

---

# 3MF Boolean Operations Extension

---

## Specification & Reference Guide

**Version**    **0.8.0**

**Status**     Draft

## Table of Contents

- [Preface](#)
  - [About this Specification](#)
  - [Document Conventions](#)
  - [Language Notes](#)
  - [Software Conformance](#)
- [Part I: 3MF Documents](#)
  - [Chapter 1. Overview of Additions](#)
  - [Chapter 2. Booleans mesh](#)
    - [2.1. Booleans](#)
- [Part II. Appendices](#)
  - [Appendix A. Glossary](#)
  - [Appendix B. 3MF XSD Schema](#)
  - [Appendix C. Standard Namespace](#)
  - [Appendix D: Example file](#)
- [References](#)

## Preface

---

### About this Specification

This 3MF Boolean Operations Extension is an extension to the core 3MF specification. This document cannot stand alone and only applies as an addendum to the core 3MF specification. Usage of this and any other 3MF extensions follow an a la carte model, defined in the core 3MF specification.

Part I, "3MF Documents," presents the details of the primarily XML-based 3MF Document format. This section describes the XML markup that defines the composition of 3D documents and the appearance of each model within the document.

Part II, "Appendices," contains additional technical details and schemas too extensive to include in the main body of the text as well as convenient reference information.

The information contained in this specification is subject to change. Every effort has been made to ensure its accuracy at the time of publication.

This extension **MUST** be used only with Core specification 1.x.

## Document Conventions

See [the 3MF Core Specification conventions](#).

In this extension specification, as an example, the prefix "bo" maps to the xml-namespace "http://schemas.microsoft.com/3dmanufacturing/booleanoperations/2022/06". See [Appendix C. Standard Namespace](#).

## Language Notes

See [the 3MF Core Specification language notes](#).

## Software Conformance

See [the 3MF Core Specification software conformance](#).

# Part I: 3MF Documents

---

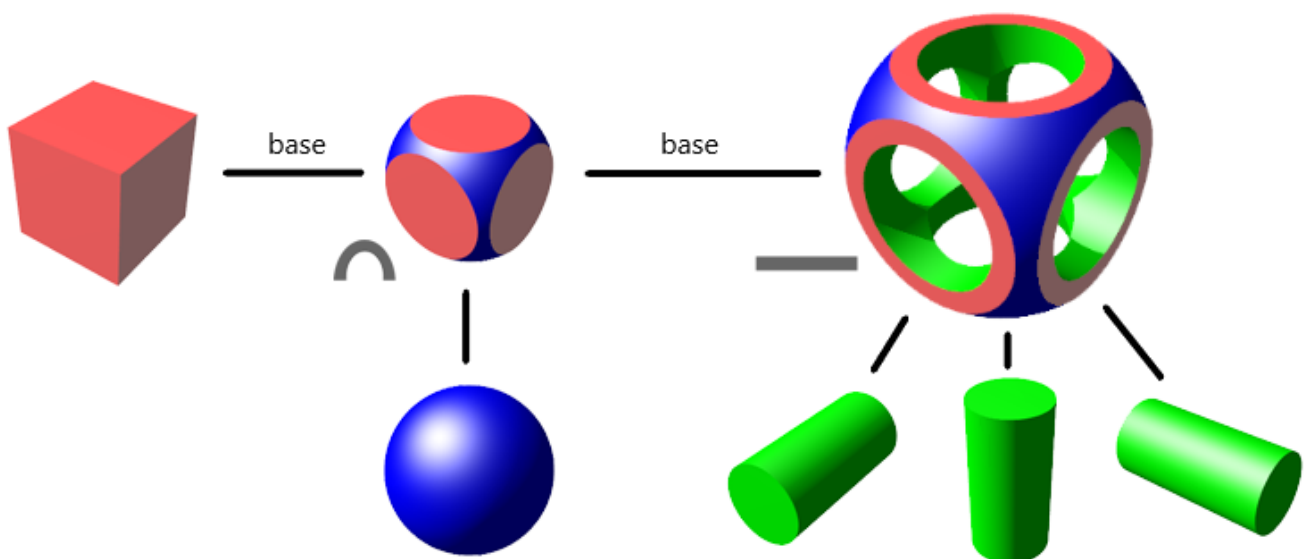
## Chapter 1. Overview of Additions

---

The 3MF Core Specification defines the <components> element in the <object> resource as definition of a tree of different objects to form an assembly, with the intent to allow the reuse of model definitions for an efficient encoding. The resultant shape of a <components> element is the aggregation of each <component> object element.

