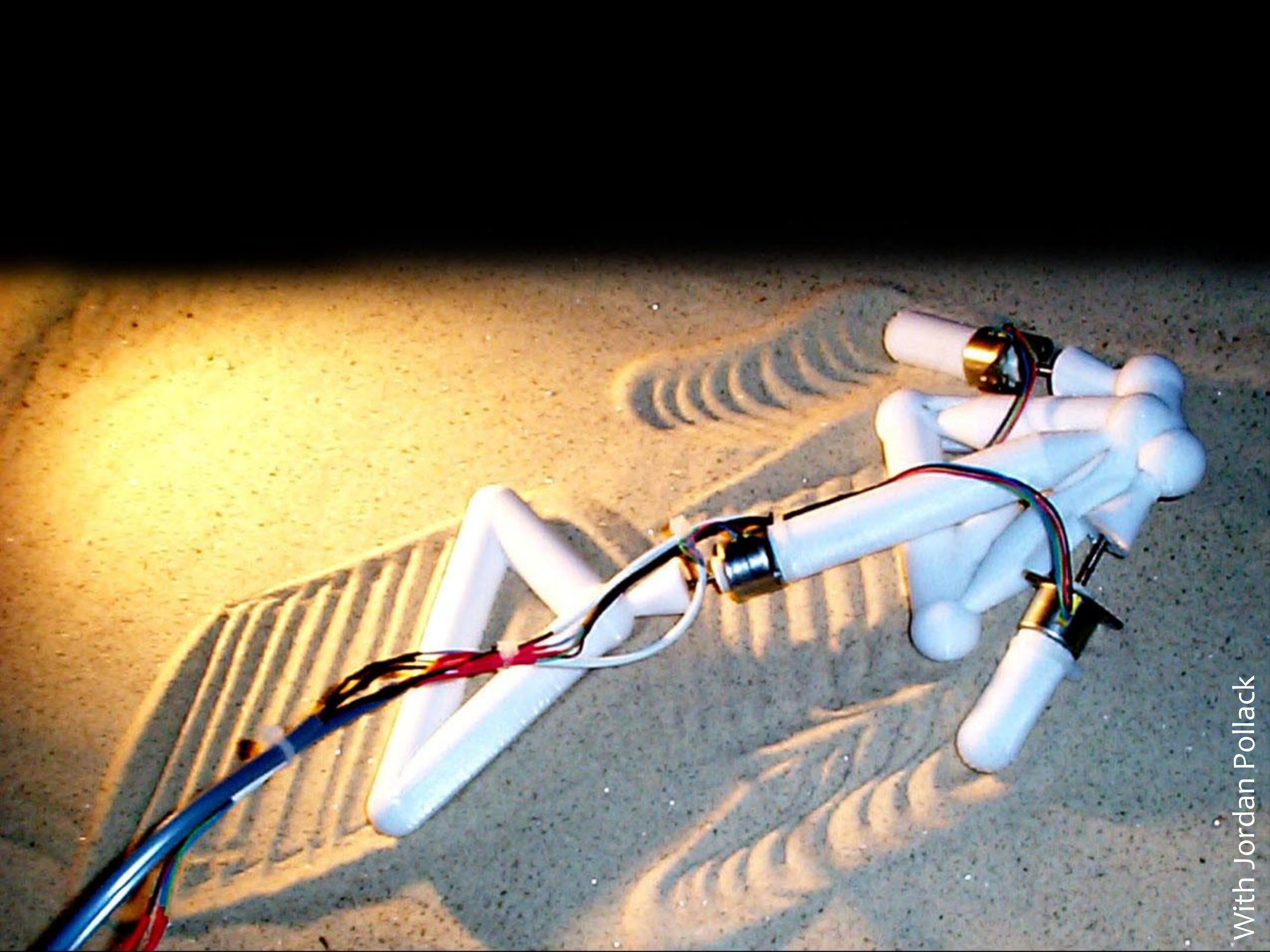
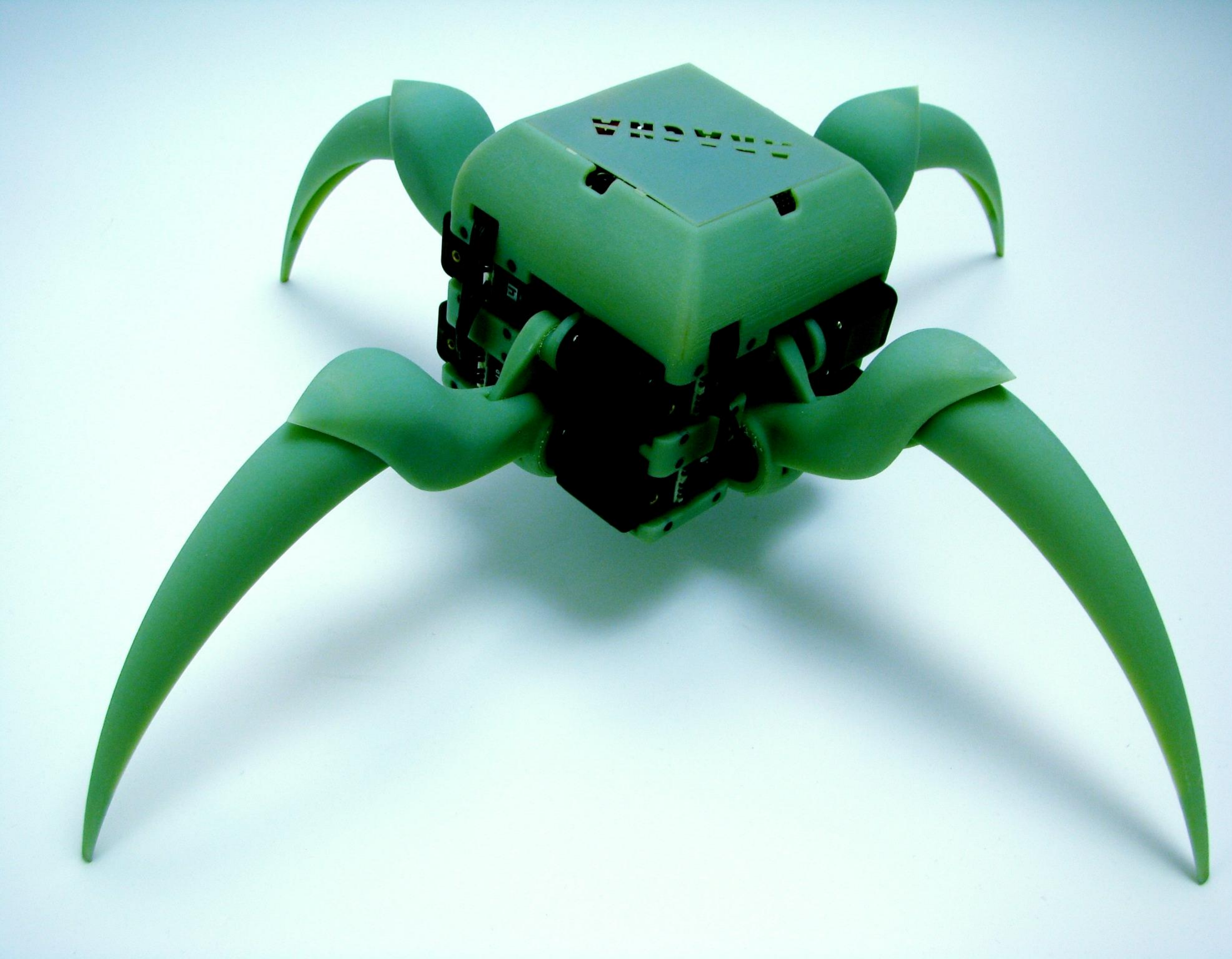
Additive Manufacturing File Format

Hod.lipson@cornell.edu

Chair, ASTM F42/Design Task Group on File Formats

Disclaimer: Information in this presentation does not constitute the final standard. Actual specifications is subject to change until finalized by ASTM

