective
Perspective Top Front Right
End Near Point Mid Cen Int Perp Tan Quad Knot Vertex Project Disable
CPlane x 260.51 y 646.38 z 0.00 Feet Default Grid Snap Ortho Planar Osnap SmartTrack Gumball Record History Filter
Grasshopper - Example - ifcjson06.03*
File Edit View Display Solution Speckle Help
Example - ifcjson06.03
Params Maths Sets Vector Curve Surface Mesh Intersect Transform Display Speckle jSwan Kangaroo2 LunchBox Gismo Extra
Domain Matrix Operators Polynomials Script Time Trig Util
85%
INPUTS
INPUT FILE
Duplex A All
50KOrig
Clinic_S
Duplex A Arch
Furn
Duplex A MEP
File
ParseJsonFile
out
data
1.4s
ALL IFC OBJECTS >>
FILTERS
SELECT FILTER # 4
Volume Filter 25000
Item
JsonPath Filter >>
filter out>>
Column
Beam
Wall
Footing
Roof
Slab
Space
FurnishingElement
BuildingElementProxy
Join
Fragment
Concat
Autosave complete (70 seconds ago)
1.0.0007

```
Select objects for cut planes
Command: _ViewportProperties
Command: Grasshopper
Command: _ViewportProperties
Command:
```

0 - Filter All
```
$..data[?(@.Class != 'ShapeRepresentation')]
```

1 - Filter by Volume
```
$..data[?(@.Volume<25000)]
```

2 - Filter by Class / User Edit
```
$..data[?(@.Class == 'Wall')]
```

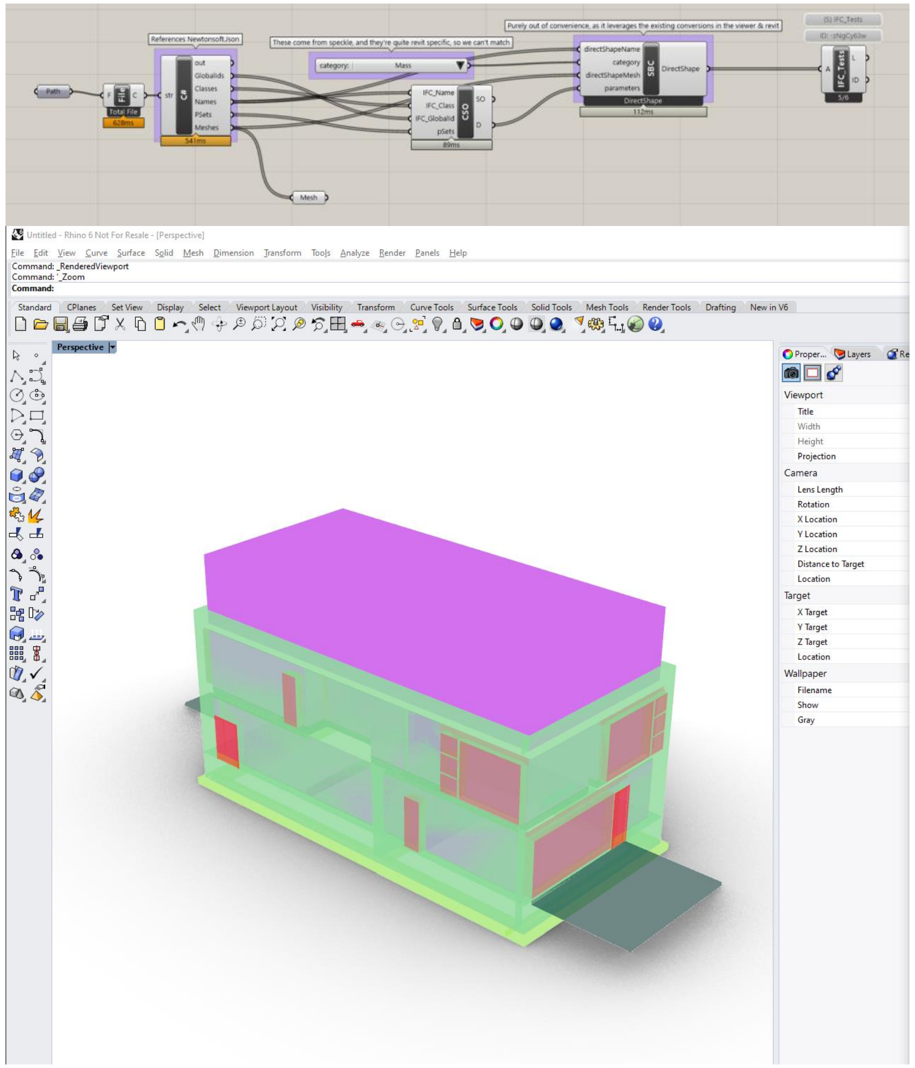
3 - Filter Boolean Combo
```
$..data[?((@.Class == 'Wall' ||
@.Class == 'Beam') &&
@.Volume<=20000.0)]
```