This extension is based in a simplified Constructive Solid Geometry (CSG).

However, to limit complexity in the consumer, this spec reduces the scope to a sequence of boolean operations.



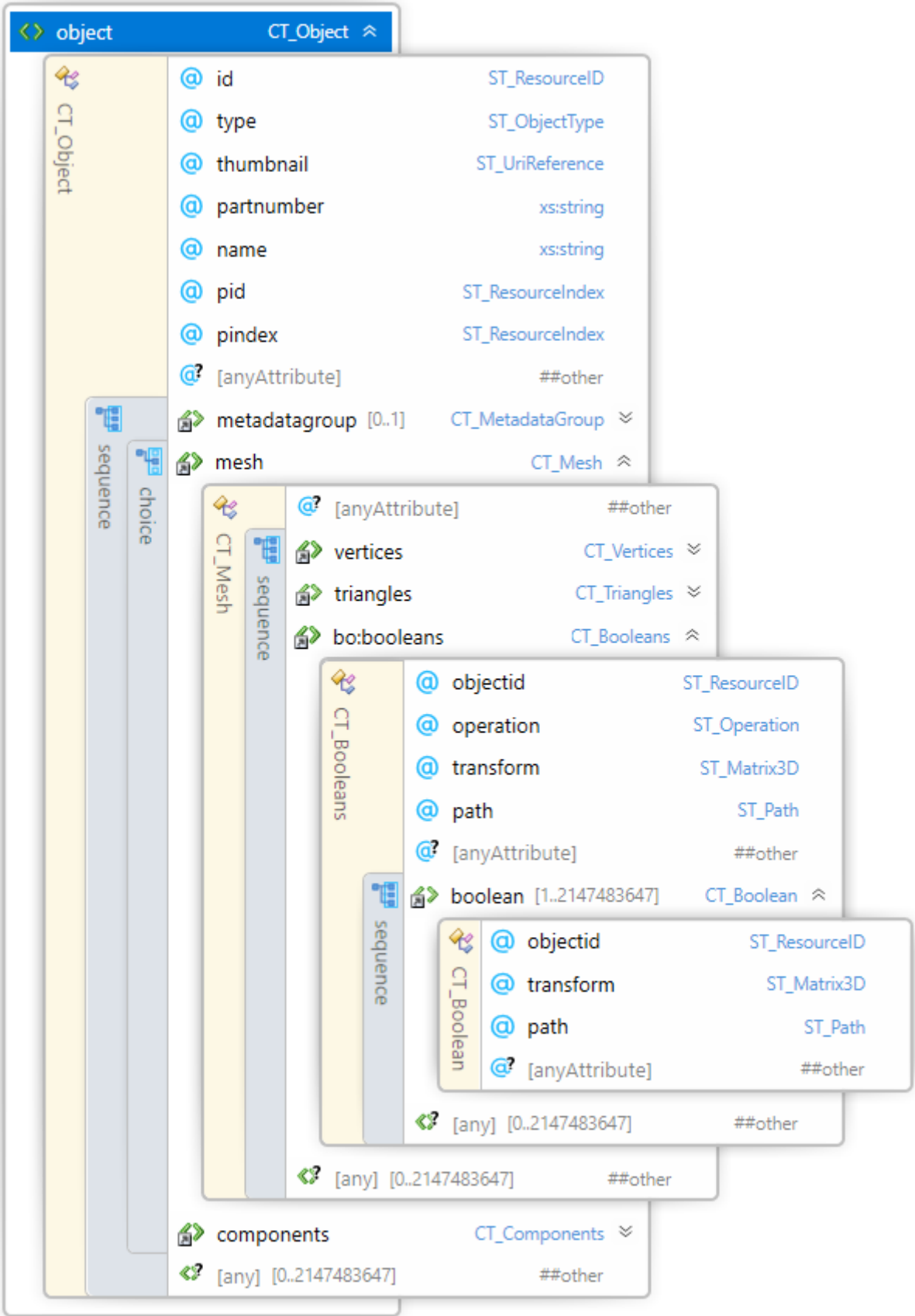
This document describes a new element <booleans> in the <mesh> elements that specifies a new mesh type, other than triangle mesh. This element is OPTIONAL for producers but MUST be supported by consumers that