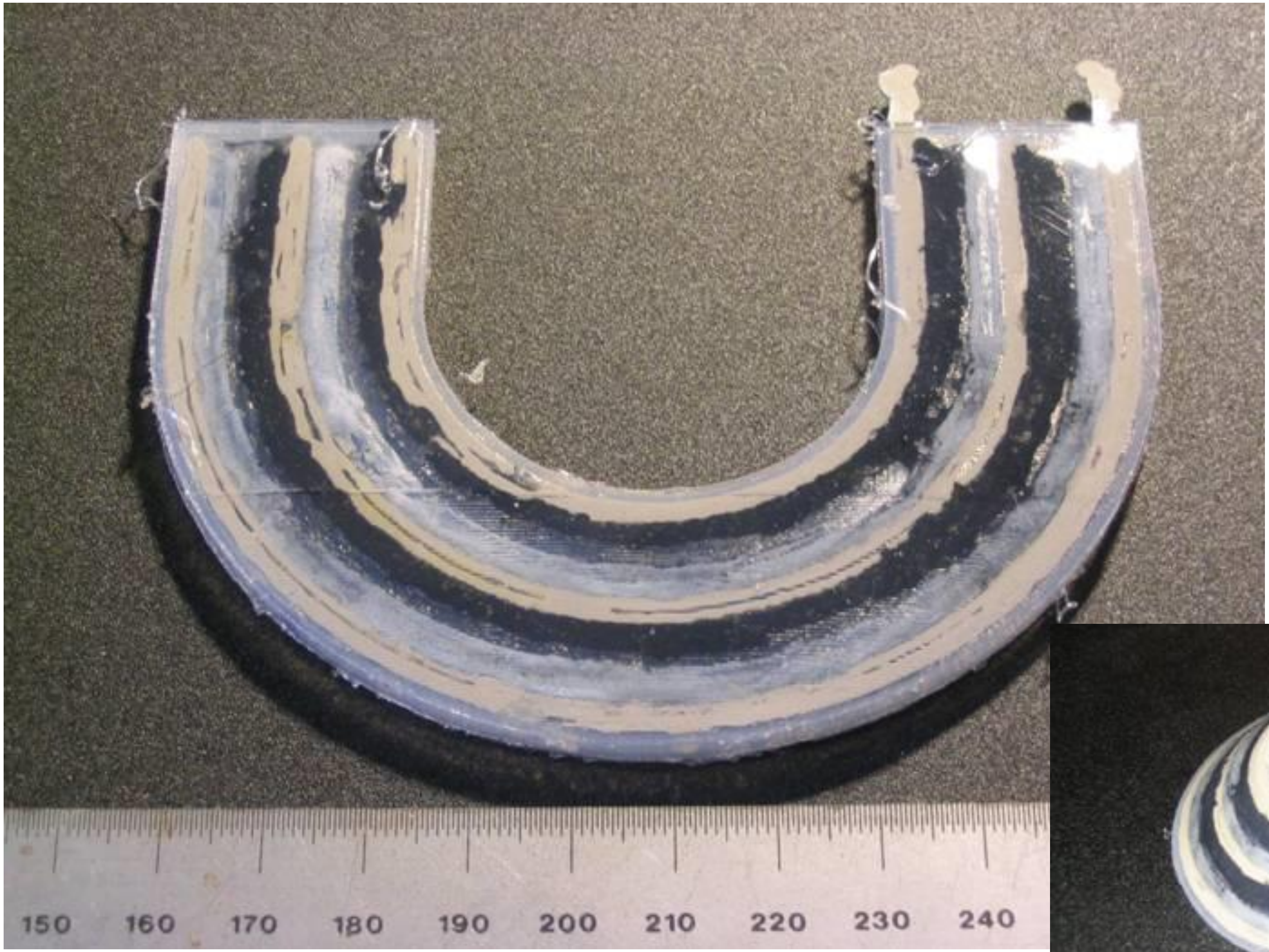

With Jordan Pollack

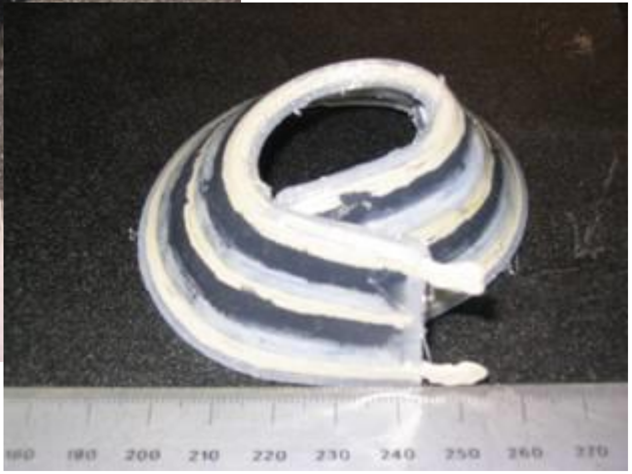


Sara Loher

Charlie Richter

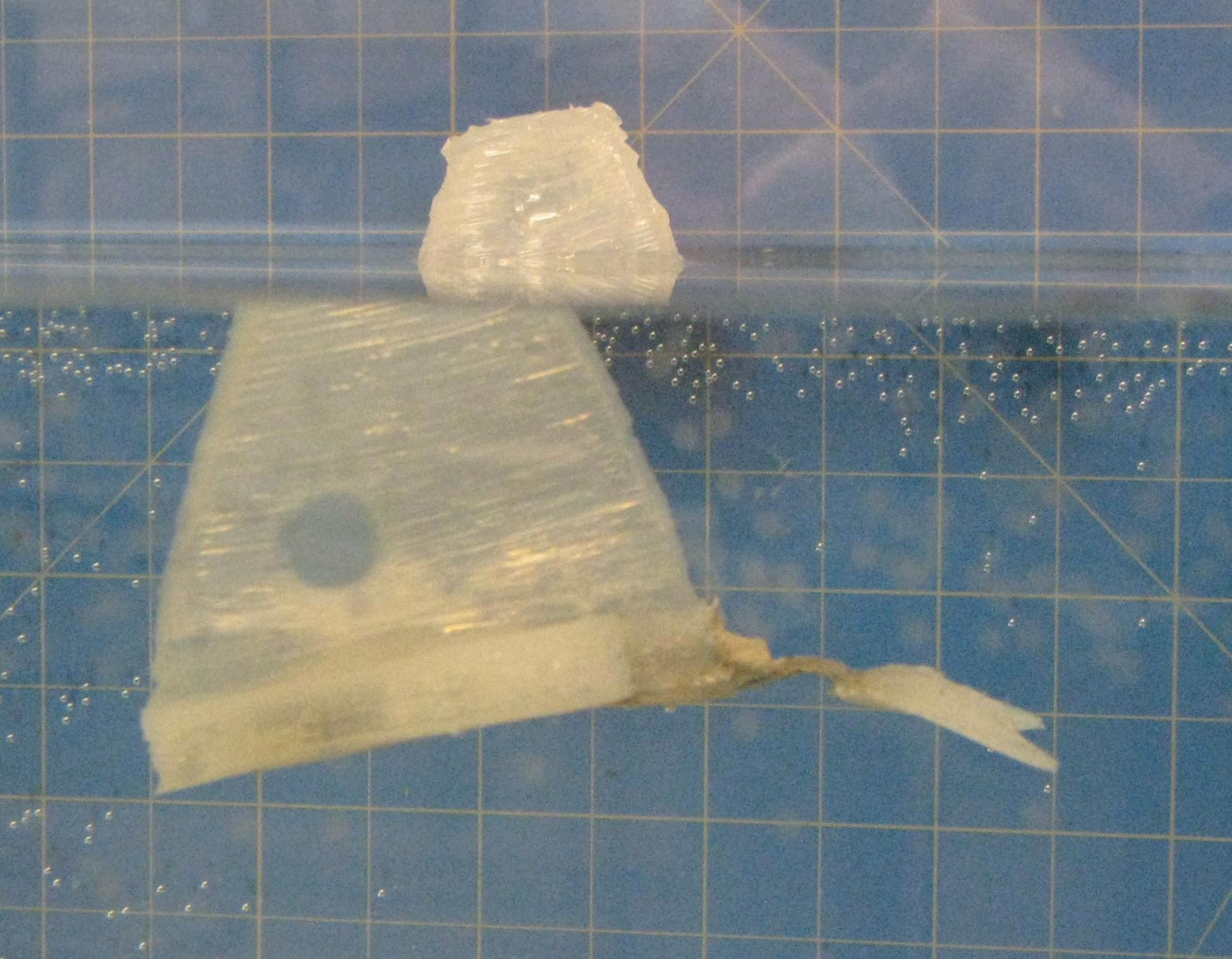
Batteries





Zinc-Air

Megan Berry



Evan Malone

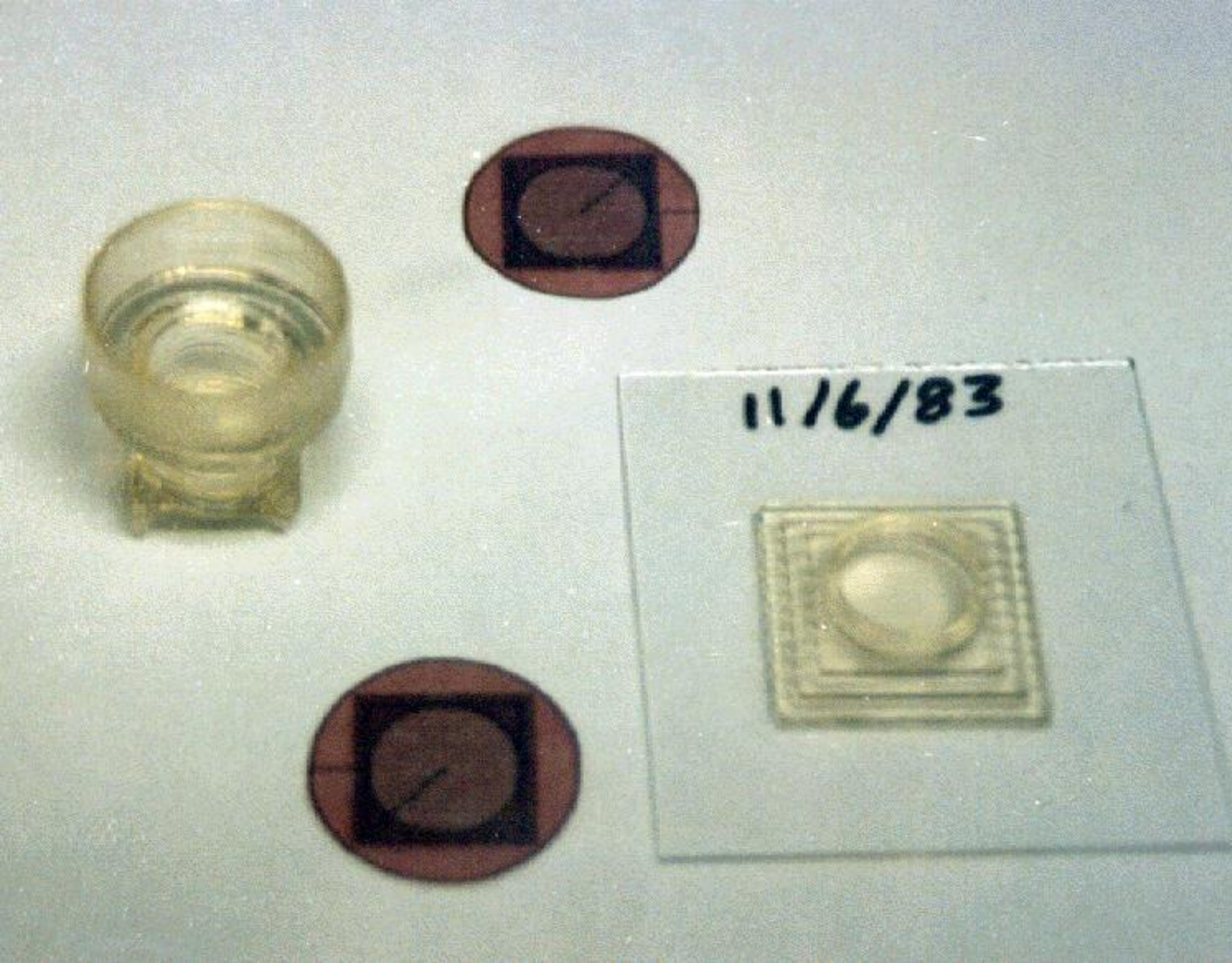
Dsnny Cohen, Larry Bonassar

Jeff Lipton, Franz Nigl


Danny Cohen, Dave Arnold

10 principles

1. Manufacturing complexity is free
2. Variety is free
3. No lead time
4. Zero skill
5. Less waste by-product
6. …



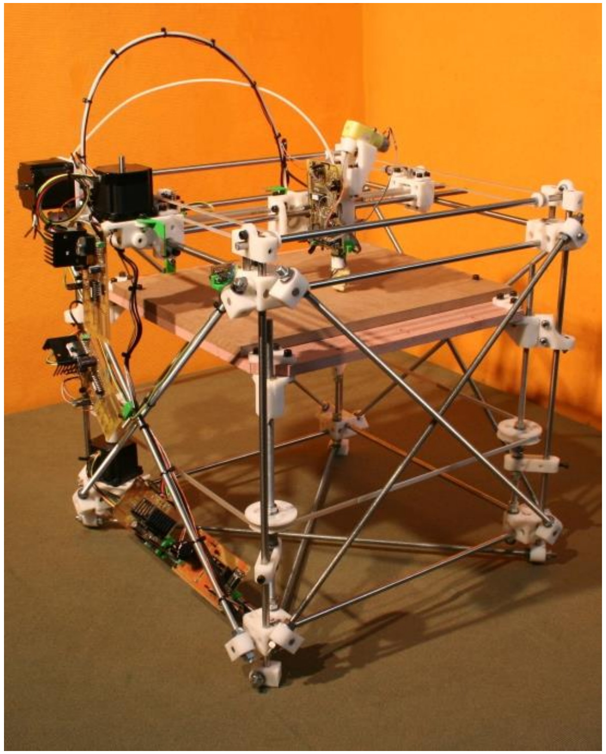
Chuck Hull

$15,000

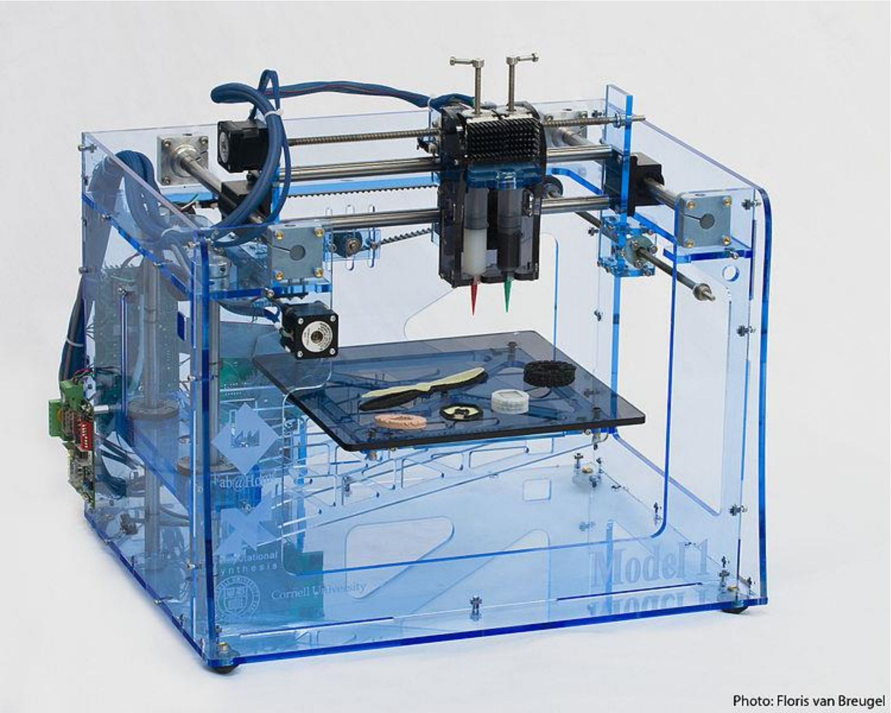
$250,000

> $500,000

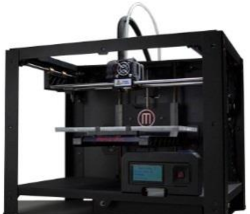
Open Source 3D Printers



RepRap (2005)
University of Bath, UK



Fab@Home (2006)
Cornell University, NY

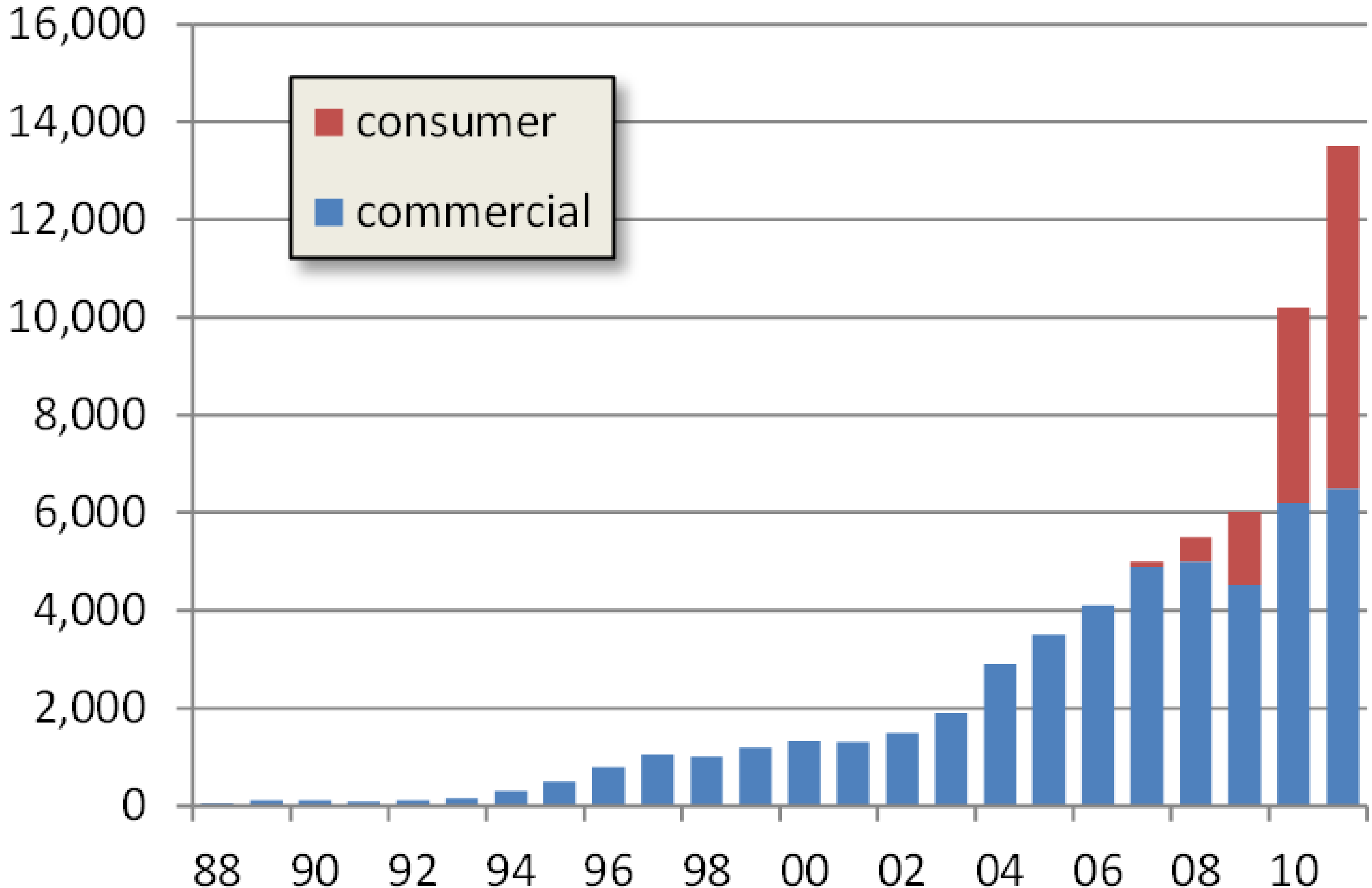


$2,000

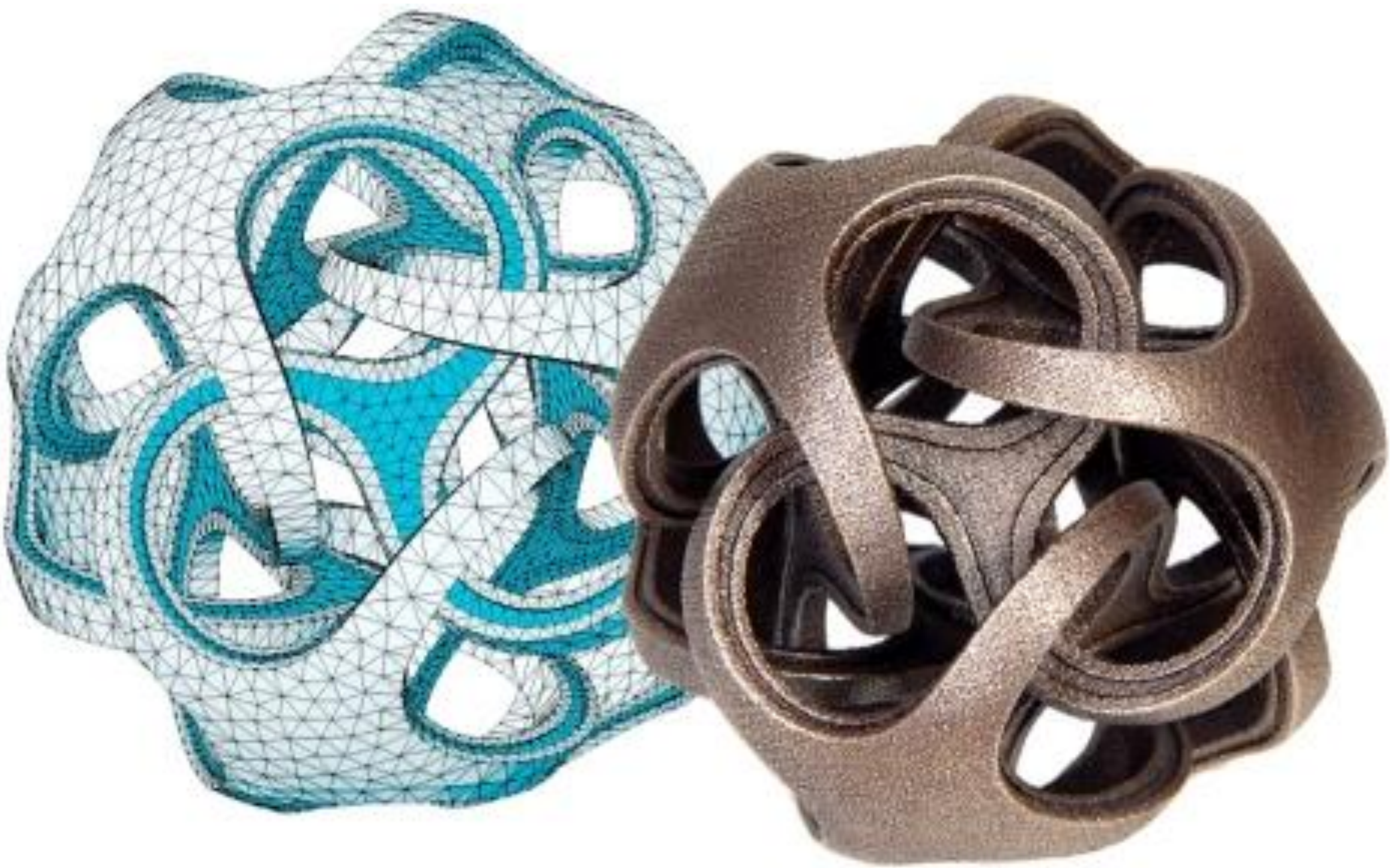
$15,000

$250,000

> $500,000

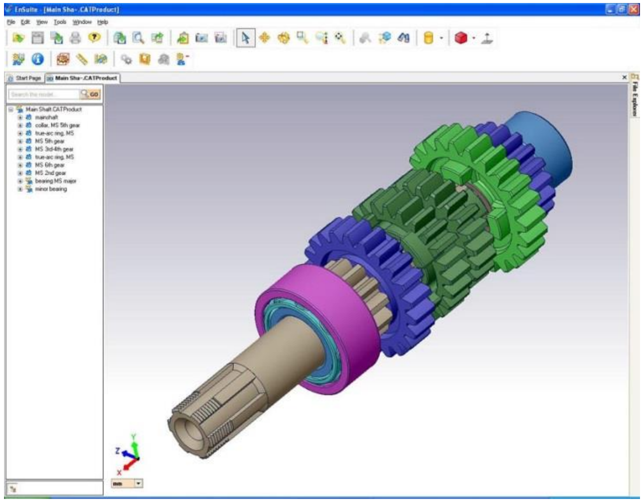


16,000
14,000
12,000
10,000
8,000
6,000
4,000
2,000
0
consumer
commercial
88
90
92
94
96
98
00
02
04
06
08
10



Bathsheba Grossman

Data Interchange



STL

Postscript

STL

- Benefits
  - *De-facto* standard
  - Very simple to read/write/process
- Challenges
  - Duplicate information, leaks, inconsistencies
  - Does not scale well to high resolution, lattices
  - Does not support color, materials, orientation

Holding back the industry

Designation: F XXXX – 10

Standard Specification for Additive Manufacturing File Format (AMF)[1]

This standard is issued under the fixed designation F XXXX; the number immediately following the designation indicates the year of original adoption or, in the case of revision, the year of last revision. A number in parentheses indicates the year of last re-approval. A superscript epsilon (ε) indicates an editorial change since the last revision or re-approval.

1. Scope[1]

For the last three decades, the STL file format has been the industry standard for transferring information between design programs and additive manufacturing equipment. As additive manufacturing technology is quickly evolving from producing primarily single-material, homogenous shapes to producing multi-material geometries in full color with functionally graded materials and microstructures, there is a growing need for a standard interchange file format that can support these features. An STL file contains information only about a surface mesh, and has no provisions for representing color, texture, material, substructure, and other properties of the fabricated target object. This standard describes a framework for a interchange format to address the current and future needs of additive manufacturing technology.

The AMF file may be prepared, displayed, and transmitted on paper or electronically, provided the information required by this specification is included. When prepared in a structured electronic format, strict adherence to an XML schema is required to support standards-compliant interoperability. The Adjunct to this specification contains a W3C XML schema and Annex A1 contains an Implementation Guide for such representation.

2.1 Contributors

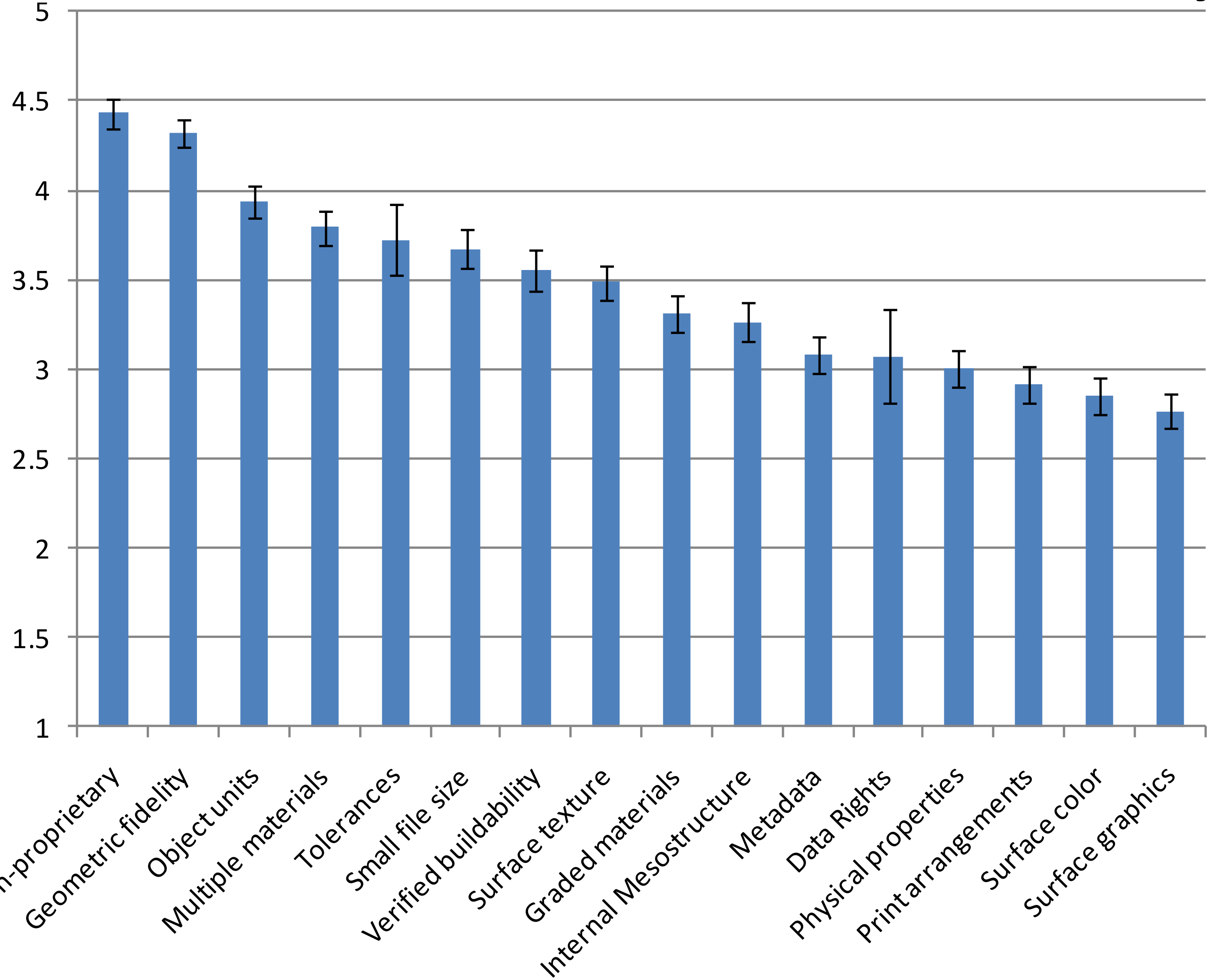
This standard has been prepared based on a survey and consensus among stakeholders representing designers, equipment manufacturers, CAD software developers, and academicians. A list of contributors and supporters is provided in Appendix 2.

2. Key considerations

There is a naturally a tradeoff between the generality of a file format, and its usefulness for a specific purpose. Thus, features designed to meet the needs of one community may hinder the usefulness of a file format for other uses. In order to be successful across the field of additive manufacturing, this file format is designed to address the following concerns:

2.1 Technology independence: The file format shall describe an object in a general way such that any machine can build it to the best of its ability. It is resolution and layer-thickness independent, and does not contain information specific to any one manufacturing process or technique. This does not negate the inclusion of properties that only certain advanced machines support (for example, color, multiple materials, etc.), but these are defined in such away to avoid exclusivity.

Prioritized features from survey



n=162 (2010). Error bars = Standard Error

Key considerations

- **Technology independence**
  - Describes target object, not how to make it
  - Every machine can make it to the best of its ability
- **Simplicity**
  - Easy to understand and implement
- **Scalability**
  - Can handle complex objects, microstructures, repetitions
- **Performance**
  - File size, read/write time, processing, accuracy
- **Backwards compatible**
  - Can covert to/from STL without additional info
- **Forward compatible**
  - Easy to extend new features in the future

XML

- Meta-format: Format of formats
  - Text based
  - Easy to read/write/parse
  - Existing editing tools
  - Extensible
  - Highly compressible

General Concept

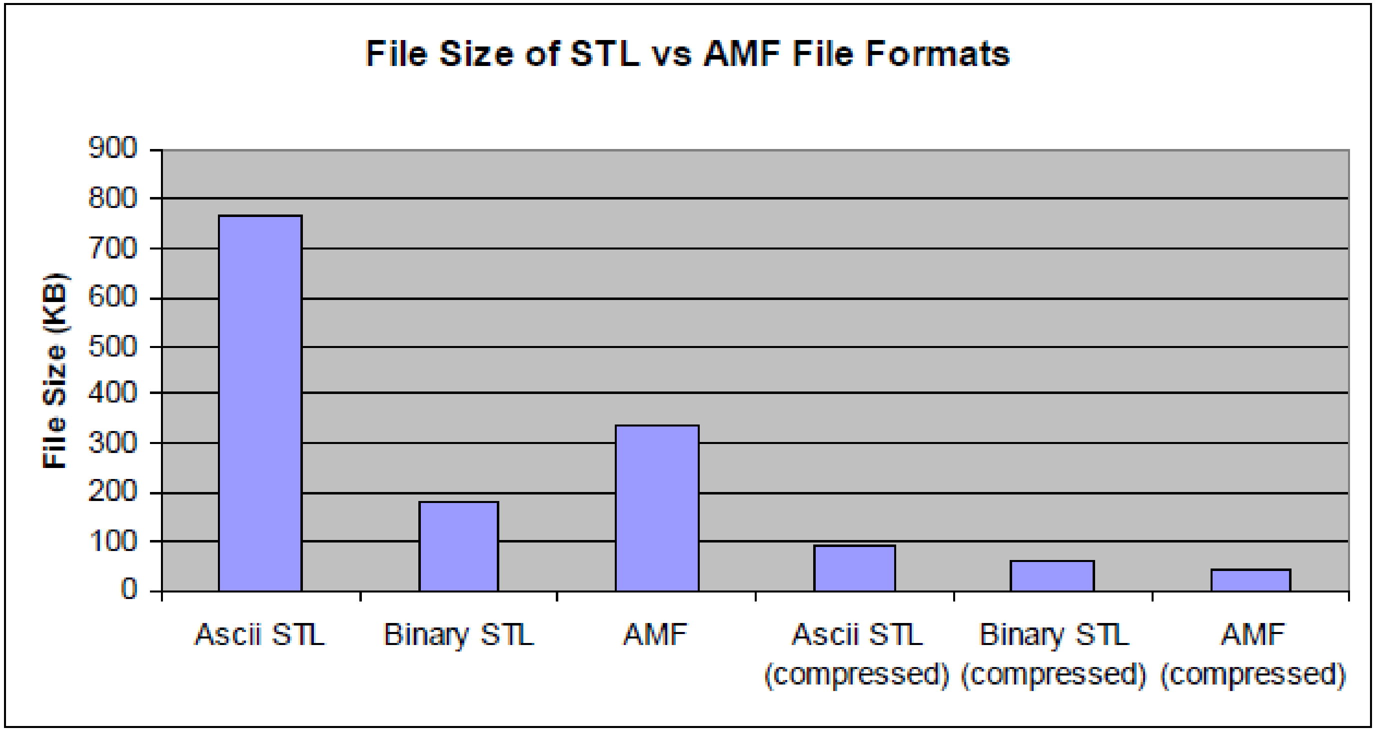
- **Objects (parts)** defined by volumes and materials
  - Volumes defined by triangular mesh
  - Materials defined by properties/names
- **Color properties** can be specified
  - Color
  - Texture mapping
- **Materials** can be combined
  - Graded materials
  - Lattice/Mesostructure
- **Constellations** of Objects can be defined
  - Repeated instances, packing, orientation