4 - Filter by Class - Checklist (slow)
```
$..data[?(@.Class == 'Column' ||
@.Class == 'Beam')]
```

filter out>>
```
$..data[?(@.Class == 'Column' || @.Class ==
'Beam')]
```

buildingSMART®

IfcJSON Speckle integration

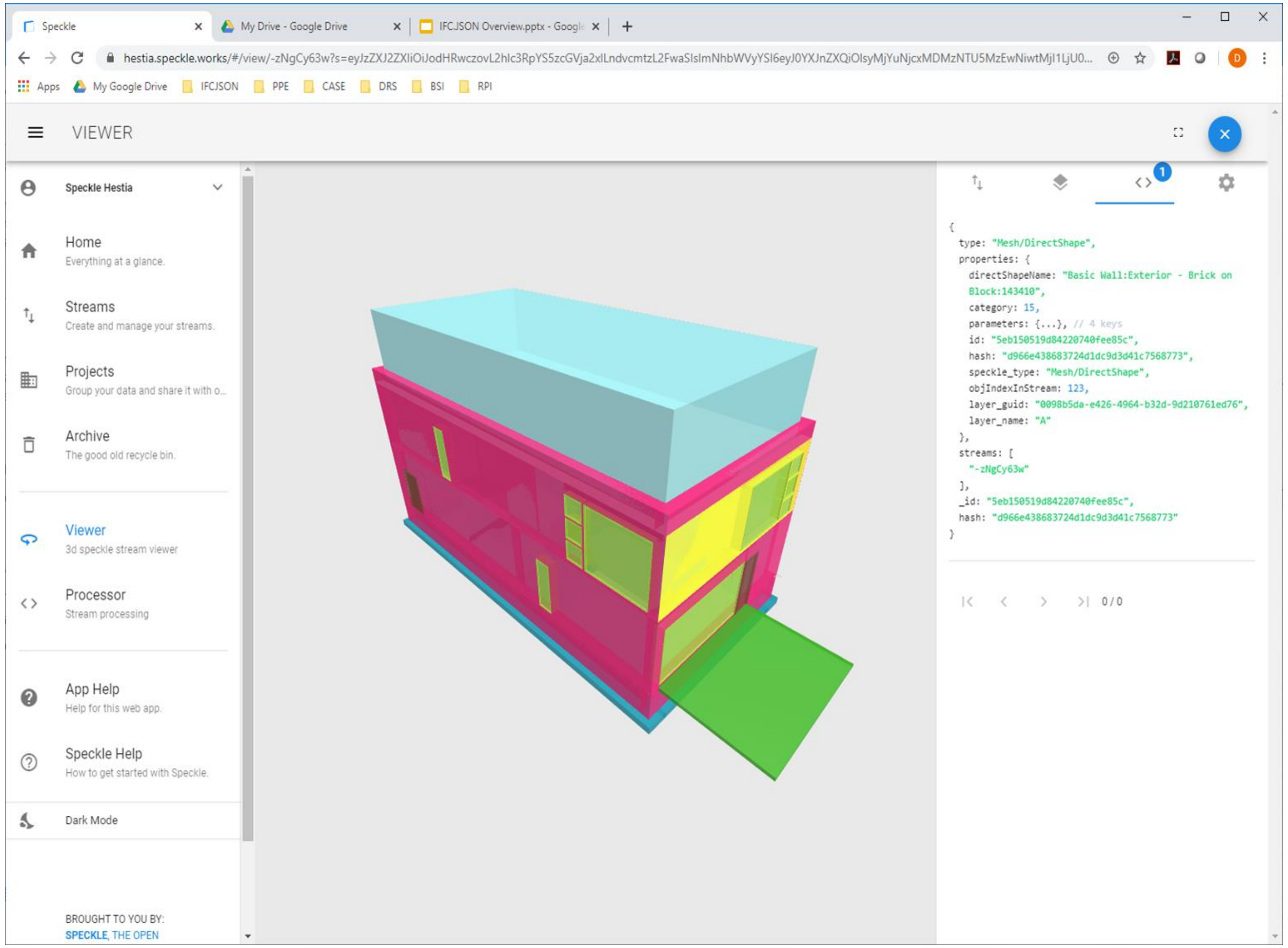


References NewtonsoftJson

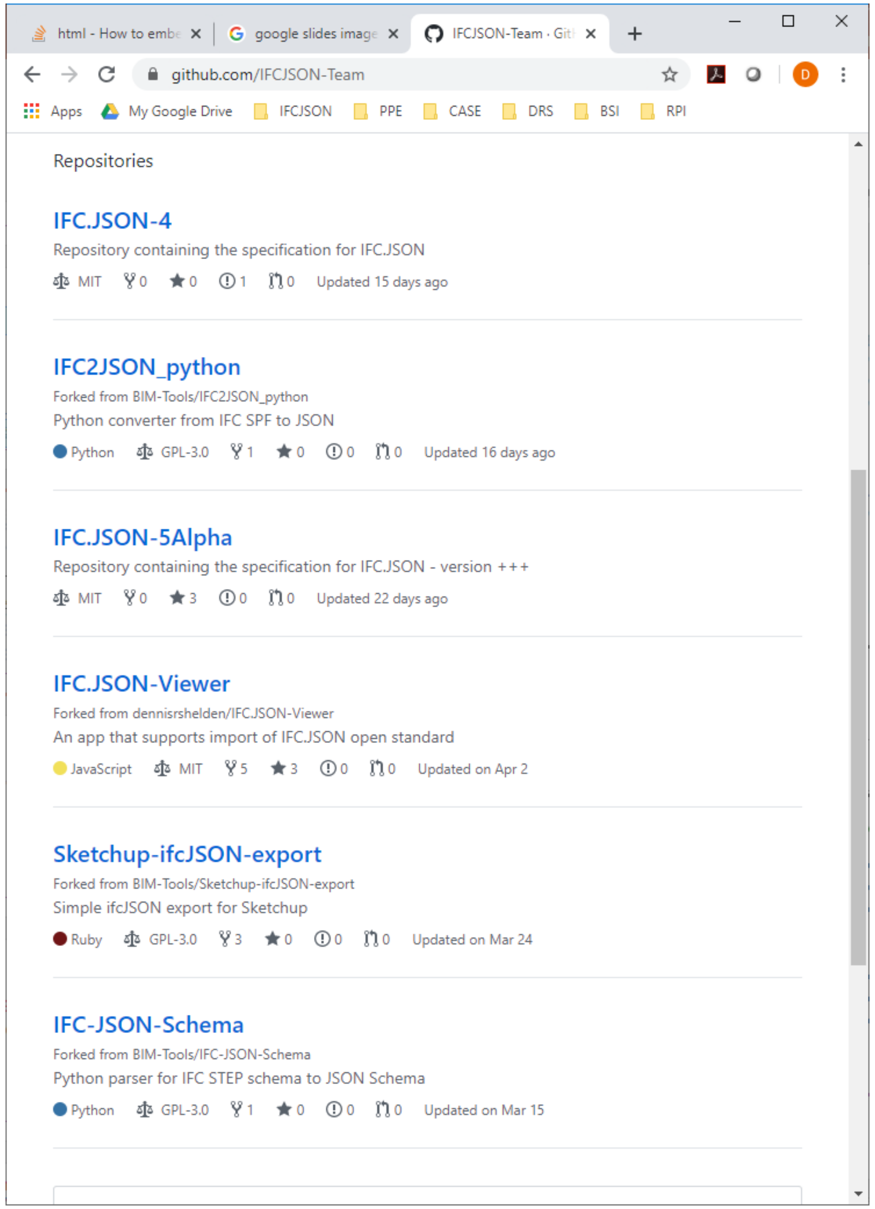
These come from speckle, and they're quite revit specific, so we can't match