specify support for the 3MF Boolean Operations Extension.

The <booleans> element defines a new mesh type conforming a sequence of boolean operations to referenced meshes.

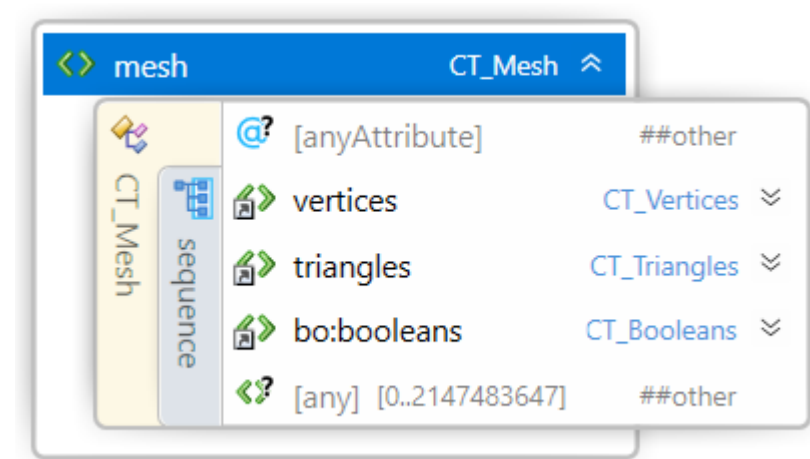
This is a non-backwards compatible change since it declares a different content of the mesh element, with empty vertices and triangles. Therefore, a 3MF package which uses "booleans" meshes MUST enlist the 3MF Boolean Operations Extension as "required extension", as defined in the core specification.

**Figure 1-1: Overview of 3MF Boolean Operations Extension XML structure**



## Chapter 2. Booleans mesh

Element <mesh>

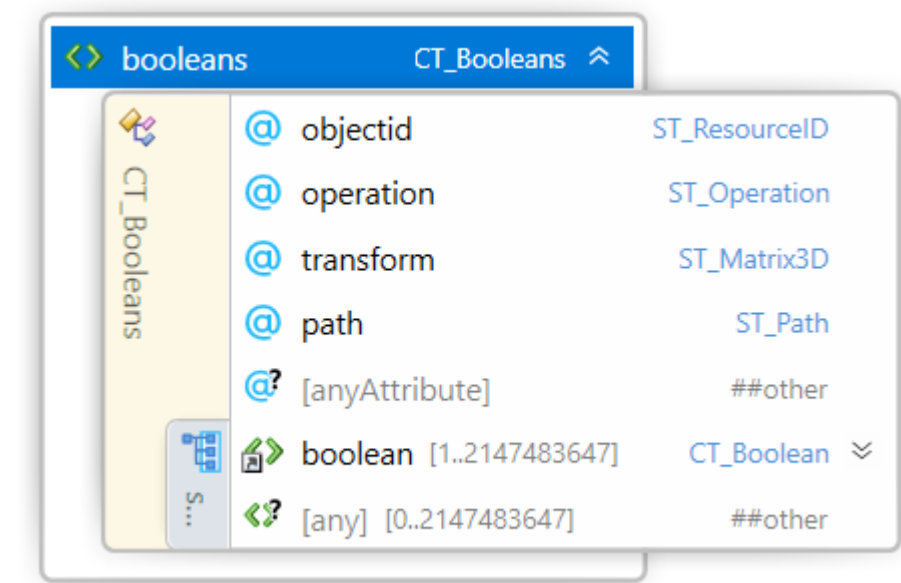


The <mesh> element, defined in [the 3MF Core Specification meshes](#), is enhanced with an optional <booleans> element, declaring that the shape of the object is exclusively defined by a "boolean operation" instead of a triangle mesh or other ways of specifying a shape as defined in other 3MF extensions.