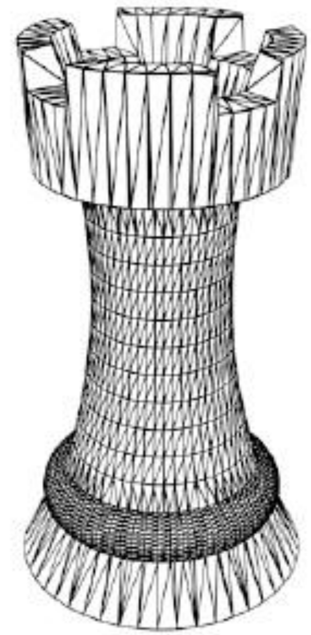
Basic AMF Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<amf units="mm">
  <object id="0">
    <mesh>
      <vertices>
        <vertex>
          <coordinates>
            <x>0</x>
            <y>1.32</y>
            <z>3.715</z>
          </coordinates>
        </vertex>
        <vertex>
          <coordinates>
            <x>0</x>
            <y>1.269</y>
            <z>2.45354</z>
          </coordinates>
        </vertex>
        ...
      </vertices>
      <volume>
        <triangle>
          <v1>0</v1>
          <v2>1</v2>
          <v3>3</v3>
        </triangle>
        <triangle>
          <v1>1</v1>
          <v2>0</v2>
          <v3>4</v3>
        </triangle>
        ...
      </volume>
    </mesh>
  </object>
</amf>
```

Addresses vertex duplication and leaks of STL

Compressibility





Comparison for 32-bit Floats; need to look at double precision

File Size

Number of Triangles	Binary STL (uncompressed)	Binary STL (compressed)	AMF (uncompressed)	AMF (compressed)
1,016,388	49.6 Mb	25.3 Mb	205.9 Mb	12.2 Mb
100,536	4.9 Mb	2.3 Mb	20.1 Mb	1.2 Mb
10,592	518 K	249 K	2.1 Mb	129 K
1,036	51 K	20 K	203 K	12 K

- Stored either as text or compressed (zip)
- Both versions have AMF extension
- Reader can determine which and decompress during read

Read/Write/Parse time

Write (seconds)

Number of Triangles	Binary STL (uncompressed)	Binary STL (compressed)	AMF (uncompressed)	AMF (compressed)
1,016,388	0.372	~3.4	6.8	15.5
100,536	0.038	0.038	0.79	1.78
10,592	0.005	0.005	0.11	0.21
1,036	0.001	0.001	0.06	0.06

Read + parse + construct data structure (seconds)

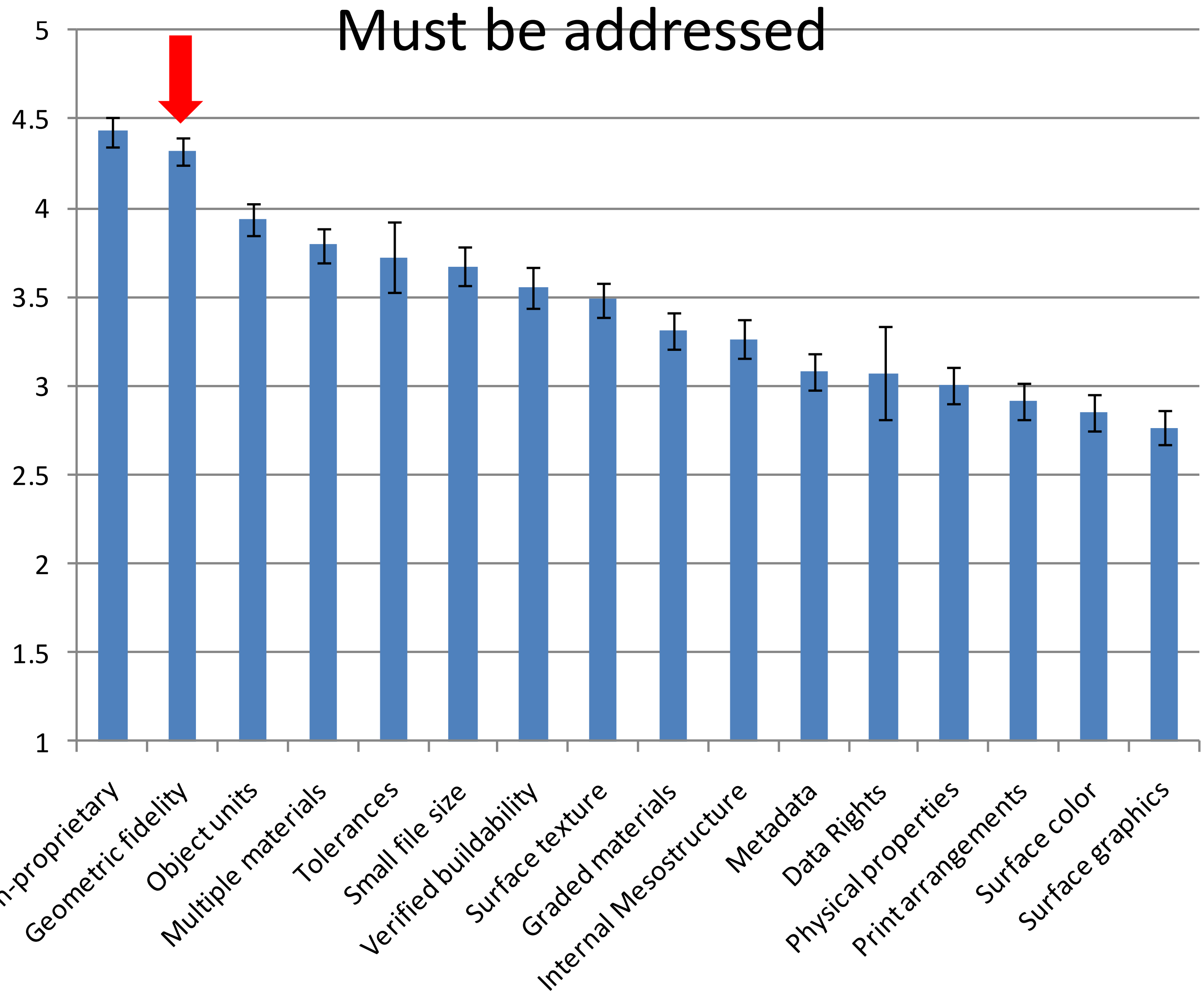
Number of Triangles	Binary STL (uncompressed)	Binary STL (compressed)	AMF (uncompressed)	AMF (compressed)
1,016,388	0.384	~1.3	6.447	6.447
100,536	0.043	0.043	0.669	0.687
10,592	0.005	0.005	0.107	0.107
1,036	0.001	0.001	0.056	0.056

Still negligible compared to slicing/processing time

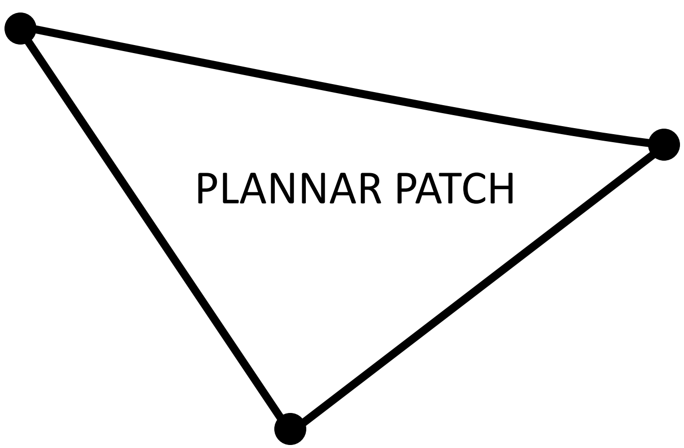
Increasing Geometric Accuracy

- **Flat triangles do not scale well for complex geometry, esepcially:**
  - Curved surfaces
  - Microstructures
- **Typical objects require millions of triangles**
  - 10M triangles not uncommon
- **Likely to get worse with increasing printer resolution**
  - 10cm sphere at 10μm requires 20,000 triangles

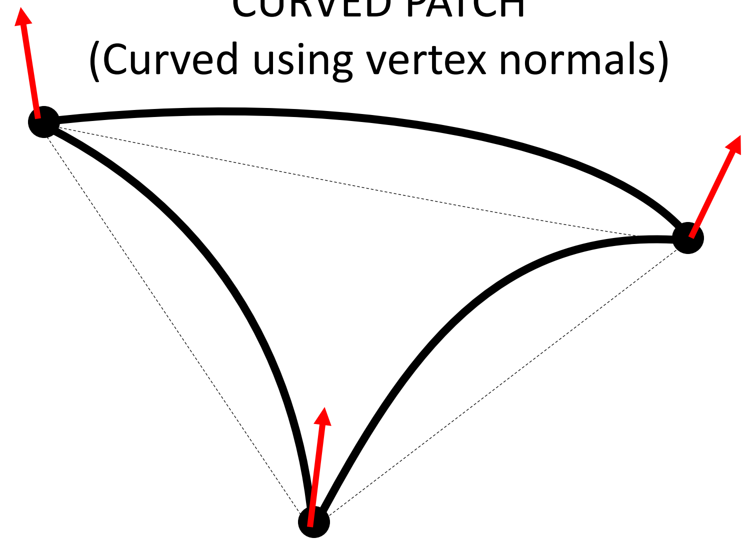
Geometric fidelity is a high priority



