

AMF Tutorial: The Basics (Part 1)

Hod Lipson¹

Abstract

This tutorial provides an introduction to the AMF language by walking through an example of a simple pyramid shape composed of two solid materials. This is the first in a series of tutorials that cover graded materials, colors, textures, lattices, curved triangles, constellations, and other AMF features.

ANYONE WORKING WITH 3D PRINTING has grown accustomed to handling STL files. STL files are the *lingua franca* of additive manufacturing, allowing transmission of geometric data between a plurality of design software tools and manufacturing systems. While most users never need to delve into the inner workings of this file format, many users have grown accustomed to its benefits. STL file format can reliably describe any shape that can be represented as a tessellation of triangles; it can be interpreted in a single pass, and it is easy to generate, parse, and display, leading to numerous applications.

Advanced users also quickly become aware of its limitations: STL files have data redundancies and unit ambiguities; they scale poorly with complex geometries such as lattices and high-resolution curved surfaces; and the format cannot accommodate colors and multiple materials. This growing gap between what 3D printers can fabricate

and what STL files can describe has led to an effort by international standardization organizations to create a new additive manufacturing language, leading to the ISO/ASTM 52915 AMF specification.¹

The goal of this series of tutorials is to demonstrate the use of AMF through examples. In this tutorial, we will begin by examining a short but complete AMF file that describes a simple pyramid composed of two materials: a hard, dark material and a soft translucent material (Fig. 1). This multimaterial, multicolor object can be described in only 523 bytes of code (after compression), yet maintains a human-readable form. The code itself, shown in Figure 2, is easily edited using any text or XML editor and visualized using any AMF viewer. The original file, as well as a viewer, can be downloaded from the AMF wiki.²

The formatting of any AMF file follows the general XML syntax, where the code is divided into markup and content. The

markup tags are identified by angle brackets, and the content is the data in between an opening and closing tag. As any XML file, the first line is a declaration of XML version and encoding. This line could optionally include other standard XML attributes, such as language, but we skipped those here. After this line follows the main body of the AMF file. Importantly, the opening `<amf>` tag declares a unit of measurement, in this case “inch.”

Every AMF file contains at least one object, defined within the `<object>` tags. The file may contain also additional objects, as well as optional materials and textures. Multiple objects can be arranged in constellations. We will discuss constellations and textures in a future tutorial; our file here has only one object and two materials.

Let’s take a closer look at the `<object>` section. The object is assigned an identifier, in this case “1,” and is then

¹Cornell University, Ithaca, New York.

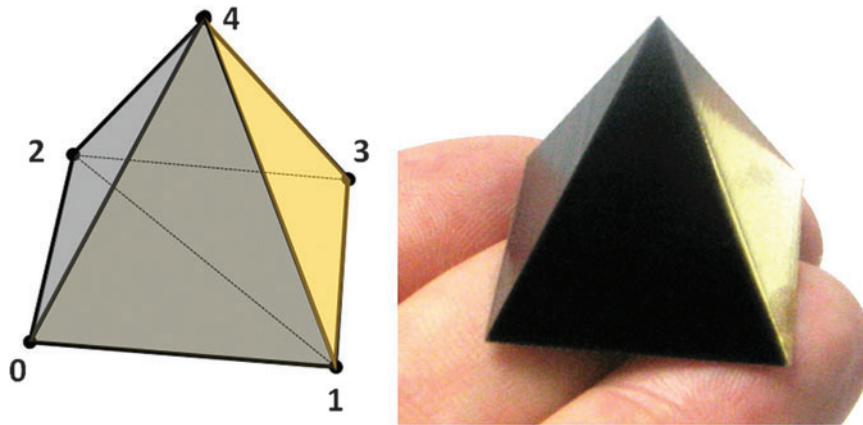


Figure 1. Split pyramid object described in this tutorial. **(a)** Schematic of vertex placement. **(b)** Final object printed directly from the AMF file. Color images available online at www.liebertpub.com/3dp

defined using a mesh tag. The mesh comprises a set of vertices and a set of triangles, all contained between the opening and closing `<mesh>` tags. The section listing vertices is outlined in green (Fig. 2c), and the sections listing the two volumes are outlined in red (Fig. 2d) and blue (Fig. 2e).

Within the green section, the AMF lists the 3D coordinates of five vertices. The vertices are implicitly numbered from 0

to 4, as shown in Figure 1a. In the future, we will explore how we can assign color, texture, and normal to each vertex, but for now each vertex contains only three coordinates.

Once the five vertices have been defined, the AMF proceeds to describe the two volumes. Each volume is described as a list of four triangles, where the vertex indices of each triangle are listed in the counter-clockwise order

when viewed from outside the volume. Importantly, each volume is also associated with a single material—one volume (Fig. 2d) is associated with a material called “Dark material” and the other volume (Fig. 2e) is associated with a material called “Soft material.” In future tutorials, we will learn how to define more sophisticated “metamaterials” by composing base materials in gradations and patterns, but for now we use solid materials.