**Note:** this requires that the <vertices> and the <triangle> elements are empty, overriding the core spec definition.

2.1. Booleans

Element <booleans>



| Name      | Type                 | Use      | Default | Annotation  |
|-----------|----------------------|----------|---------|---|
| objectid  | <b>ST_ResourceID</b> | required |         | It references the base object id to apply the boolean operation.                                |
| operation | <b>ST_Operation</b>  |          | union   | The boolean operation: union, difference and intersection.                                      |
| transform | <b>ST_Matrix3D</b>   |          |         | A matrix transform (see <a href="#">3.3. 3D Matrices</a> ) applied to the item to be outputted. |

| Name | Type           | Use | Default | Annotation   |
|------|----------------|-----|---------|--|
| path | <b>ST_Path</b> |     |         | A file path to the base object file being referenced. The path is an absolute path from the root of the 3MF container. |

@anyAttribute

The optional <booleans> element contains one or more <boolean> elements to perform the boolean operation to the referenced object.

**objectid** - Selects the base object to apply the boolean operation. The object MUST be a mesh or another booleans object of type "model". It MUST NOT reference a components object.

**operation** - The boolean operation to perform. The options for the boolean operations are the following:

1. *union*. The resulting object shape is defined as the merger of the shapes. The resulting object surface property is defined by the property of the surface property defining the outer surface, as defined by [the 3MF Core Specification overlapping order](#)

$$\text{union}(\text{base}, a, b, c) = \text{base} \cup (a \cup b \cup c) = ((\text{base} \cup a) \cup b) \cup c$$

2. *difference*. The resulting object shape is defined by the shape in the base object shape that is not in any other object shape. The resulting object surface property, where overlaps, is defined by the object surface property of the subtracting object(s), or no-property is the subtracting object has no property defined in that surface.

$$\text{difference}(\text{base}, a, b, c) = \text{base} - (a \cup b \cup c) = ((\text{base} - a) - b) - c$$

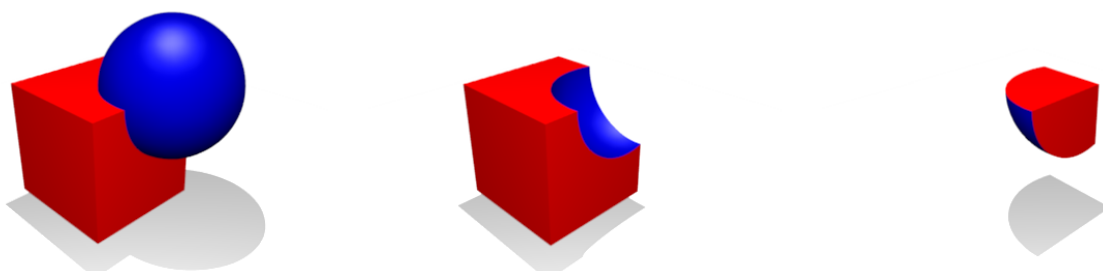
3. *intersection*. The resulting object shape is defined as the common (clipping) shape in all objects. The resulting object surface property is defined as the object surface property of the object defining the new surface, or no-property when the subtracting object has no property defined in that surface.