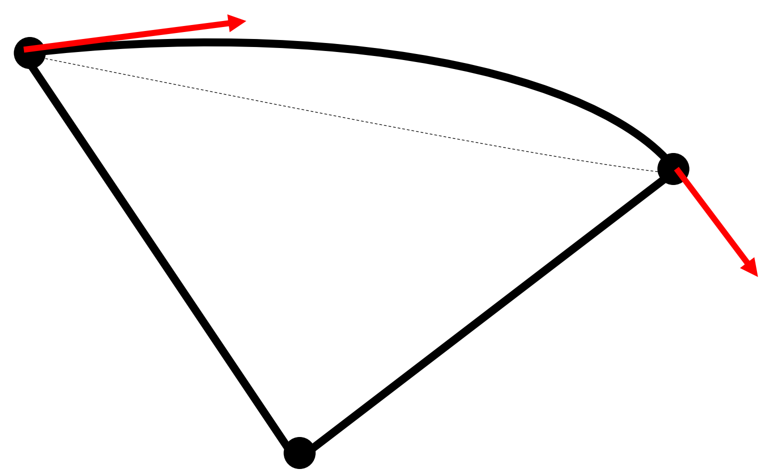
Curved patches



CURVED PATCH
(Curved using vertex normals)



CURVED PATCH
(or curved using edge tangents)



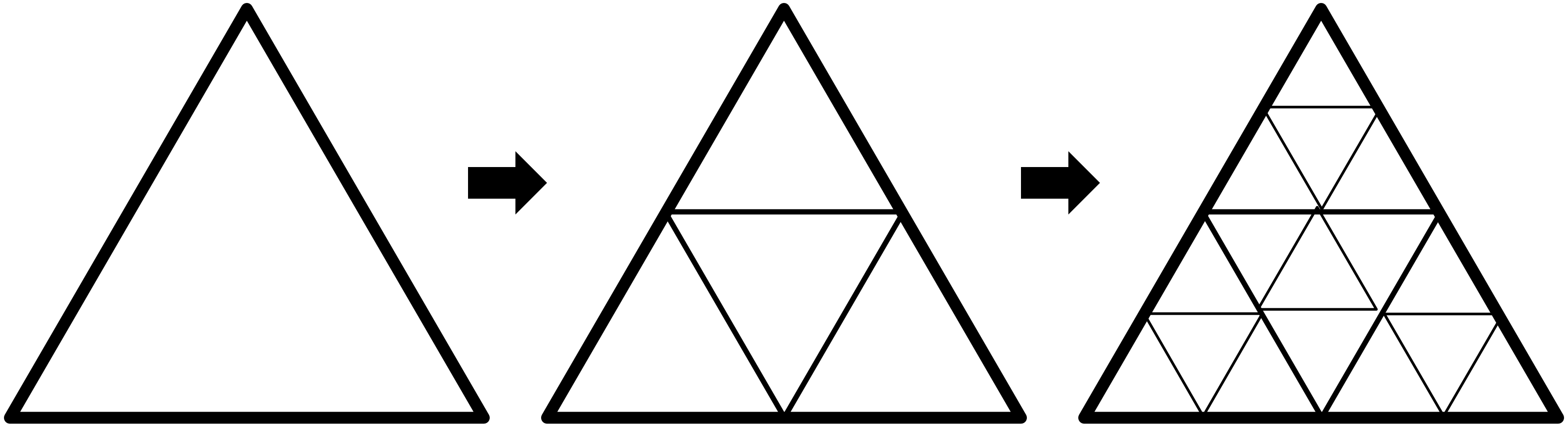
Optionally add normal/tangent vectors to some triangle mesh edges to allow for more accurate geometry.

Only needed for curved surfaces

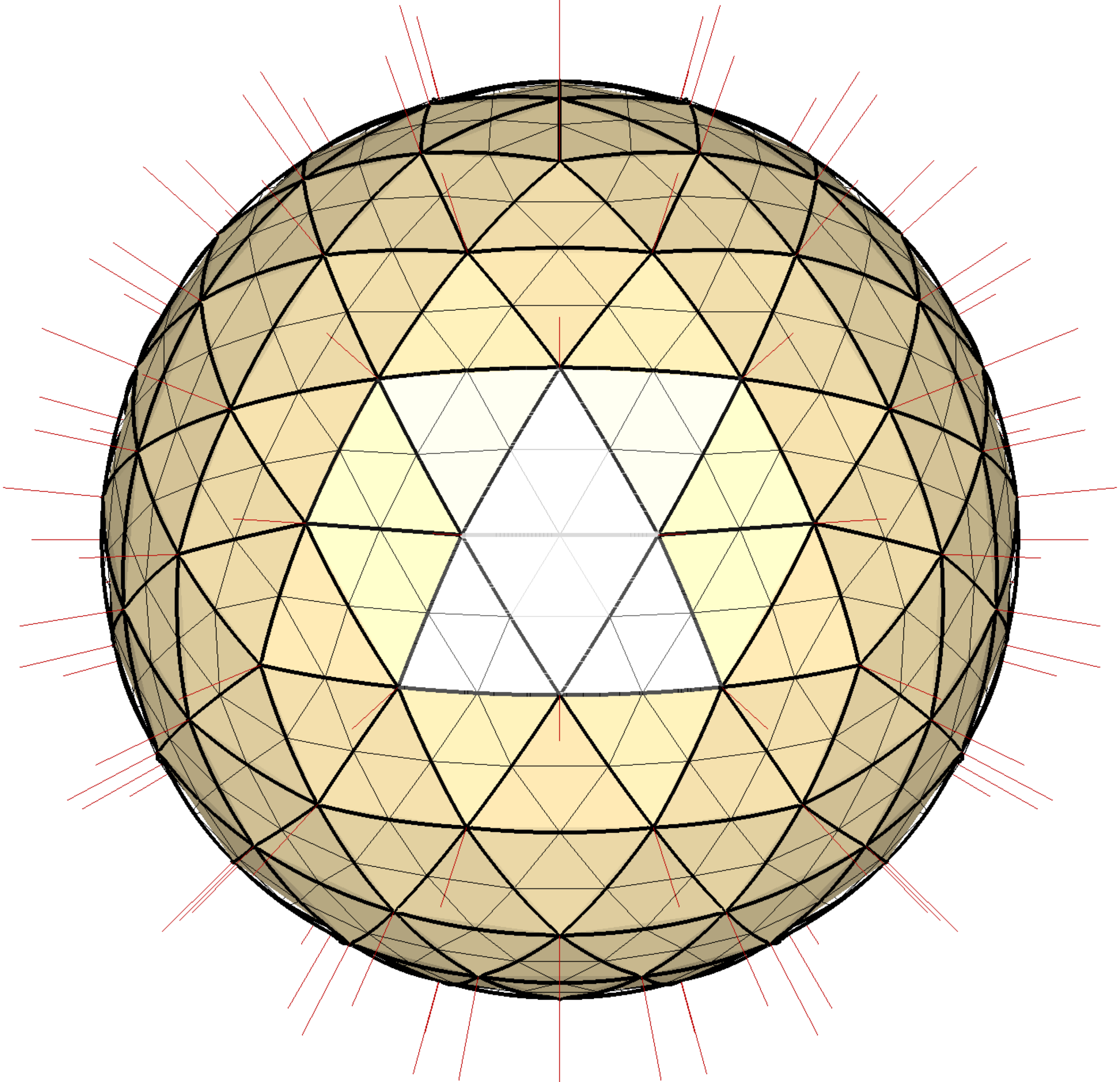
Only needed for curved edges (rare)

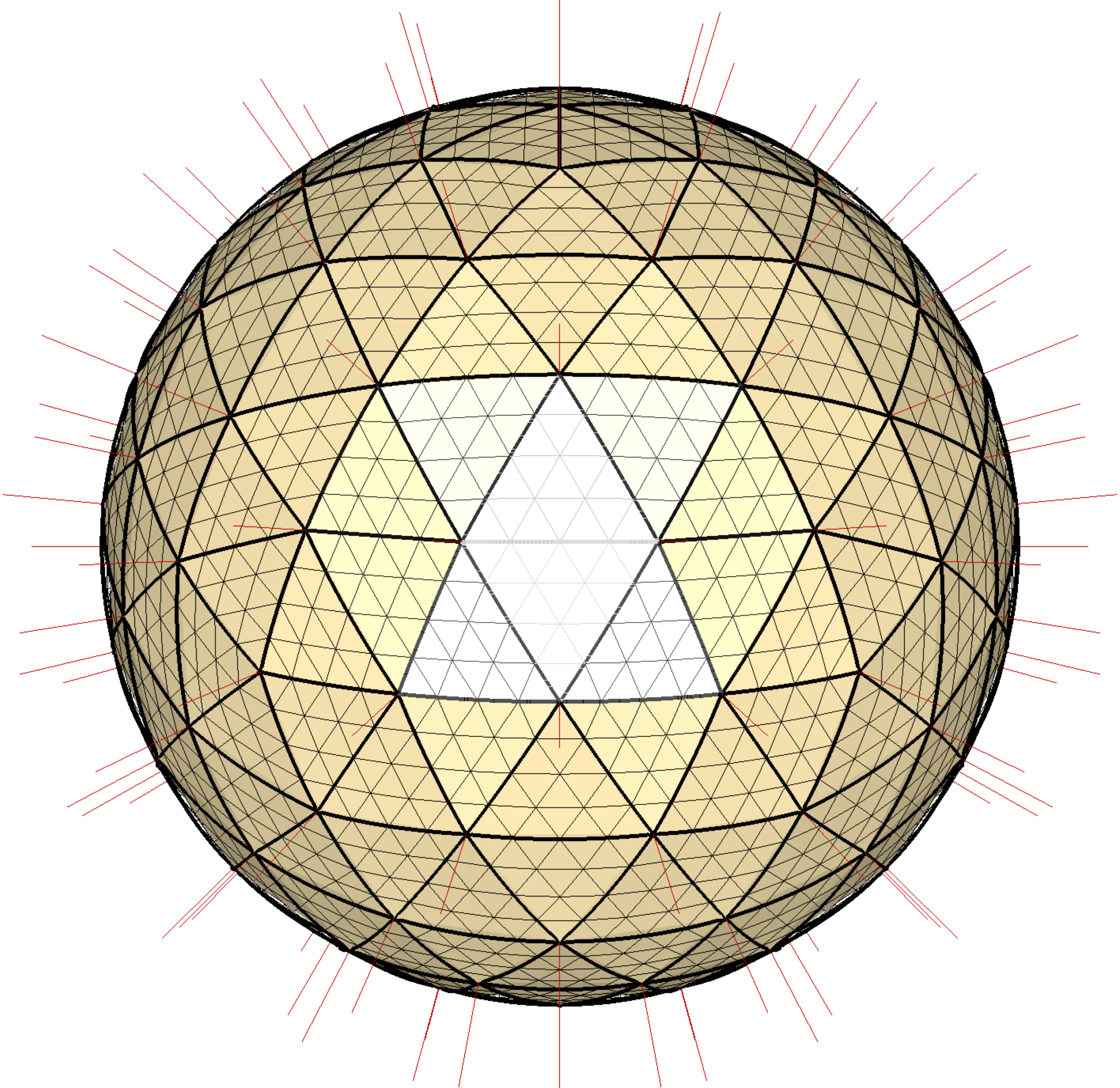
```
<?xml version="1.0" encoding="UTF-8"?>
<amf units="mm">
  <object id="0">
    <mesh>
      <vertices>
        <vertex>
          <coordinates >
            ...
          </coordinates >
          <normal>
            <nx>0</nx>
            <ny>0.707</ny>
            <nz>0.707</nz>
          </normal>
        </vertex>
        ...
        <edge>
          <v1>0</v1>
          <dx1>0.577</dx1>
          <dy1>0.577</dy1>
          <dz1>0.577</dz1>
          <v2>1</v2>
          <dx2>0.707</dx2>
          <dy2>0</dy2>
          <dz2>0.707</dz2>
        </edge>
        ...
      </vertices>
      <region materialid="0">
        <triangle>
          ...
        </triangle>
        ...
      </region>
    </mesh>
  </object>
</amf>
```

Recursive Triangle Subdivision

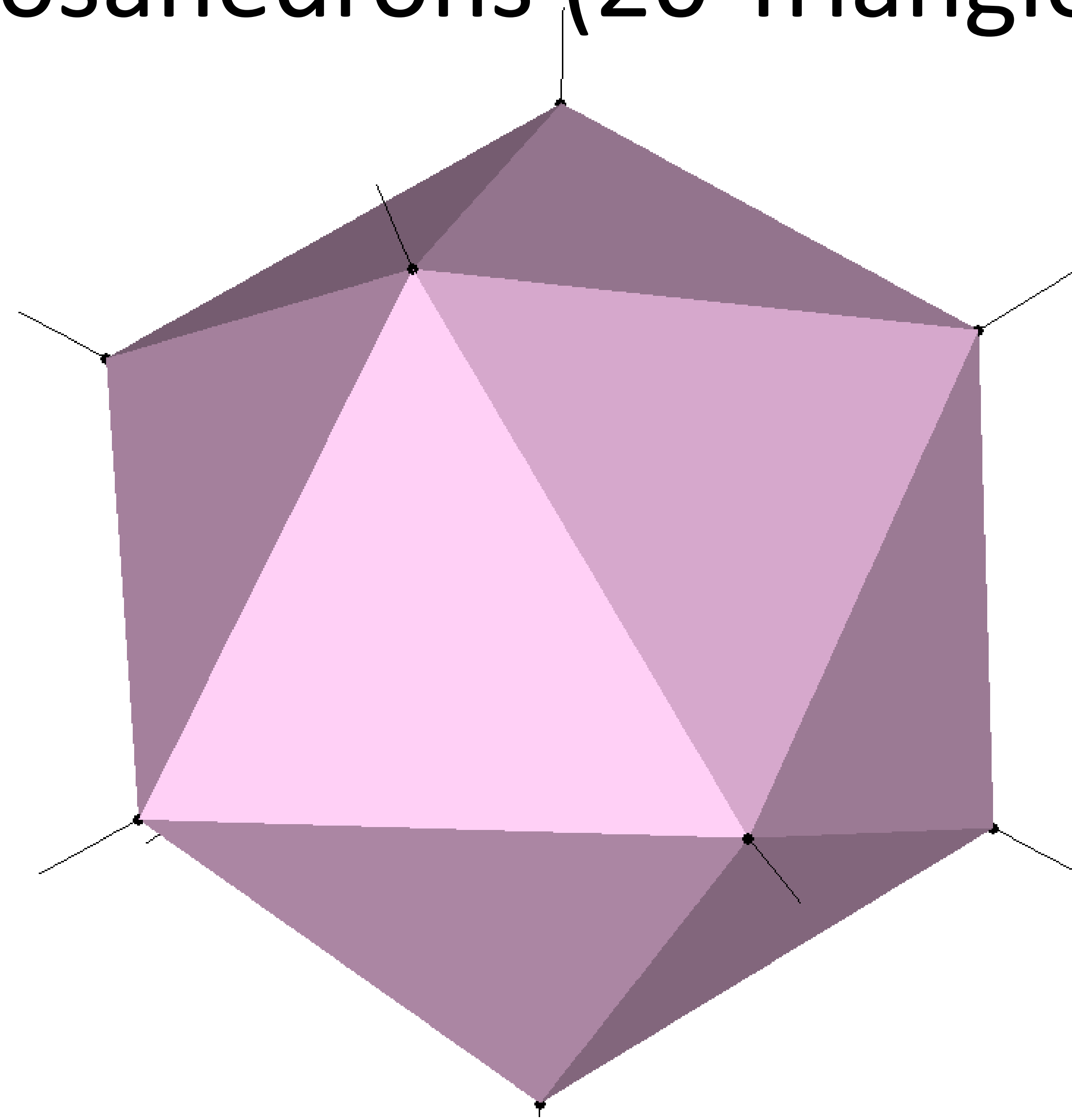


Importer temporarily subdivides each curved triangle into a set of 4^n planar triangles then uses those to calculate slice

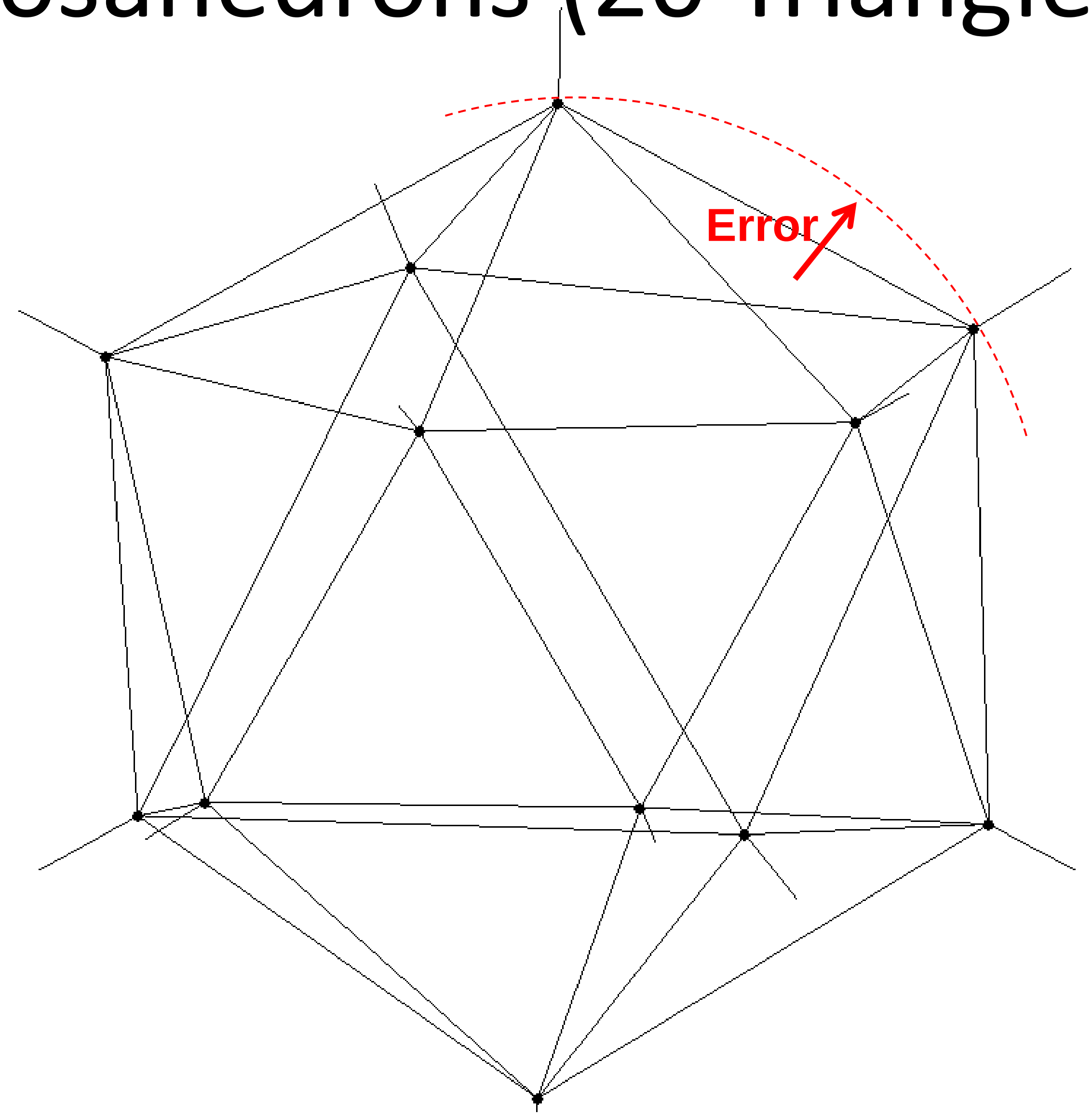




Icosahedrons (20 Triangles)

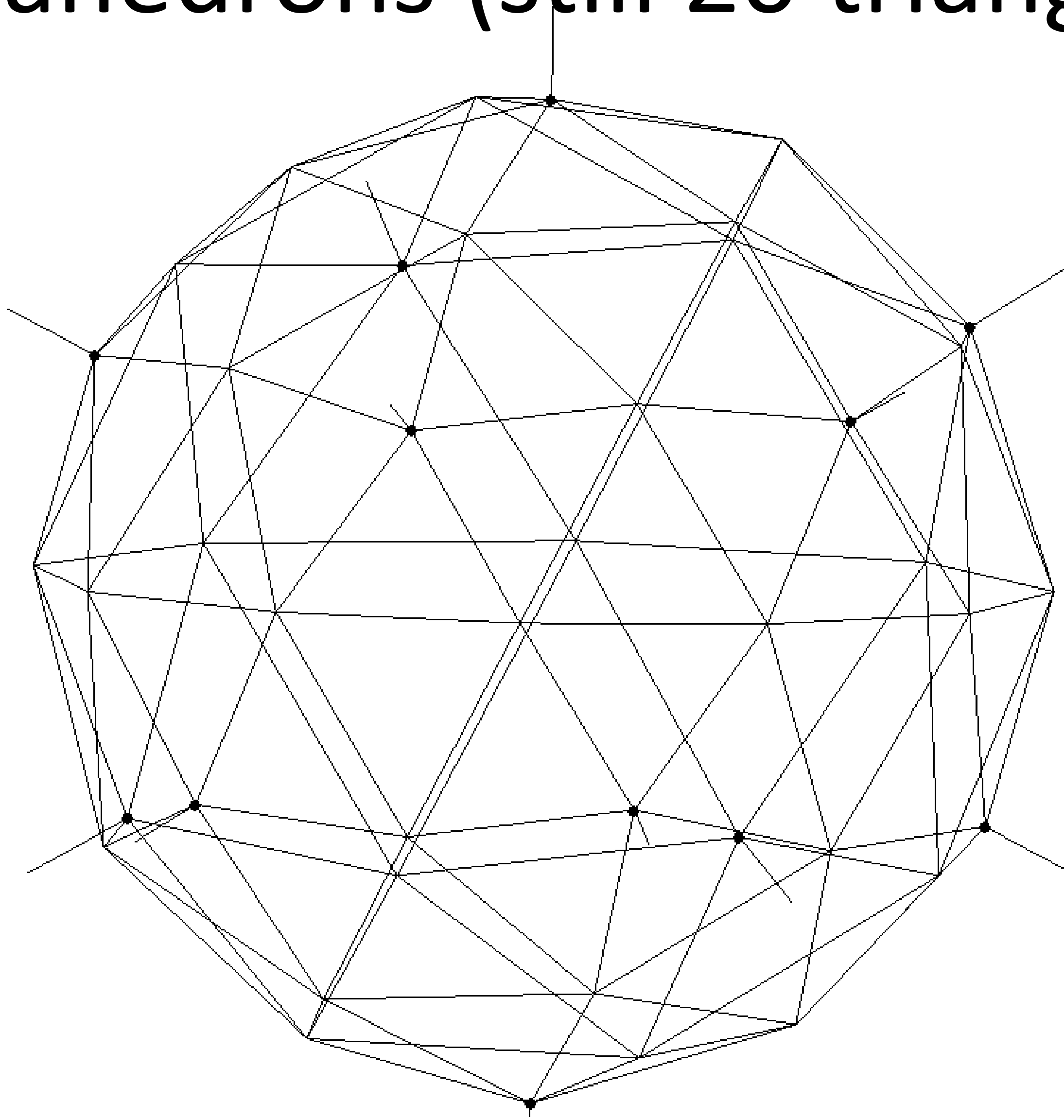


Icosahedrons (20 Triangles)



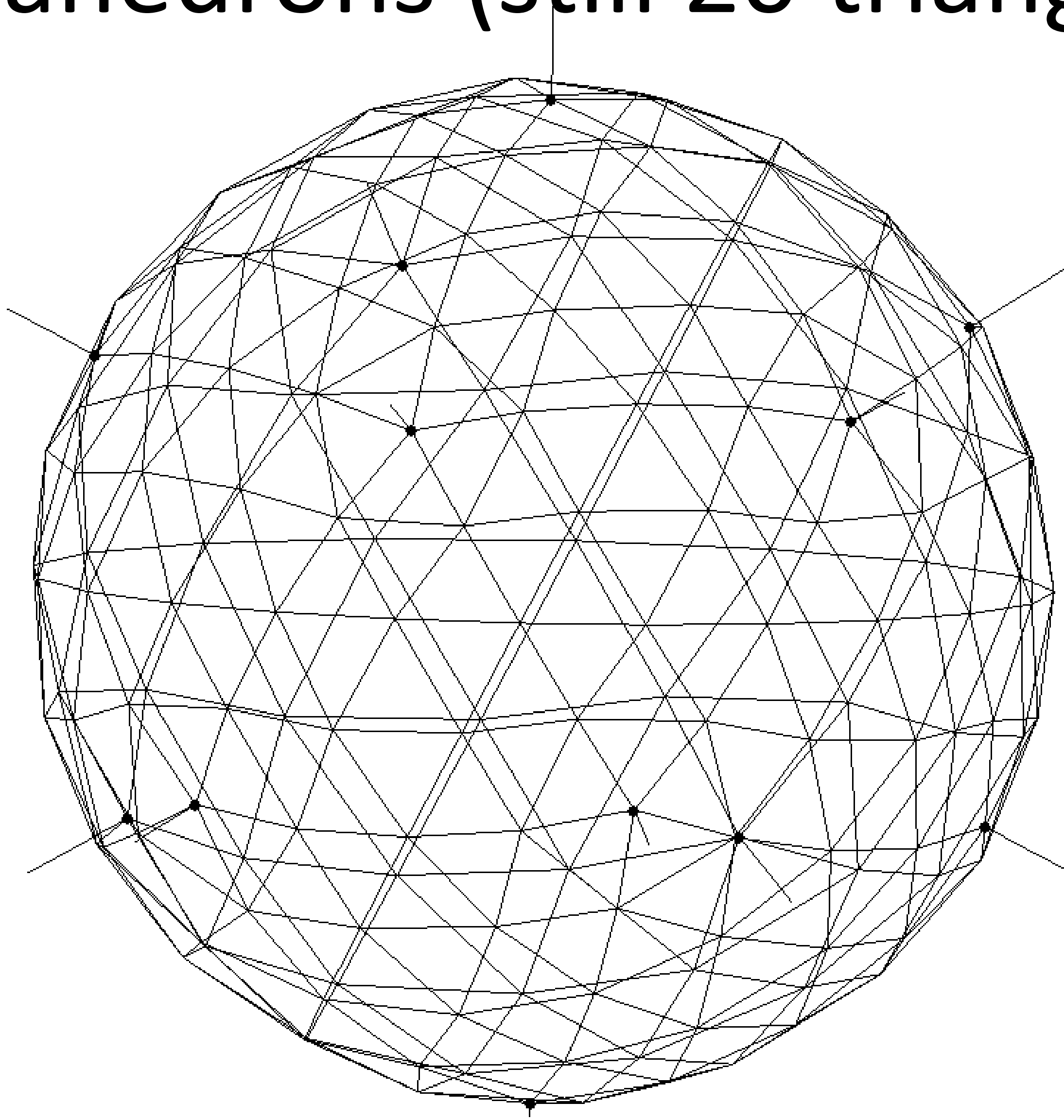
Flat triangles, error = 10.26% of diameter

Icosahedrons (still 20 triangles)



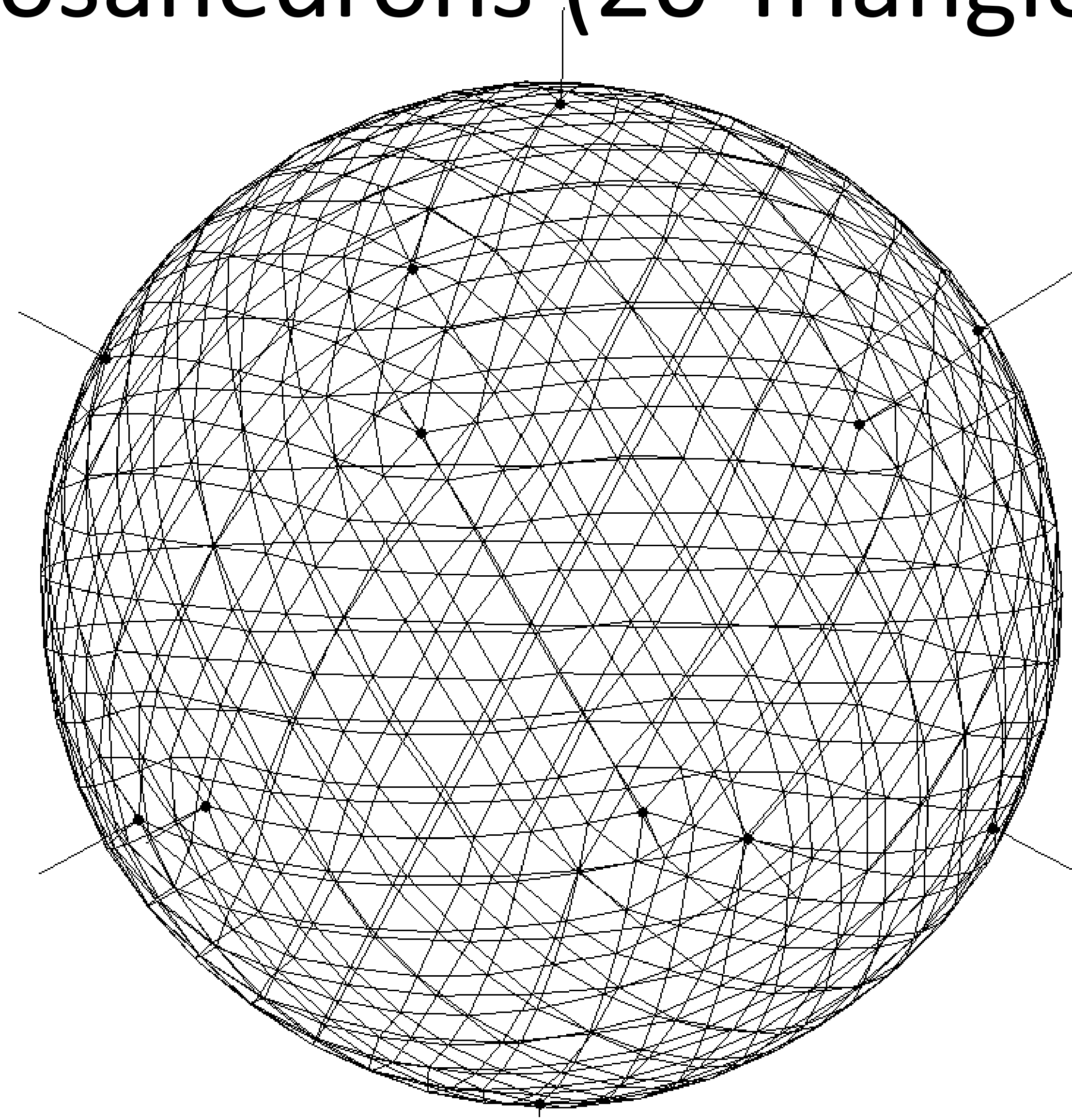
One subdivision, error = 3.81%

Icosahedrons (still 20 triangles)



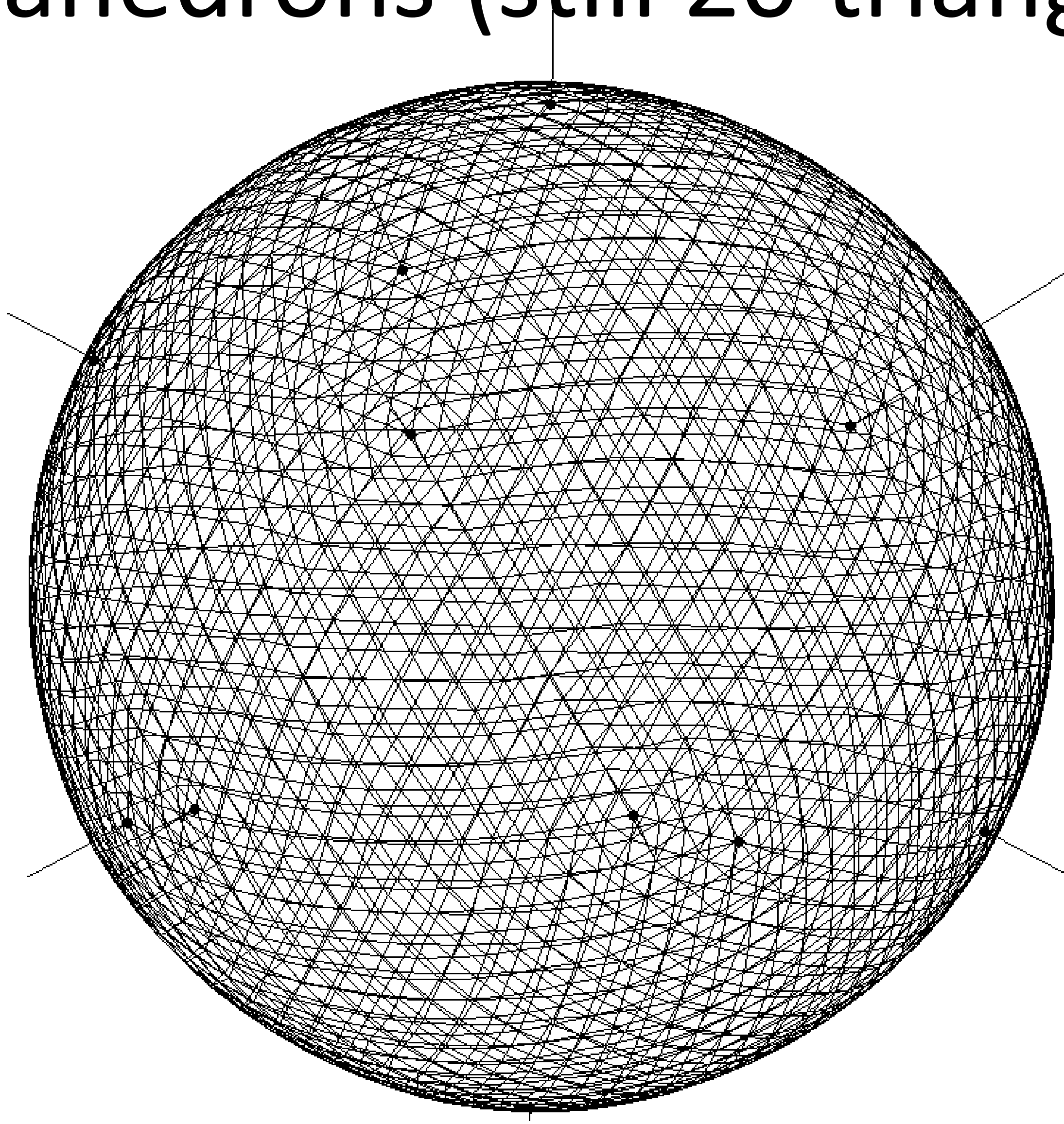
Two-fold subdivisions, error = 1.49%

Icosahedrons (20 Triangles)



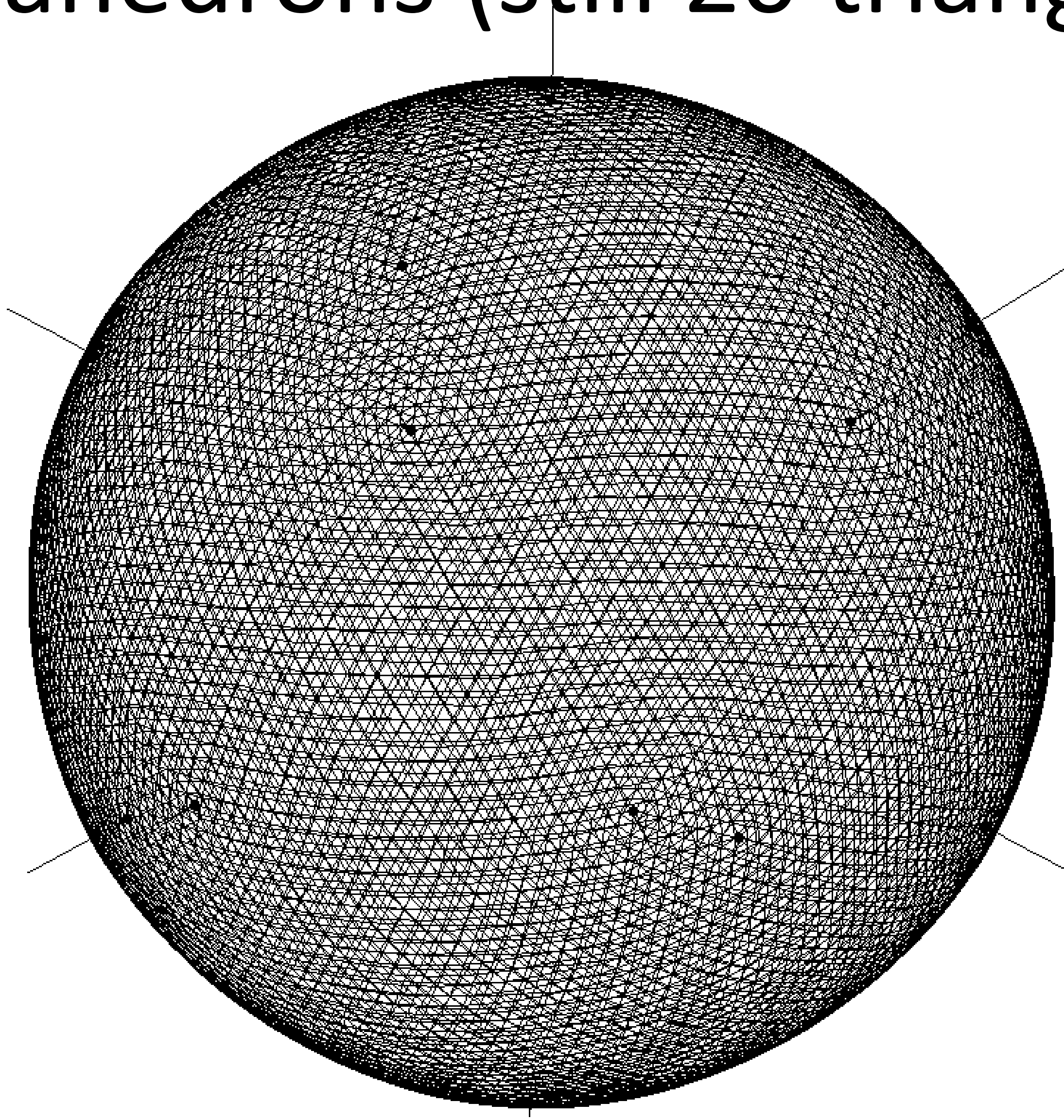
Three-fold subdivisions, error = 0.84%

Icosahedrons (still 20 triangles)



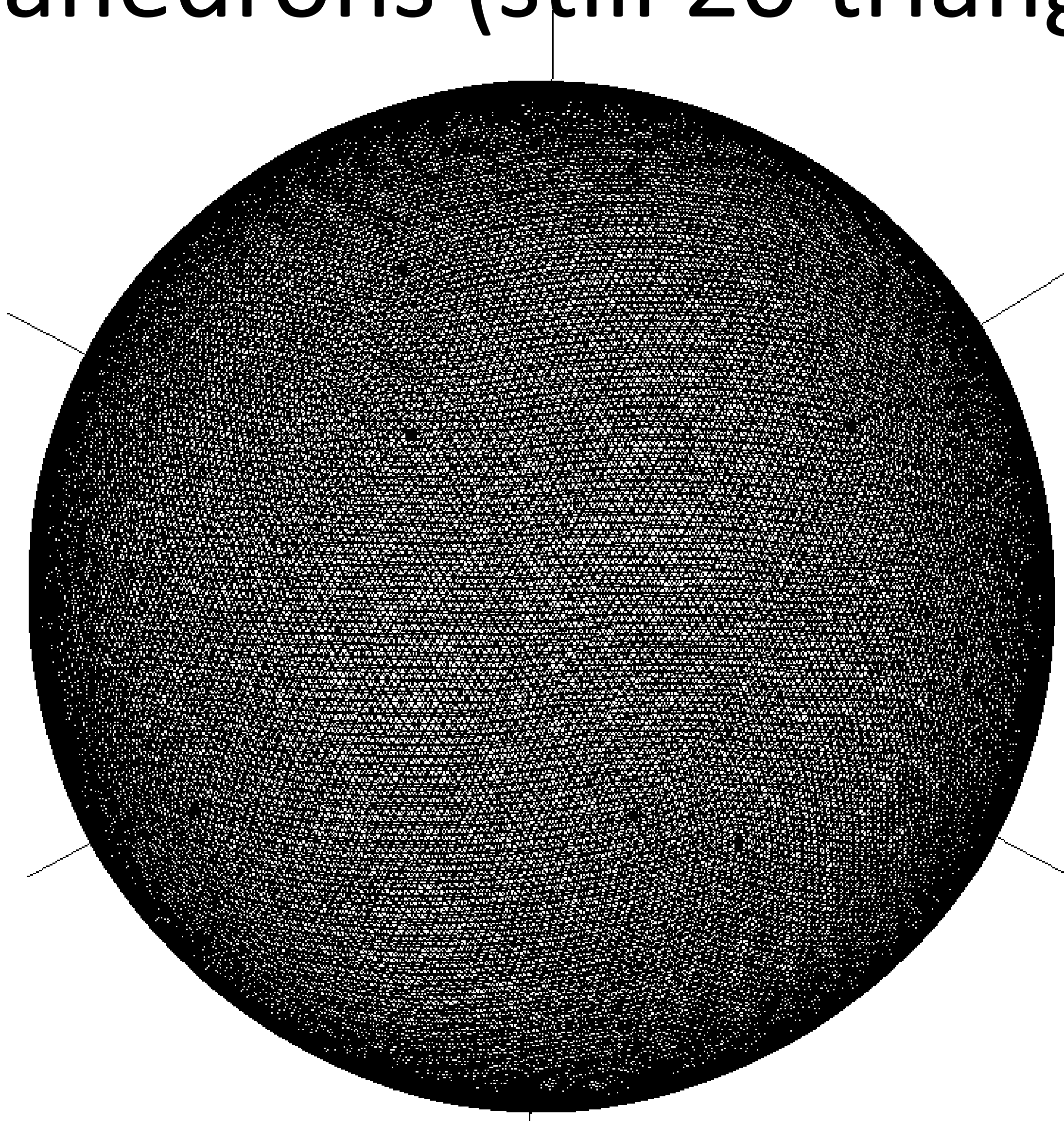
Four-fold subdivisions, error = 0.67%

Icosahedrons (still 20 triangles)



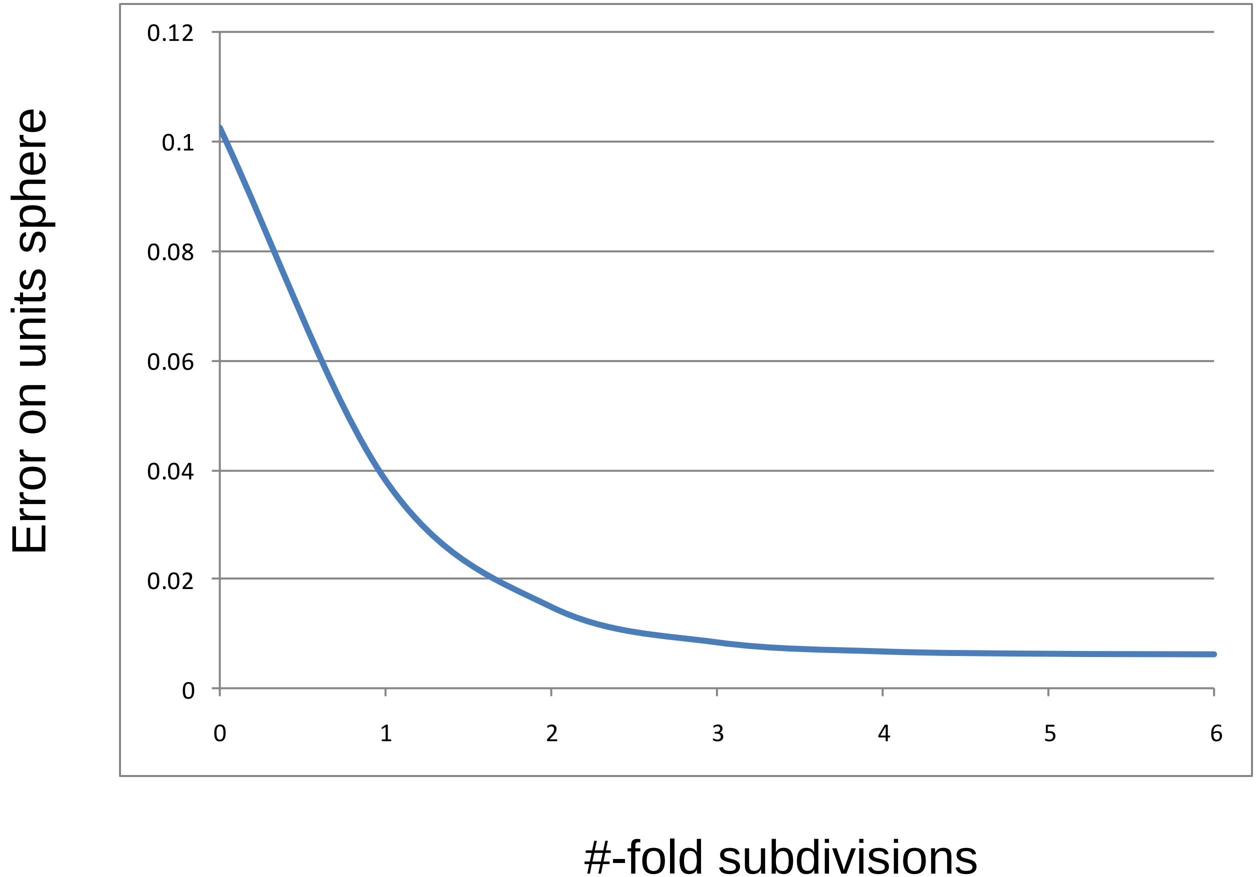
Five-fold subdivisions, error = 0.635%

Icosahedrons (still 20 triangles)

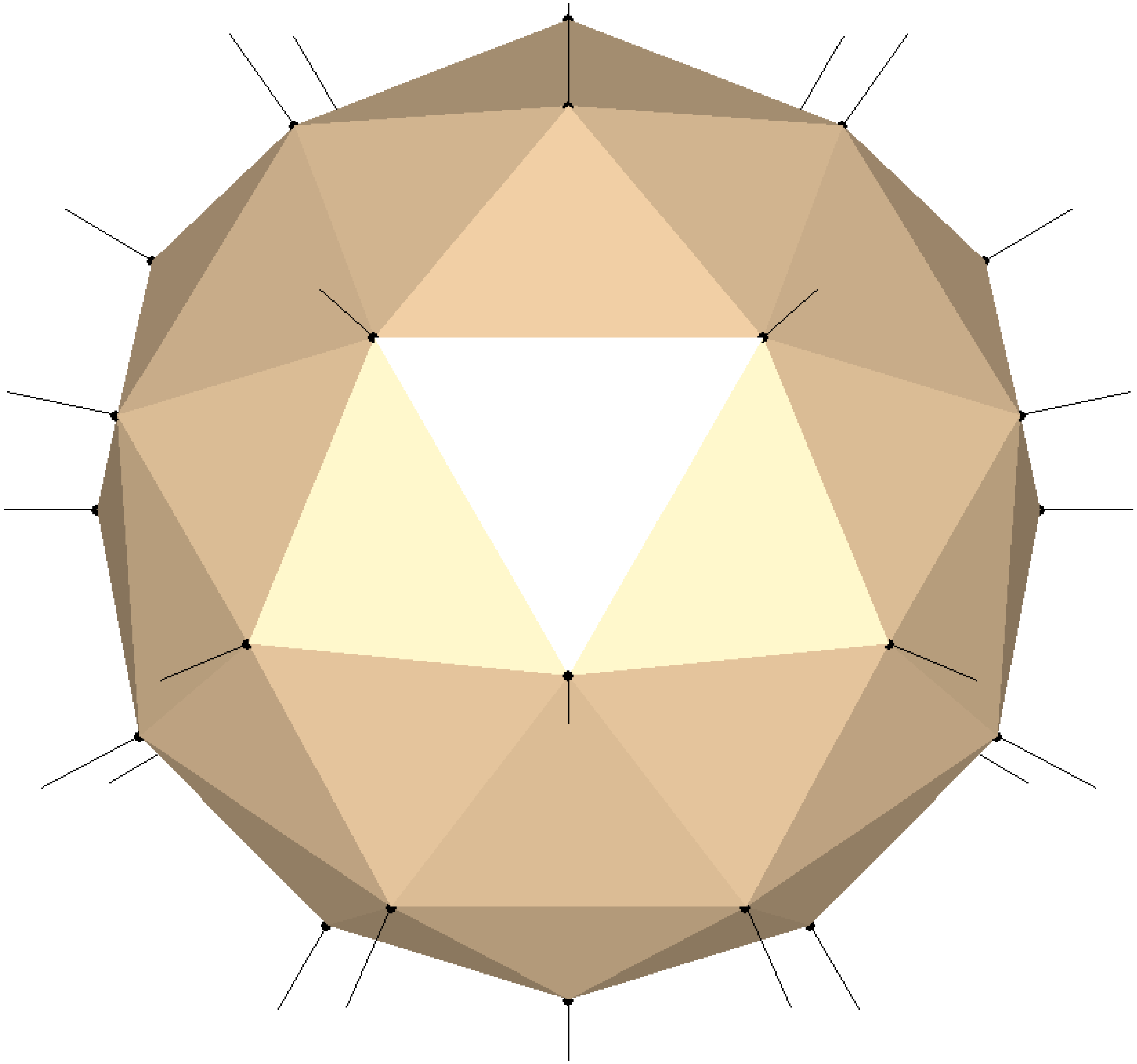


Six-fold subdivisions, error = 0.625%

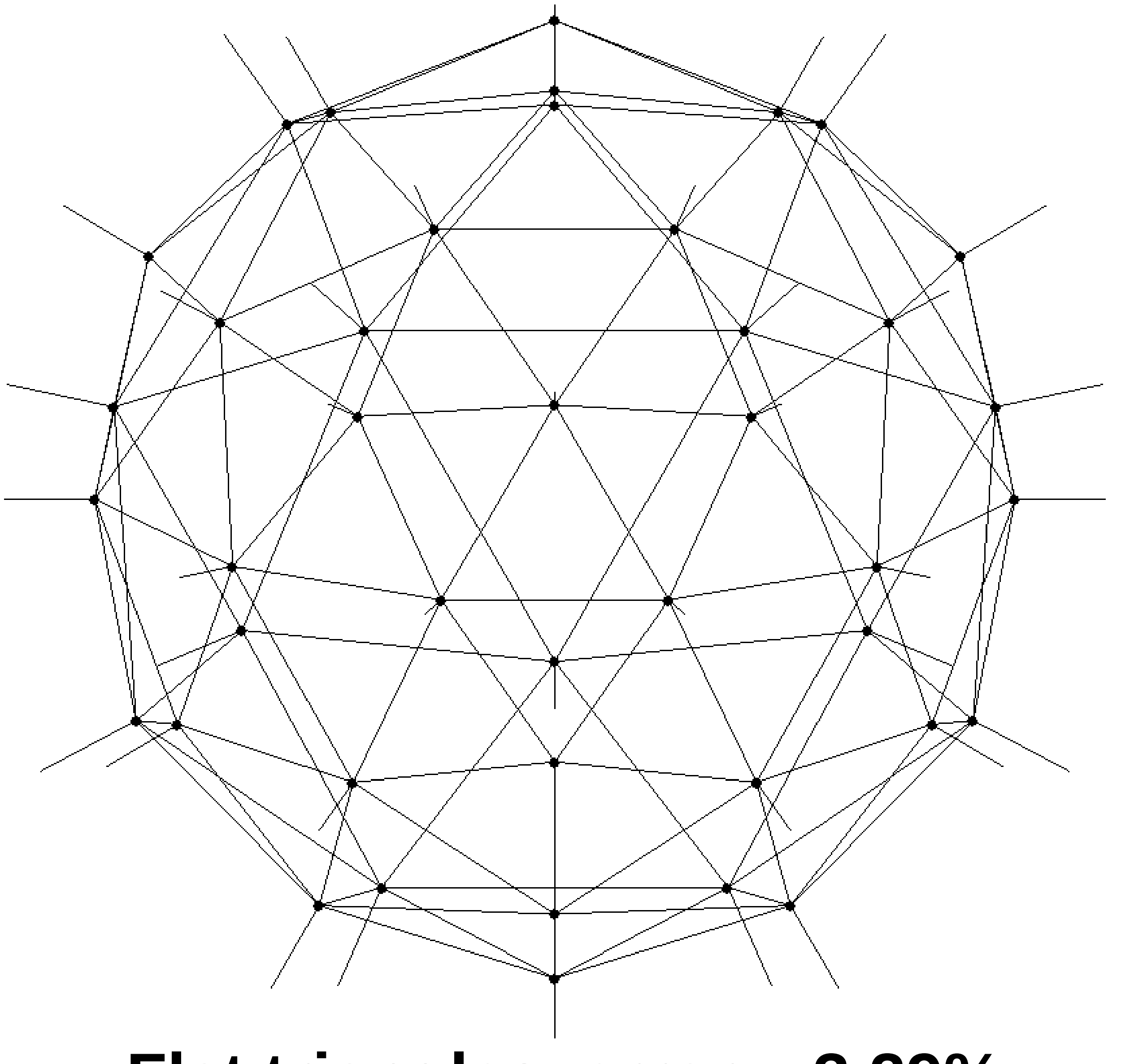
Curving the triangle patches using surface normal reduces error