Purely out of convenience, as it leverages the existing conversions in the viewer & revit

```
Untitled - Rhino 6 Not For Resale - [Perspective]
Command: _RenderedViewport
Command: '_Zoom
Command:
```


Speckle
My Drive - Google Drive
IFCJSON Overview.pptx - Google
hestia.speckle.works/#/view/-zNgCy63w?s=eyJzZXJ2ZXIiOiJodHRwczovL2hlc3RpYS5zcGVja2xlLndvcmtzL2FwaSIsImNhbWVyYSI6eyJ0YXJnZXQiOlsyMjYuNjcxMDMzNTU5MzEwNiwtMjI1LjU0...
Apps
My Google Drive
IFCJSON
PPE
CASE
DRS
BSI
RPI
VIEWER
Speckle Hestia
Home
Everything at a glance.
Streams
Create and manage your streams.
Projects
Group your data and share it with o...
Archive
The good old recycle bin.
Viewer
3d speckle stream viewer
Processor
Stream processing
App Help
Help for this web app.
Speckle Help
How to get started with Speckle.
Dark Mode
BROUGHT TO YOU BY:
SPECKLE, THE OPEN
0 / 0

```
{
  type: "Mesh/DirectShape",
  properties: {
    directShapeName: "Basic Wall:Exterior - Brick on
    Block:143410",
    category: 15,
    parameters: {...}, // 4 keys
    id: "5eb150519d84220740fee85c",
    hash: "d966e438683724d1dc9d3d41c7568773",
    speckle_type: "Mesh/DirectShape",
    objIndexInStream: 123,
    layer_guid: "0098b5da-e426-4964-b32d-9d210761ed76",
    layer_name: "A"
  },
  streams: [
    "-zNgCy63w"
  ],
  _id: "5eb150519d84220740fee85c",
  hash: "d966e438683724d1dc9d3d41c7568773"
}
```



Repositories

IFC.JSON-4
Repository containing the specification for IFC.JSON
MIT 0 0 1 0 Updated 15 days ago

IFC2JSON_python
Forked from BIM-Tools/IFC2JSON_python
Python converter from IFC SPF to JSON
Python GPL-3.0 1 0 0 0 Updated 16 days ago

IFC.JSON-5Alpha
Repository containing the specification for IFC.JSON - version +++
MIT 0 3 0 0 Updated 22 days ago

IFC.JSON-Viewer
Forked from dennisrshelden/IFC.JSON-Viewer
An app that supports import of IFC.JSON open standard
JavaScript MIT 5 3 0 0 Updated on Apr 2

Sketchup-ifcJSON-export
Forked from BIM-Tools/Sketchup-ifcJSON-export
Simple ifcJSON export for Sketchup
Ruby GPL-3.0 3 0 0 0 Updated on Mar 24

IFC-JSON-Schema
Forked from BIM-Tools/IFC-JSON-Schema
Python parser for IFC STEP schema to JSON Schema
Python GPL-3.0 1 0 0 0 Updated on Mar 15

docs, demos and utilities

On Github https://github.com/**IFCJSON-Team**:

- Documentation
- Sketchup IfcJSON exporter
- IfcJSON Schema and schema generator
- Node.js & MongoDB web server & viewer
- Grasshopper importer
- Speckle integration

Thanks

Open house invite & more info

email: sheldd@rpi.edu
https://github.com/**IFCJSON-Team**