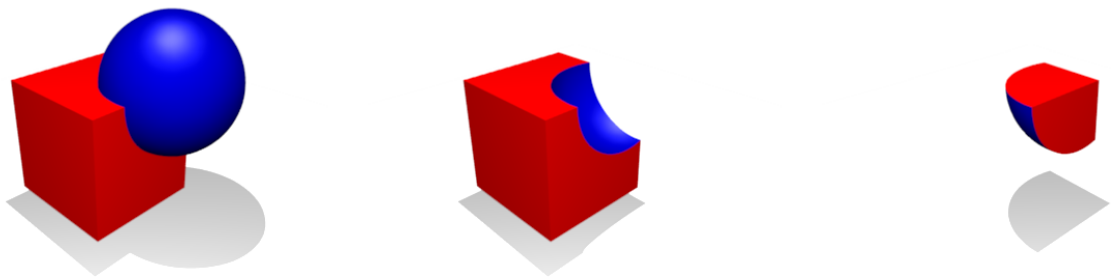
$$\text{intersection}(\text{base}, a, b, c) = \text{base} \cap (a \cup b \cup c) = ((\text{base} \cap a) \cap b) \cap c$$

**transform** - The transform to apply to the selected base object.

**path** - When used in conjunction with [the 3MF Production extension](#), the "path" attribute references objects in non-root model files. Path is an absolute path to the target model file inside the 3MF container that contains the target object. The use of the path attribute in a <booleans> element is ONLY valid in the root model file.

The following diagrams, from the **CSG** Wikipedia, show the three operations:





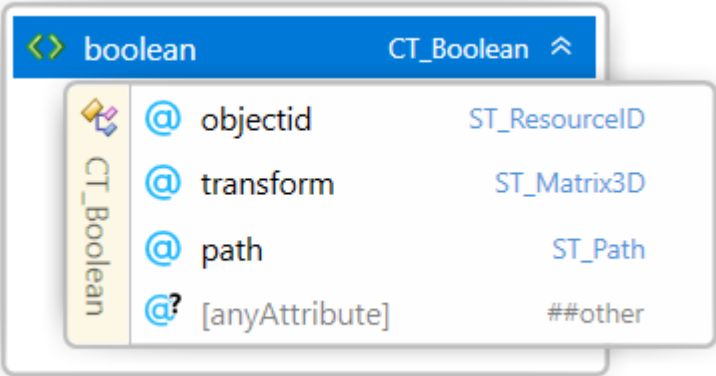
**union:** Merger of two objects into one      **difference:** Subtraction of object from another one      **intersection:** Portion common to objects

**Note:** The subtracting (difference or intersection) objects MUST **only** contain references to meshes containing surfaces defined by triangle meshes. They MUST NOT contain surfaces defined by any other 3MF extension.

Similarly as defined in [the 3MF Core Specification object resources](#), consumers MUST ignore the object type of objects containing a <boolean> element, since the type is always overridden by descendant objects. Producers MUST NOT assign pid or pindex attributes to objects that contain booleans. This ensures that an object with no material will not be split into two representations with different materials due to being referenced as a boolean in multiple objects.

2.1.1. Boolean

Element <boolean>



| Name      | Type          | Use      | Default | Annotation   |
|-----------|---------------|----------|---------|--|
| objectid  | ST_ResourceID | required |         | It references the mesh object id performing the boolean operation.   |
| transform | ST_Matrix3D   |          |         | A matrix transform (see <a href="#">3.3. 3D Matrices</a> ) applied to the item to be outputted.                  |
| path      | ST_Path       |          |         | A file path to the model file being referenced. The path is an absolute path from the root of the 3MF container. |

@anyAttribute

The <boolean> element selects a pre-defined object resource to perform a boolean operation to the base object referenced in the enclosing <booleans> element.

**objectid** - Selects the object with the mesh to apply the boolean operation. The object MUST be a mesh or another booleans object of type "model". It MUST NOT reference a components object.

**transform** - The transform to apply to the selected object before the boolean operation.

**path** - When used in conjunction with [the 3MF Production extension](#), the "path" attribute references objects in non-root model files. Path is an absolute path to the target model file inside the 3MF container that contains the target object. The use of the path attribute in a <boolean> element is ONLY valid in the root model file.

The boolean operations are sequentially applied in the order defined by the <boolean> sequence, and they follow the fill rule conversion defined by [the 3MF Core Specification fill rule](#).

## Part II. Appendices

---

### Appendix A. Glossary

See [the 3MF Core Specification glossary](#).