Double Icosahedrons (n=80)

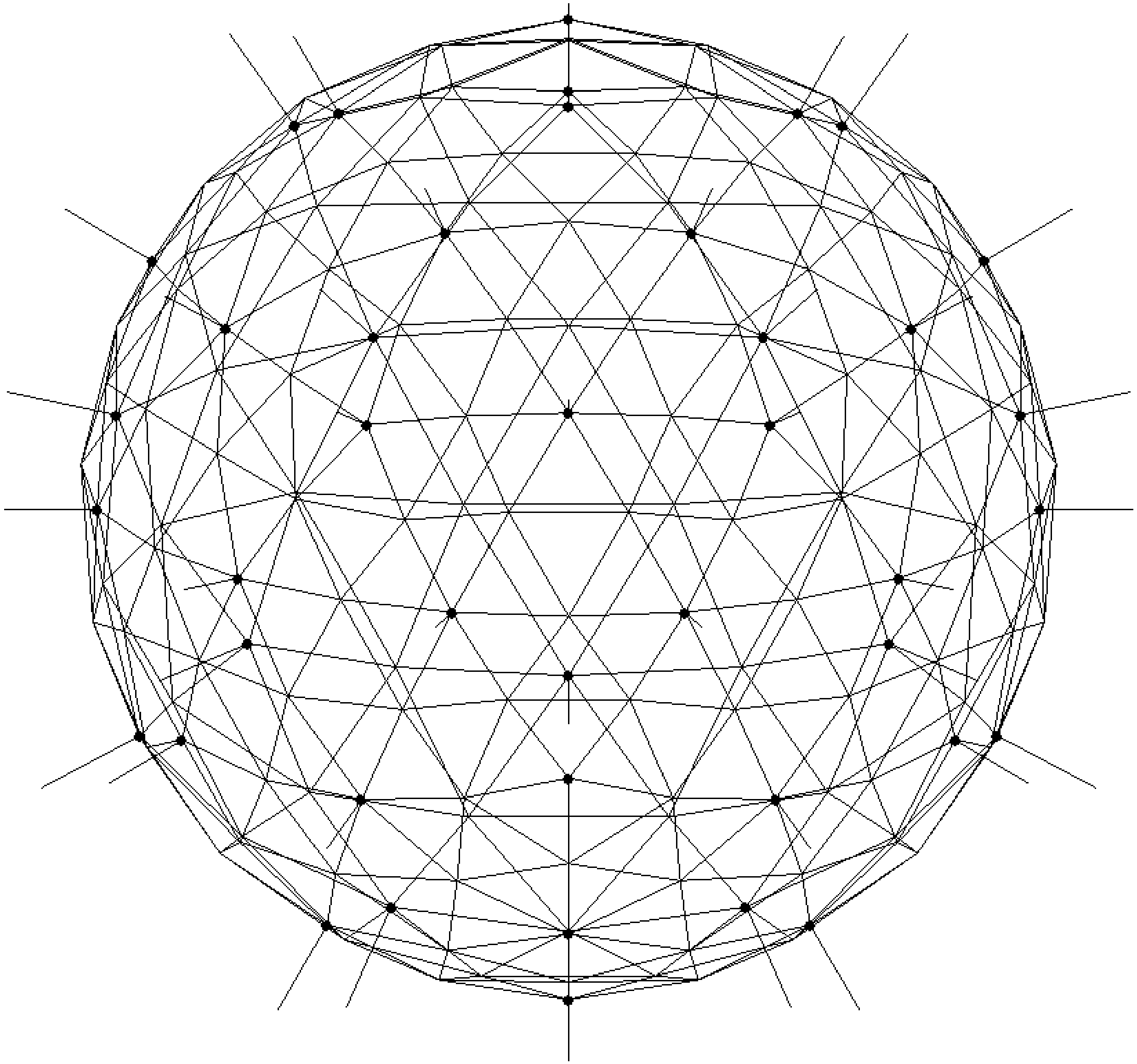


Double Icosahedrons (n=80)



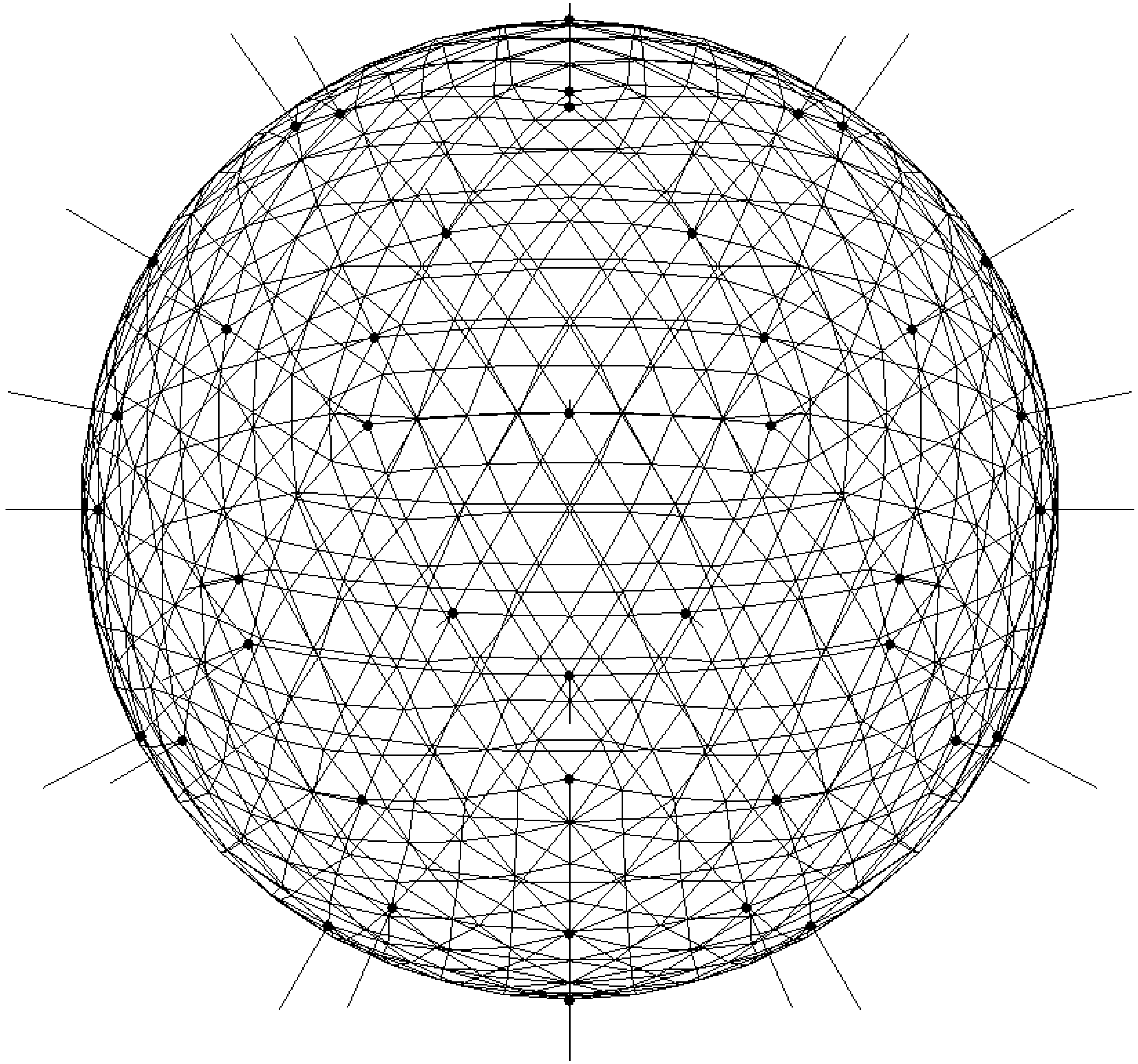
Flat triangles, error = 3.29%

Double Icosahedrons (n=80)



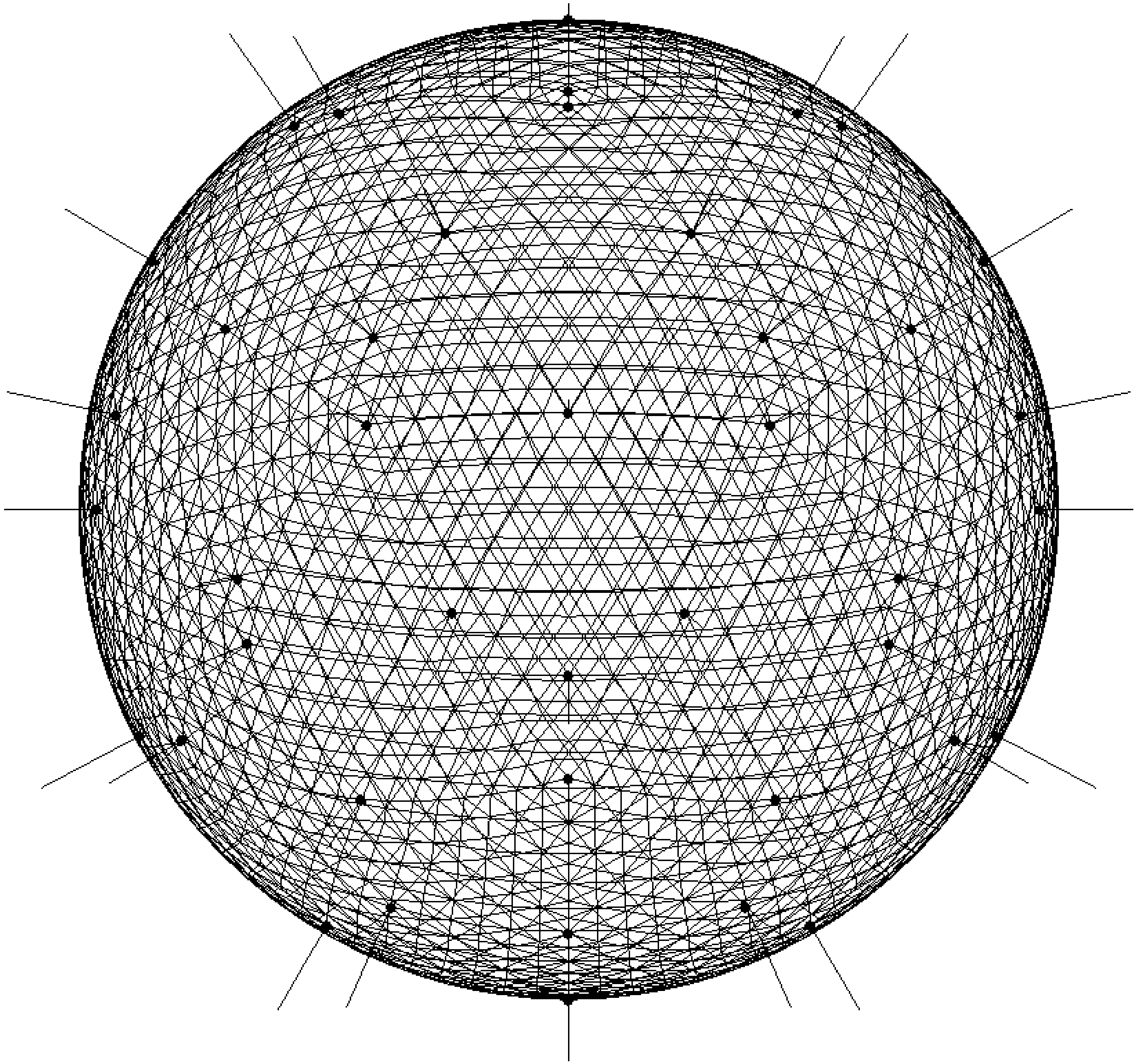
One subdivision, error = 0.946%

Double Icosahedrons (n=80)



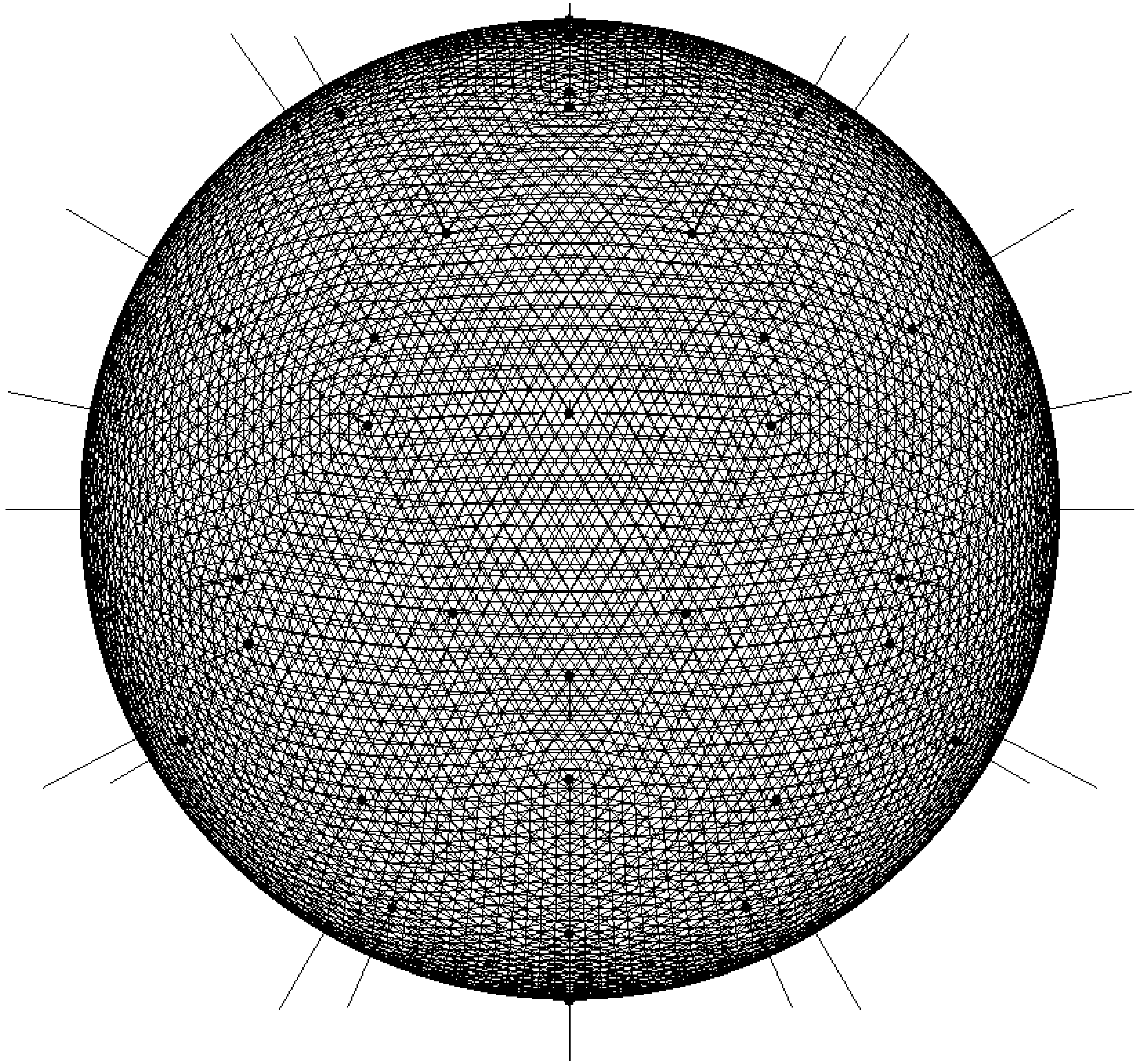
Two-fold subdivision, error = 0.290%

Double Icosahedrons (n=80)



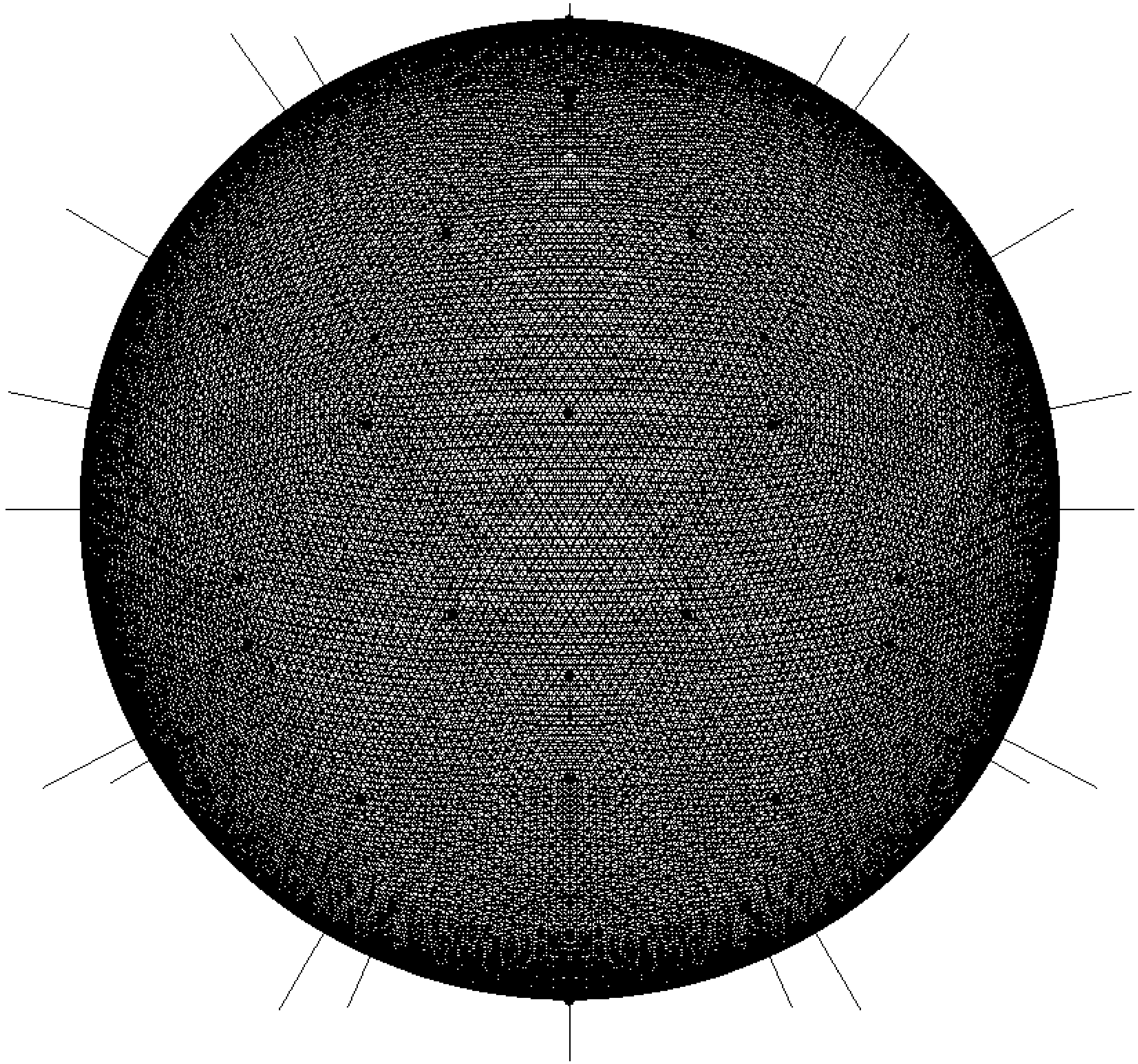
Three-fold subdivision, error = 0.121%

Double Icosahedrons (n=80)



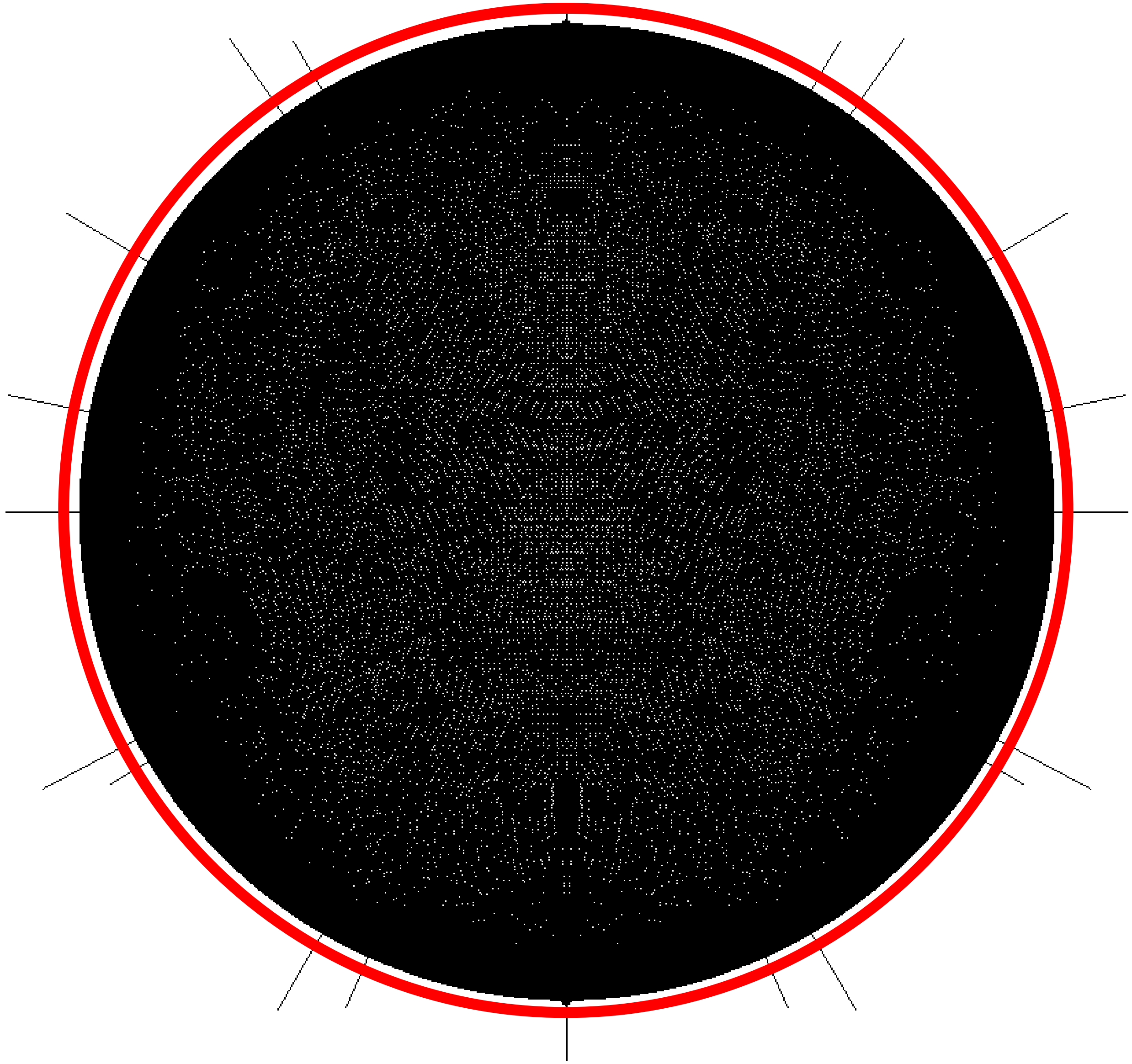
Four-fold subdivision, error = 0.079%

Double Icosahedrons (n=80)



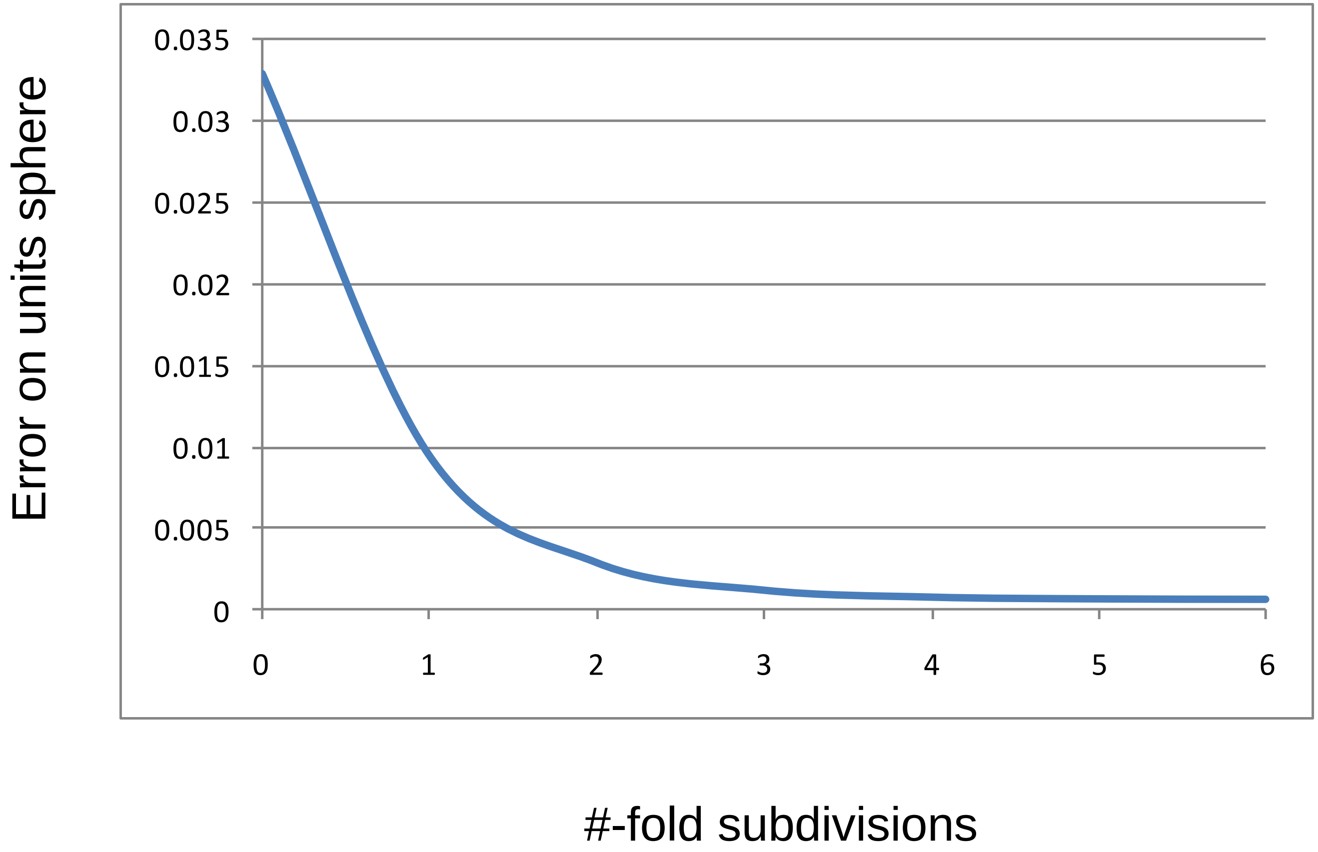
Five-fold subdivision, error = 0.068%

Double Icosahedrons (n=80)

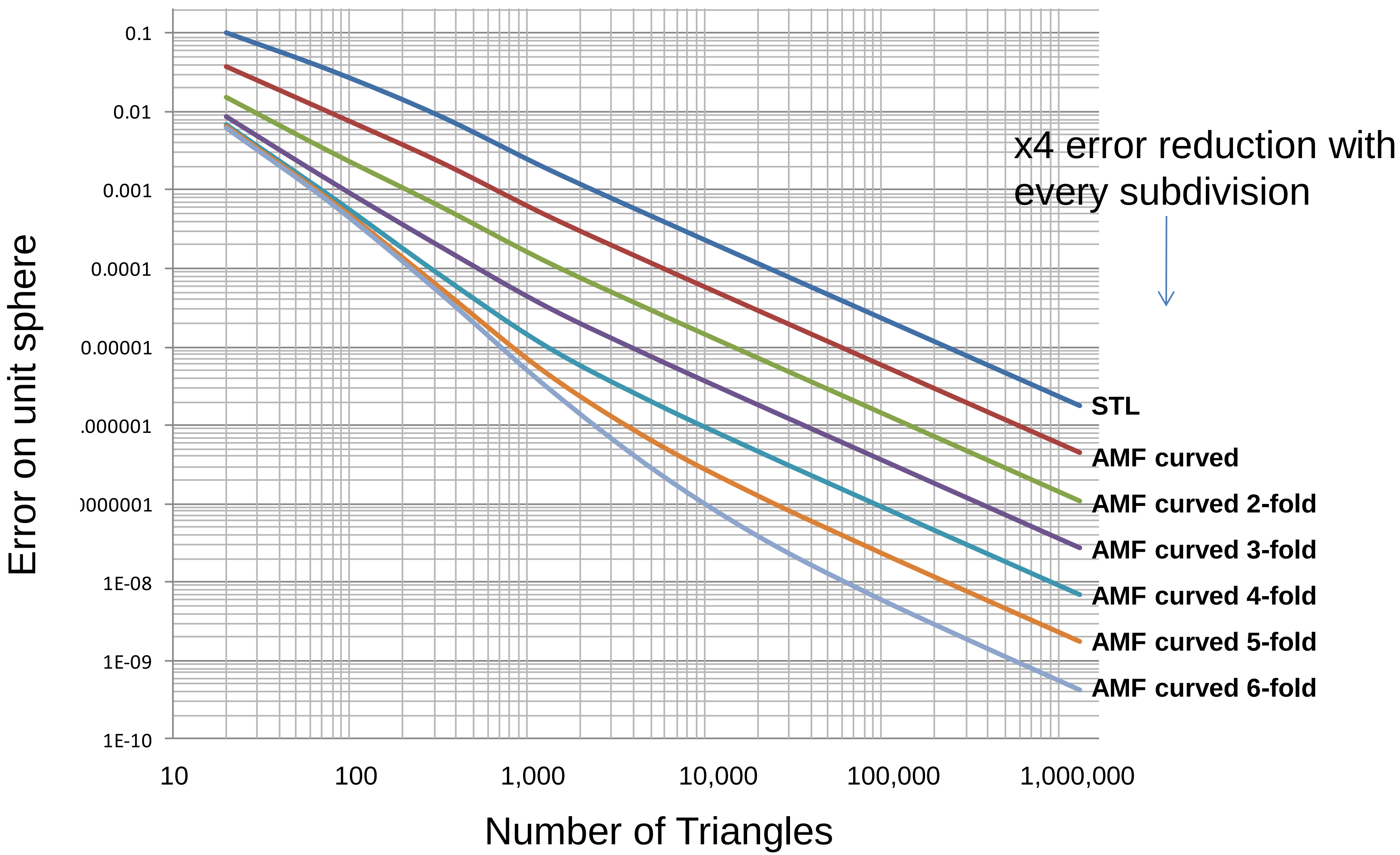


Six-fold subdivision, error = 0.065%

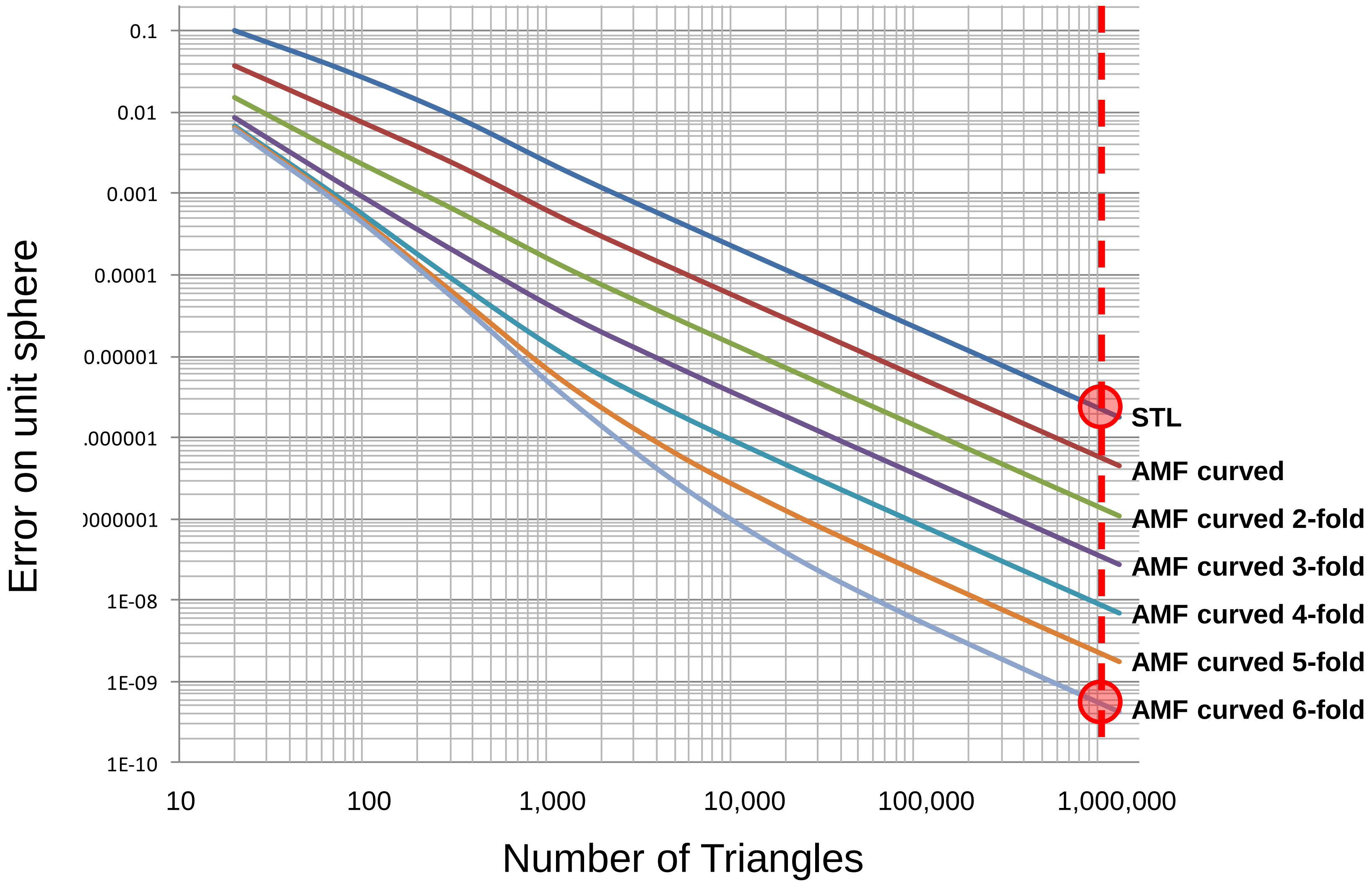
Curving the triangle patches using surface normal reduces error



Curved Triangles

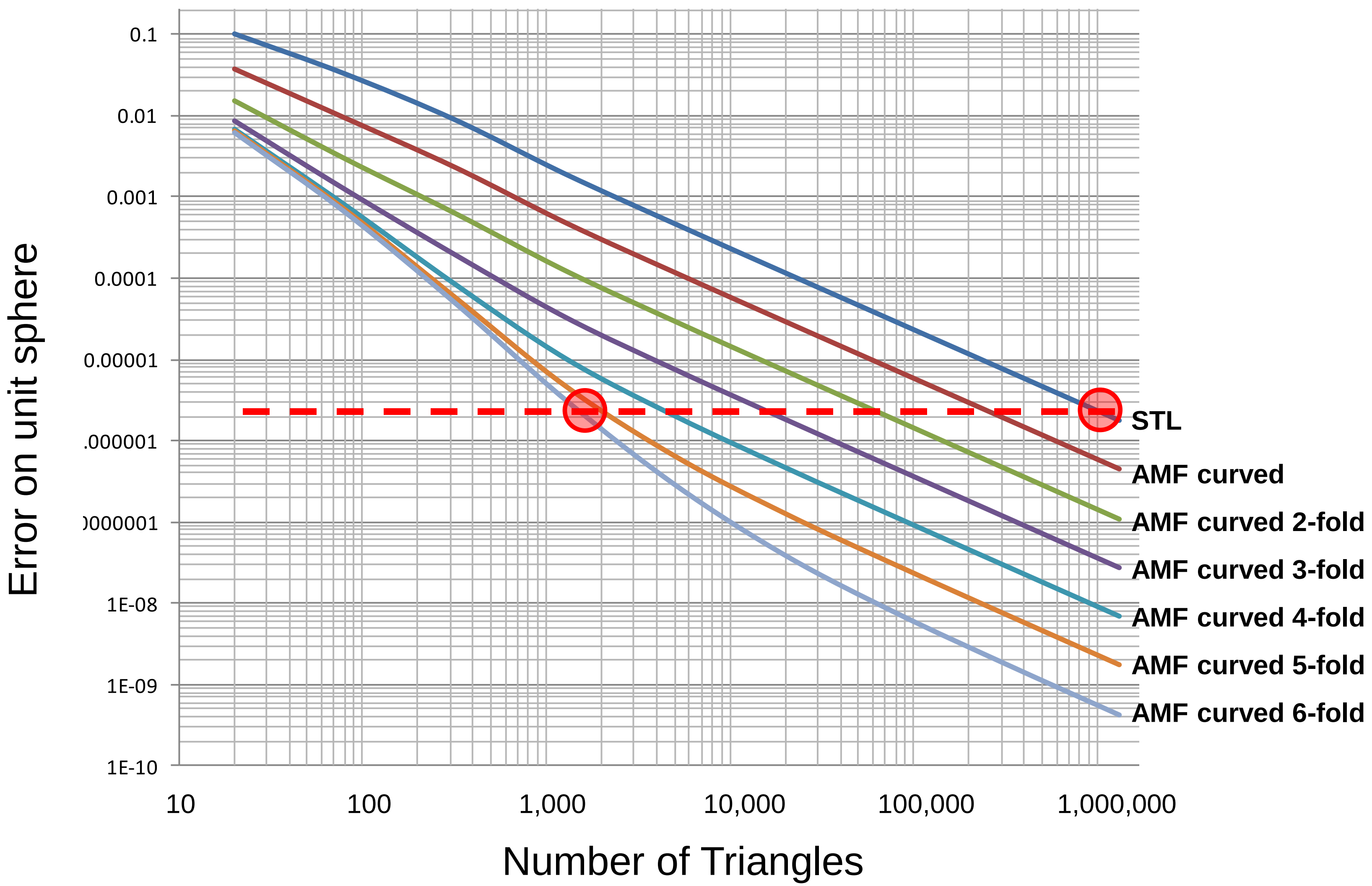


Curved Triangles



Three orders of magnitude improvement in accuracy for same number of triangles

Curved Triangles



Three orders of magnitude reduction in number of triangles for same accuracy

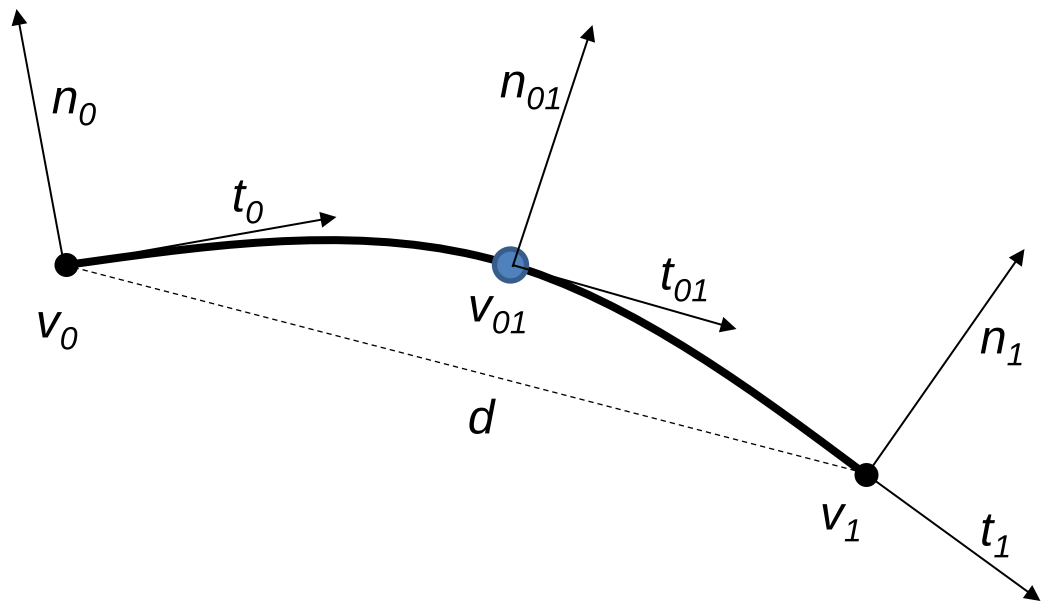
Accuracy

Number of Triangles	STL	AMF (with normals)
20	0.102673	0.006777
80	0.032914	0.000788
320	0.008877	8.28E-05
1,280	0.001893	1.01E-05
5,120	0.000455	1.95E-06
20,480	1.13E-04	4.51E-07
81,920	2.81E-05	1.11E-07
327,680	7.03E-06	2.75E-08
1,310,720	1.76E-06	6.87E-09

Examples

- Fabricate 10cm diameter sphere
    - with 10µm Precision
  - STL: 20,480 Flat Triangles
    - 500K Compressed Binary STL
  - AMF: 320 Curved Triangles
    - 10K Compressed AMF
- Fabricate 1m Sphere with 1nm precision
  - AMF: 1M Triangles
  - STL: !?

Simple to implement



1. If tangents t_0 or t_1 not specified, compute tangents from normals

$$t_0 = |d| \frac{(n_0 \times d) \times n_0}{\|(n_0 \times d) \times n_0\|}, \quad t_1 = |d| \frac{(n_1 \times d) \times n_1}{\|(n_1 \times d) \times n_1\|}$$

2. Compute center point $v_{01}=h(0.5)$ and center tangent t_{01} using Hermite curve

$$h(s)=(2s^3-3s^2+1)v_0+(s^3-2s^2+s)t_0+(-2s^3+3s^2)v_1+(s^3-s^2)t_1$$

3. Repeat for three triangle edges, then split triangle into four
4. Recurse as much as possible (diminishing returns after ~4 levels)
5. No ambiguities. Detailed procedure in specification.

Multiple Materials