Once the vertices and the triangles have been defined, the mesh section can be closed using the `</mesh>` tag. It is important to know that, in principle, we could define an object with as many volumes as we like, as long as we follow some basic rules. First, volumes cannot overlap each other, but they are allowed to touch at points, along edges or across triangles. Volumes must be fully closed (watertight), cannot have zero volume, and cannot self-intersect.

Finally, the AMF describes the two materials referred to by the volumes. The first material, shown in Figure 2g, has a dark-gray RGB color (0.1,0.1,0.1), and is labeled as “Hard material.” The second material, shown in Figure 2h, has a yellowish translucent RGBA color (0,0.9,0.9,0.5), and is labeled as “Soft material.” These labels, defined using `<metadata>` tag, are not interpreted by the computer. Instead, the labels are used to instruct the printer’s operator to load a “Hard material” into material bay 1, and a “Soft material” into the second material bay. The RGB values tell the AMF viewer how to display the material on the screen.

AMF allows for a variety of metadata types to be included anywhere in the XML hierarchy. For example, names can be assigned to materials, to objects, to volumes, and even to individual vertices or triangles. These labels often serve for documentation, for inspection, and even for debugging. For example, a printer can inform the designer that a certain named volume is causing an error. The AMF standard provides other metadata types, including “author,” “e-mail,” “copyright,” “url,” “producer,” “revision,” and more (see ref.¹). Custom metadata can also be provided, but must use a different XML namespace so as not to be confused with

```

<?xml version="1.0" encoding="utf-8"?>
<amf unit="inch" version="1.1">
  <metadata type="name">Split Pyramid</metadata>
  <metadata type="author">Hod Lipson</metadata>
  <object id="1">
    <mesh>
      <vertices>
        <vertex><coordinates><x>0</x><y>0</y><z>0</z></coordinates></vertex>
        <vertex><coordinates><x>1</x><y>0</y><z>0</z></coordinates></vertex>
        <vertex><coordinates><x>0</x><y>1</y><z>0</z></coordinates></vertex>
        <vertex><coordinates><x>1</x><y>1</y><z>0</z></coordinates></vertex>
        <vertex><coordinates><x>0.5</x><y>0.5</y><z>1</z></coordinates></vertex>
      </vertices>
      <volume materialid="2">
        <metadata type="name">Hard side</metadata>
        <triangle><v1>2</v1><v2>1</v2><v3>0</v3></triangle>
        <triangle><v1>0</v1><v2>1</v2><v3>4</v3></triangle>
        <triangle><v1>4</v1><v2>1</v2><v3>2</v3></triangle>
        <triangle><v1>0</v1><v2>4</v2><v3>2</v3></triangle>
      </volume>
      <volume materialid="3">
        <metadata type="name">Soft side</metadata>
        <triangle><v1>2</v1><v2>3</v2><v3>1</v3></triangle>
        <triangle><v1>1</v1><v2>3</v2><v3>4</v3></triangle>
        <triangle><v1>4</v1><v2>3</v2><v3>2</v3></triangle>
        <triangle><v1>4</v1><v2>2</v2><v3>1</v3></triangle>
      </volume>
    </mesh>
  </object>
  <material id="2">
    <metadata type="name">Hard material</metadata>
    <color><r>0.1</r><g>0.1</g><b>0.1</b></color>
  </material>
  <material id="3">
    <metadata type="name">Soft material</metadata>
    <color><r>0</r><g>0.9</g><b>0.9</b><a>0.5</a></color>
  </material>
</amf>

```

Figure 2. **(a–i)** AMF code for the split pyramid object. 523 bytes after compression. Color images available online at www.liebertpub.com/3dp

the standard metadata types, for example, `<xamf:metadata type="myattribute">mydata </xamf:metadata>`

The file ends with a closing `</amf>` tag. Once the AMF file is saved, it is compressed using standard ZIP file compression and the file type renamed back to ".amf." The compressed file can be viewed by an AMF viewer or printed by any AMF-compatible printer. Software that generates AMF files internally can compress them "on the fly" using any one of a number of ZIP compression libraries.

To test your understanding of this tutorial, try the following exercises:

1. *Change the colors of the two materials to yellow and green.*
2. *Double the height of the pyramid.*
3. *Split the pyramid into four sections instead of two. You'll need to add one more vertex.*

References

1. Standard Specification for Additive Manufacturing File Format (AMF)

Version 1.2, ISO/ASTM 52915. www.astm.org/Standards/ISOASTM52915.htm (Last accessed June 6, 2014).
 2. AMF wiki. <http://STL2.org> (Last accessed June 6, 2014).

Address correspondence to:

*Hod Lipson
 242 Upson Hall
 Cornell University
 Ithaca, NY 14853-7501*

E-mail: hod.lipson@cornell.edu