### Appendix B. 3MF XSD Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns="http://schemas.microsoft.com/3dmanufacturing/booleanoperations/2022/06"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"

  targetNamespace="http://schemas.microsoft.com/3dmanufacturing/booleanoperations/2022/06"
  elementFormDefault="unqualified" attributeFormDefault="unqualified"
  blockDefault="#all">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xs:annotation>
    <xs:documentation><![CDATA[ Schema notes:

    Items within this schema follow a simple naming convention of appending a prefix
    indicating the type of element for references:

    Unprefixed: Element names
    CT_: Complex types
    ST_: Simple types

    ]]></xs:documentation>
  </xs:annotation>
  <!-- Complex Types -->
  <xs:complexType name="CT_Mesh">
    <xs:sequence>
```



9 / 11

```

    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="ST_Path">
    <xs:restriction base="xs:string"> </xs:restriction>
  </xs:simpleType>

  <!-- Elements -->
  <xs:element name="booleans" type="CT_Booleans"/>
  <xs:element name="boolean" type="CT_Boolean"/>
</xs:schema>

```

## Appendix C. Standard Namespace

---

BooleanOperation <http://schemas.microsoft.com/3dmanufacturing/booleanoperations/2022/06>

## Appendix D: Example file

---

### 3D model

```

<?xml version="1.0" encoding="utf-8" standalone="no"?>
<model xmlns="http://schemas.microsoft.com/3dmanufacturing/core/2015/02"
  xmlns:bo="http://schemas.microsoft.com/3dmanufacturing/booleanoperations/2022/
06"
  requiredextensions="bo" unit="millimeter" xml:lang="en-US">
  <resources>
    <basematerials id="2">
      <base name="Red" displaycolor="#FF0000" />
      <base name="Green" displaycolor="#00FF00" />
      <base name="Blue" displaycolor="#0000FF" />
    </basematerials>
    <object id="3" type="model" name="Cube" pid="2" pindex="0">
      <mesh>
        <vertices>...</vertices>
        <triangles>...</triangles>
      </mesh>
    </object>
    <object id="4" type="model" name="Sphere" pid="2" pindex="2">
      <mesh>
        <vertices>...</vertices>
        <triangles>...</triangles>
      </mesh>
    </object>
    <object id="5" type="model" name="Cylinder" pid="2" pindex="1">
      <mesh>
        <vertices>...</vertices>
        <triangles>...</triangles>
      </mesh>
    </object>
  </resources>
</model>

```

```

    </mesh>
  </object>
  <object id="6" type="model" name="Intersected">
    <mesh>
      <vertices/>
      <triangles/>
      <bo:booleans objectid="3" operation="intersection" transform="0.0741111
0 0 0 0.0741111 0 0 0 0.0741111 2.91124 -0.400453 1.60607">
        <bo:boolean objectid="4" transform="0.0741111 0 0 0 0.0741111 0 0 0
0.0741111 2.91124 -0.400453 1.60607"/>
      </bo:booleans>
    </mesh>
  </object>
  <object id="10" type="model" name="Full part">
    <mesh>
      <vertices/>
      <triangles/>
      <bo:booleans objectid="6" operation="difference">
        <bo:boolean objectid="5" transform="0.0271726 0 0 0 0 0.0271726 0
-0.0680034 0 4.15442 3.58836 5.23705" />
        <bo:boolean objectid="5" transform="0.0272014 0 0 0 0.0272012 0 0 0
0.0680035 4.05357 6.33412 3.71548" />
        <bo:boolean objectid="5" transform="0 0 -0.0272013 0 0.0272013 0
0.0680032 0 0 5.05103 6.32914 3.35287" />
      </bo:booleans>
    </mesh>
  </object>
</resources>
<build>
  <item objectid="10" transform="25.4 0 0 0 25.4 0 0 0 25.4 0 0 0" />
</build>
</model>

```

## References

---

**CSG** Wikipedia, the free encyclopedia: [Constructive solid geometry](#)

See the [3MF Core Specification references](#) for additional references.