```
<?xml version="1.0" encoding="UTF-8"?>
<amf units="mm">
  <material id="1">
    <metadata type="Name">StiffMaterial</metadata>
  </material>
  <material id="2">
    <metadata type="Name">FlexibleMaterial</metadata>
  </material>
  <material id="3">
    <metadata type="Name">MediumMaterial</metadata>
    <composite materialid="1">0.4</composite>
    <composite materialid="2">0.6</composite>
  </material>
  <material id="4">
    <metadata type="Name">VerticallyGraded</metadata>
    <composite materialid="1">z</composite>
    <composite materialid="2">10-z</composite>
  </material>
  <material id="5">
    <metadata type="Name">Checkerboard</metadata >
    <composite materialid="1">
          floor(x+y+z%1)+0.5) </composite>
    <composite materialid="2">
          1-floor(x+y+z%1)+0.5) </composite>
  </material>
  <object id="0">
    <mesh>
      <vertices>
        ...
      </vertices>
      <region materialid="1">
        ...
      </region>
      <region materialid="2">
        ...
      </region>
    </mesh>
  </object>
</amf>
```

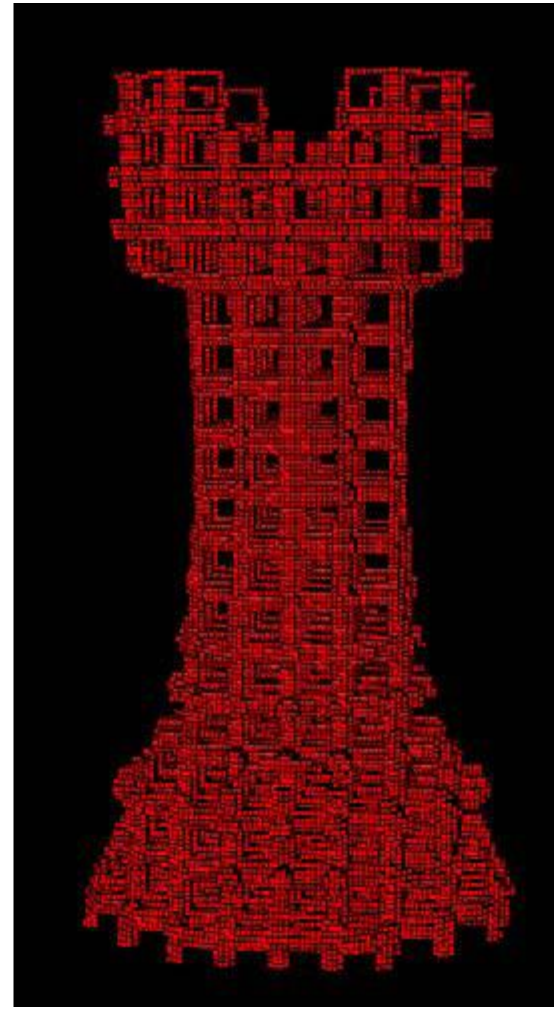
Graded Materials

```
<?xml version="1.0" encoding="UTF-8"?>
<amf units="mm">
  <material id="1">
    <metadata type="Name">StiffMaterial</metadata>
  </material>
  <material id="2">
    <metadata type="Name">FlexibleMaterial</metadata>
  </material>
  <material id="3">
    <metadata type="Name">MediumMaterial</metadata>
    <composite materialid="1">0.4</composite>
    <composite materialid="2">0.6</composite>
  </material>
  <material id="4">
    <metadata type="Name">VerticallyGraded</metadata>
    <composite materialid="1">z</composite>
    <composite materialid="2">10-z</composite>
  </material>
  <material id="5">
    <metadata type="Name">Checkerboard</metadata >
    <composite materialid="1">
          floor(x+y+z%1)+0.5) </composite>
    <composite materialid="2">
          1-floor(x+y+z%1)+0.5) </composite>
  </material>
  <object id="0">
    <mesh>
      <vertices>
        ...
      </vertices>
      <region materialid="1">
        ...
      </region>
      <region materialid="2">
        ...
      </region>
    </mesh>
  </object>
</amf>
```

Microstructure

```
<?xml version="1.0" encoding="UTF-8"?>
<amf units="mm">
  <material id="1">
    <metadata type="Name">StiffMaterial</metadata>
  </material>
  <material id="2">
    <metadata type="Name">FlexibleMaterial</metadata>
  </material>
  <material id="3">
    <metadata type="Name">MediumMaterial</metadata>
    <composite materialid="1">0.4</composite>
    <composite materialid="2">0.6</composite>
  </material>
  <material id="4">
    <metadata type="Name">VerticallyGraded</metadata>
    <composite materialid="1">z</composite>
    <composite materialid="2">10-z</composite>
  </material>
  <material id="5">
    <metadata type="Name">Checkerboard</metadata >
    <composite materialid="1">
          floor(x+y+z%1)+0.5) </composite>
    <composite materialid="2">
          1-floor(x+y+z%1)+0.5) </composite>
  </material>
  <object id="0">
    <mesh>
      <vertices>
        ...
      </vertices>
      <region materialid="1">
        ...
      </region>
      <region materialid="2">
        ...
      </region>
    </mesh>
  </object>
</amf>
```

Can also reference a texture map



Precedence	Operator	Description
1	()	Parentheses block
2	^	Power
3	*	Multiply
3	/	Divide
3	%	Modulus
4	+	Add
4	-	Subtract
5	=	Equal
5	<, <=	Less than (or equal to)
5	>, >=	Greater than (or equal to)
6	&	Intersection (Logical AND)
6	|	Union (Logical OR)
6	\	Difference (Logical XOR)
6	~	Negation (Logical NOT)

Periodic functions can be used to describe linear and nonlinear lattice materials

Material properties

- By name
  - <metatdata type="Name"> **ABS** </metadata>
  - <metatdata type="Name"> **Nylon 1234**</metadata>
- By physical property
  - <metatdata type="Elastic Modulus"> **2GPa**</metadata>

Color and Graphics

- Can be assigned to
  - A material
  - A region
  - A vertex
- Specified
  - Fixed RGBA values
  - By formula
  - By reference to an image

Print Constellation

- Print orientation
- Duplicated objects
- Sets of different objects
- Efficient packing
- Hierarchical

Metadata

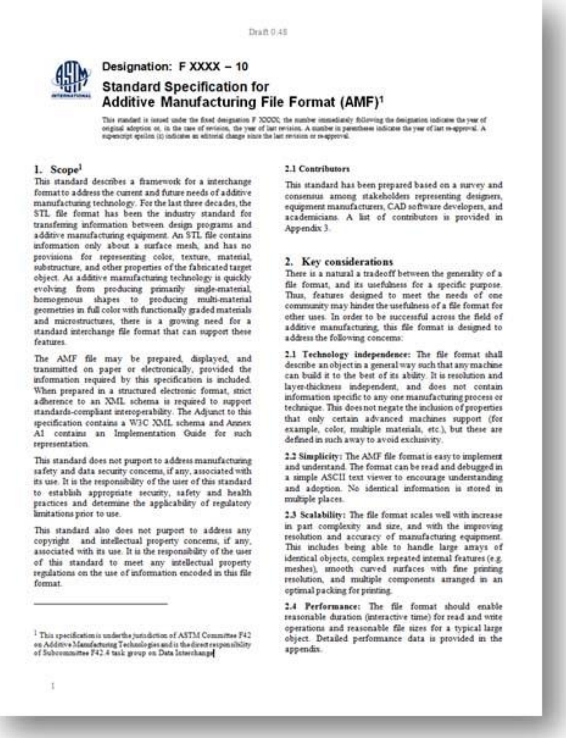
```xml
<metadata type=“Author”>John Doe”></metadata>
<metadata type=“Software”>SolidX 2.3”></metadata>
<metadata type=“Name”>Product 1></metadata>
<metadata type=“Revision”>12A”></metadata>

<object id=“1”>
		<metadata type=“Name”>Part A ></metadata>
</object id=“1”>
```

Future plans

- Tolerances
- Surface/depth textures
- Data encryption, copyright
- External references and subassemblies
- Process control
- Non-volumetric support structures
- Non mesh geometry specification methods
  - Voxel, FRep

Current Status

- AMF approved May 2011 as ASTM F2915
- Revision 1.1 in 2012
- Now: The test of adoption



Draft 0.45

Designation: F XXXX – 10

Standard Specification for Additive Manufacturing File Format (AMF)[1]

This standard is issued under the fixed designation F XXXX; the number immediately following the designation indicates the year of original adoption or, in the case of revision, the year of last revision. A number in parentheses indicates the year of last re-approval. A superscript epsilon (ε) indicates an editorial change since the last revision or re-approval.

1. Scope[1]

This standard describes a framework for a interchange format to address the current and future needs of additive manufacturing technology. For the last three decades, the STL file format has been the industry standard for transferring information between design programs and additive manufacturing equipment. An STL file contains information only about a surface mesh, and has no provisions for representing color, texture, material, substructure, and other properties of the fabricated target object. As additive manufacturing technology is quickly evolving from producing primarily single-material, homogenous shapes to producing multi-material geometries in full color with functionally graded materials and microstructures, there is a growing need for a standard interchange file format that can support these features.

The AMF file may be prepared, displayed, and transmitted on paper or electronically, provided the information required by this specification is included. When prepared in a structured electronic format, strict adherence to an XML schema is required to support standards-compliant interoperability. The Adjunct to this specification contains a W3C XML schema and Annex A1 contains an Implementation Guide for such representation.

This standard does not purport to address manufacturing safety and data security concerns, if any, associated with its use. It is the responsibility of the user of this standard to establish appropriate security, safety and health practices and determine the applicability of regulatory limitations prior to use.

This standard also does not purport to address any copyright and intellectual property concerns, if any, associated with its use. It is the responsibility of the user of this standard to meet any intellectual property regulations on the use of information encoded in this file format.

[1] This specification is under the jurisdiction of ASTM Committee F42 on Additive Manufacturing Technologies and is the direct responsibility of Subcommittee F42.4 task group on Data Interchange.

2.1 Contributors

This standard has been prepared based on a survey and consensus among stakeholders representing designers, equipment manufacturers, CAD software developers, and academicians. A list of contributors is provided in Appendix 3.

2. Key considerations

There is a natural a tradeoff between the generality of a file format, and its usefulness for a specific purpose. Thus, features designed to meet the needs of one community may hinder the usefulness of a file format for other uses. In order to be successful across the field of additive manufacturing, this file format is designed to address the following concerns:

2.1 Technology independence: The file format shall describe an object in a general way such that any machine can build it to the best of its ability. It is resolution and layer-thickness independent, and does not contain information specific to any one manufacturing process or technique. This does not negate the inclusion of properties that only certain advanced machines support (for example, color, multiple materials, etc.), but these are defined in such away to avoid exclusivity.

2.2 Simplicity: The AMF file format is easy to implement and understand. The format can be read and debugged in a simple ASCII text viewer to encourage understanding and adoption. No identical information is stored in multiple places.

2.3 Scalability: The file format scales well with increase in part complexity and size, and with the improving resolution and accuracy of manufacturing equipment. This includes being able to handle large arrays of identical objects, complex repeated internal features (e.g. meshes), smooth curved surfaces with fine printing resolution, and multiple components arranged in an optimal packing for printing.

2.4 Performance: The file format should enable reasonable duration (interactive time) for read and write operations and reasonable file sizes for a typical large object. Detailed performance data is provided in the appendix.

1

More Information

Designation: F XXXX – 10

Standard Specification for Additive Manufacturing File Format (AMF)[1]

This standard is issued under the fixed designation F XXXX; the number immediately following the designation indicates the year of original adoption or, in the case of revision, the year of last revision. A number in parentheses indicates the year of last re-approval. A superscript epsilon (ε) indicates an editorial change since the last revision or re-approval.

1. Scope[1]

For the last three decades, the STL file format has been the industry standard for transferring information between design programs and additive manufacturing equipment. As additive manufacturing technology is quickly evolving from producing primarily single-material, homogenous shapes to producing multi-material geometries in full color with functionally graded materials and microstructures, there is a growing need for a standard interchange file format that can support these features. An STL file contains information only about a surface mesh, and has no provisions for representing color, texture, material, substructure, and other properties of the fabricated target object. This standard describes a framework for a interchange format to address the current and future needs of additive manufacturing technology.

The AMF file may be prepared, displayed, and transmitted on paper or electronically, provided the information required by this specification is included. When prepared in a structured electronic format, strict adherence to an XML schema is required to support standards-compliant interoperability. The Adjunct to this specification contains a W3C XML schema and Annex A1 contains an Implementation Guide for such representation.

2.1 Contributors

This standard has been prepared based on a survey and consensus among stakeholders representing designers, equipment manufacturers, CAD software developers, and academicians. A list of contributors and supporters is provided in Appendix 2.

2. Key considerations

There is a naturally a tradeoff between the generality of a file format, and its usefulness for a specific purpose. Thus, features designed to meet the needs of one community may hinder the usefulness of a file format for other uses. In order to be successful across the field of additive manufacturing, this file format is designed to address the following concerns:

2.1 Technology independence: The file format shall describe an object in a general way such that any machine can build it to the best of its ability. It is resolution and layer-thickness independent, and does not contain information specific to any one manufacturing process or technique. This does not negate the inclusion of properties that only certain advanced machines support (for example, color, multiple materials, etc.), but these are defined in such away to avoid exclusivity.

ASTM INTERNATIONAL

Standards Worldwide - Home

- Standards
- Standards and Engineering Digital Library
- Books & Journals
- Technical Committees
- Membership
- Meetings & Symposia
- Training Courses
- Proficiency Testing
- Certification & Declarations
- Lab Directory
- About ASTM
- Magazine & Newsletters
- News Releases
- Students & Professors
- Global Cooperation

Product Information

- Get Product Updates
- Request a Free Catalog
- View Catalog

RSS News Feed

F42 on Additive Manufacturing Technologies

Technical Committees / Committee F42 /

Next: Meeting Symposium

Committee F42 on Additive Manufacturing Technologies

Staff Manager: Pat Picariello 610-832-9720

ASTM Committee F42 on Additive Manufacturing Technologies was formed in 2009. F42 meets twice a year, usually in January and July, with about 70 members attending two days of technical meetings. The Committee, with a current membership of approximately 100, has 3 technical subcommittees; all standards developed by F42 are published in the Annual Book of ASTM Standards, Volume 10.04 . Information on the F42 subcommittee structure, portfolio of approved standards, and Work Items under development, is available from the List of Subcommittees, Standards and Work Items below. These standards will play a preeminent role in all aspects of additive manufacturing technologies.

See all ASTM Intl. videos at YouTube

General Information

- F42 Scope
- Committee Officers and Staff Support
- Future Meetings
- Search Past ASTM Symposia

Get Involved

- Membership Information and Application
- New Member Orientation
- Invite a Colleague to Join

Standards Development

- List of Subcommittees, Standards and, Work Items
- Standards Development Tools

RSS News Feed

What's New

- >> Committee News
- >> Next Meeting
- >> Next Symposium

Committee-Sponsored Products

Additive Manufacturing File Format

From Wikipedia, the free encyclopedia

This article **relies largely or entirely upon a single source**. Please help improve this article by introducing appropriate citations to additional sources. *(June 2011)*

"AMF" redirects here. For other uses, see AMF (disambiguation).

Additive Manufacturing File Format (AMF) is an open standard for describing objects for layered manufacturing processes such as 3D Printing. The official ASTM F2915[1]standard is an XML-based format designed to allow any design software to describe the shape and composition of any 3D object to be fabricated on any 3D printer. Unlike its predecessor STL format, AMF has native support for color, materials, and constellations.

Additive Manufacturing Format

AMF icon

Filename extension	.amf
Internet media type	application/x-amf
Developed by	ASTM International
Initial release	May 2, 2011
Latest release	1.0
Standard(s)	ASTM F2915[1]

Contents [hide]

Technical Information

[edit]

This section provides an overview of the file format. A detailed specification is published by ASTM F2915[1] and may be revised from time to time.

An AMF can represent one object, or multiple objects arranged in a constellation. Each object is described as a set of non-overlapping volumes. Each volume is described by

Geometry specification

[edit]

The AMF format maintains the triangle-mesh geometry representation used in the STL format in order to take advantage of existing optimized slicing algorithm and code infrastructure already in existence. The top level `<object>` element specifies a unique id, and contains two child elements: `<vertices>` and `<volume>`. The `<object>` element can optionally specify a material. The required <vertices> element lists all vertices that are used in this object. Each vertex is implicitly assigned a number in the order in which it was declared, starting at zero. The required child element `<coordinates>` gives the position of the point in 3D space using the `<x>`, `<y>` and `<z>` elements. After the vertex information, at least one <volume> element must be included. Each volume encapsulates a closed volume of the object, Multiple volumes can be specified in a single object. Volumes may share vertices at interfaces but may not have any overlapping volume. Within each volume, the child element `<triangle>` is used to define triangles that tessellate the surface of the volume. Each `<triangle>` element will list three vertices from the set of indices of the previously defined vertices. The indices of the three vertices of the triangles are specified using the `<v1>`, `<v2>` and `<v3>` elements. The order of the vertices must be according to the right-hand rule, such that vertices are listed in counter-clockwise order as viewed from the outside. Each triangle is implicitly assigned a number in the order in which it was declared, starting at zero.

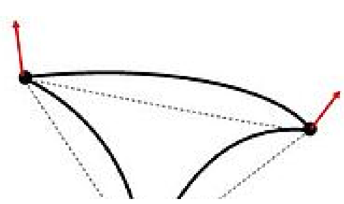
```
<?xml version="1.0" encoding="UTF-8"?>
<amf unit="millimeter">
  <object id="0">
    <mesh>
      <vertices>
        <vertex>
          <coordinates>
            <x>0</x>
            <y>1.32</y>
            <z>3.715</z>
          </coordinates>
        </vertex>
        ...
      </vertices>
      <volume>
        <triangle>
          <v1>0</v1>
          <v2>1</v2>
          <v3>3</v3>
        </triangle>
        ...
      </volume>
    </mesh>
  </object>
</amf>
```

Wikipedia provides an overview of key concepts

Refers to ASTM for formal spec

Curved triangles

[edit]



In order to improve geometric fidelity, the format allows curving the triangle patches. By default, all triangles are assumed to be flat and all triangle edges are assumed to be straight lines connecting their two vertices. However, curved triangles and curved edges can optionally be specified in order to reduce the number of mesh elements required to describe a curved surface. The curvature information has been shown to reduce the error of a spherical surface by a factor of 1000 as compared to a surface described by the same number of planar triangles[1]. Curvature should not create a deviation of from the plane of the flat triangle that exceeds 50% of the largest guest · Join · Help · Sign In · wikispaces

STL2.org

home | PAGE | DISCUSSION | HISTORY | NOTIFY ME | PROTECTED

- Join this Wiki
- Recent Changes
- Manage Wiki

Search

All Pages
- home
- AMF open-source software
- AMF test files
- STL to AMF converter

ASTM Additive Manufacturing File Format (AMF)

This wiki contains information and resources regarding the new AMF File format. AMF is an official ASTM standard that describes a framework for a data interchange to address the current and future needs of additive manufacturing technology. AMF is an XML-based format designed to allow any design software to describe the shape and composition of any object to be fabricated on any 3D-printer. This format has been developped by ASTM Committee *F42* on Additive Manufacturing Technologies, specifically the Task Group on File Formats.

On this wiki

- **Get the official** ASTM F2915 standard specification
- **Read the** unofficial draft of the AMF standard
- **Edit** the wikipedia AMF entry
- **Join the discussion forum** STL2.0 google group to discuss the AMF format and proposals for revisions.
- View the PowerPoint Presentaion on the new format.
- Browse and contribute AMF open-source software
- Browse and contribute AMF test files
- **Contact** the ASTM F42 Task group chair Hod Lipson by email or by phone.

Full open-source reference implementation

Sample files

Join The Discussion

http://groups.google.com/group/stl2

Google groups

STL 2.0

Se

Home **New since last time:** 2 messages

The STL 2.0 group is an ASTM-driven consortium interested in defining a new Additive-Manufacturing file format. The new format is to replace the current de-facto standard STL file format. This email group is open to anyone interested in shaping this specification.

[edit welcome message]

Discussions 9 of 157 messages view all » + new post

URLs / transformations / encryption
By Jacob Barhak - Jul 7 - 7 authors - 30 replies

AMF File format - Lattice and porous structures
By Hod Lipson - Jul 7 - 13 authors - 25 replies

"low level" data communication needed
By Turlif - Jul 3 - 1 author - 0 replies

new element layer thickness in AMF
By Steven Adler (A3DM) - Jul 2 - 4 authors - 7 replies

AMF tools: purging+certification
By Markus Hitter - Jul 2 - 7 authors - 9 replies

STL to AMF (v0.44) ported to mac and *nix
By Reinoud Zandijk - Jul 1 - 2 authors - 3 replies

Edge curvature in detail

Inside3Dprinting

April 22-23, NYC

HOME | REGISTER | SPEAKERS | AGENDA | SPONSORS | LOCATION | CONTACT US

Welcome

The 3D printing revolution has arrived and **Inside 3D Printing** is the first business to business conference and expo to delve into the present and future impact of 3D printing. Join us on April 22-23 in New York City, and immerse yourself in two full days of tutorials and seminars that will give you a blueprint for how to invest and utilize 3D printing in coming years. And, you will not want to miss the exhibition hall packed with the latest 3D printers and services. Be a part of this cutting-edge event!

REGISTER NOW

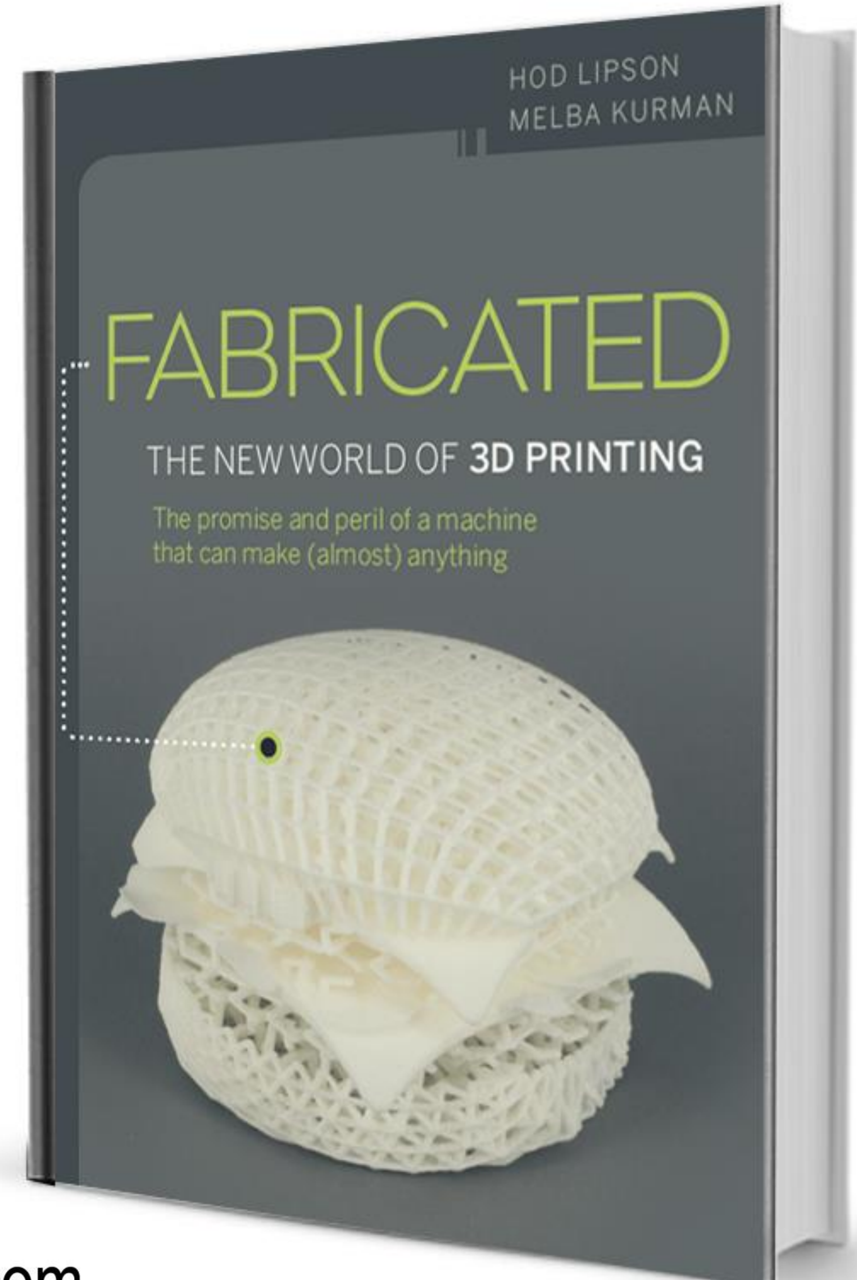
Keynote Presentations

Terry Wohlers, Principal Consultant and President, Wohlers Associates, Inc. has been named as one of the most influential individuals in rapid product development and additive manufacturing.

Peter Weijmarshausen, CEO and Co-Founder of Shapeways is recognized as a global expert on 3D printing, entrepreneurship, internet marketing, and business development.

View Agenda »

- Technologies
- New business models
- New design tools
- Bioprinting
- Food printing
- Education
- Law and safety
- Future directions



amazon.com