

# 3D Manufacturing Format

---

## コア仕様 & リファレンスガイド



バージョン	1.1
状態	公開済み

THESE MATERIALS ARE PROVIDED "AS IS." The contributors expressly disclaim any warranties (express, implied, or otherwise), including implied warranties of merchantability, non-infringement, fitness for a particular purpose, or title, related to the materials. The entire risk as to implementing or otherwise using the materials is assumed by the implementer and user. IN NO EVENT WILL ANY MEMBER BE LIABLE TO ANY OTHER PARTY FOR LOST PROFITS OR ANY FORM OF INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS DELIVERABLE OR ITS GOVERNING AGREEMENT, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND WHETHER OR NOT THE OTHER MEMBER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## 序文

この仕様について .....	4
て .....	4
表記規則 .....	5
言葉に関する覚書 .....	5
ソフトウェア適合 .....	6
性 .....	7
パート□. 3MF ドキュメン .....	7
ト .....	8
第 1 章. 3MF ドキュメント フォーマツ .....	9
ト .....	9
1.1. 本仕様の構成 .....	9
1.2. パッケージ .....	11
ジ .....	11
第 2 章. パーツとリレーションシッ .....	11
プ .....	12
2.1. 3D ペイロー .....	12
ド .....	13
2.1.1. 3D パーツとペイロードのリレーションシッ .....	13
プ .....	14
2.1.2. 3D Model パー .....	14
ツ .....	14
2.1.3. Thumbnail パー .....	15
ツ .....	15
2.1.4. PrintTicket パーツ .....	15
2.2. 推奨パーツ名 .....	16
2.3. 3MF ドキュメント マークアッ .....	17
プ .....	19
2.3.1. バージョンとエクステンションのサポー .....	20
ト .....	20

2.3.2. XML の使用 .....	22
2.3.3. マークアップ モデル .....	23
2.3.4. ホワイトスペース .....	25
2.3.5. 言語 .....	28
第 3 章. 3D モデル .....	28
3.1. 座標空間 .....	30
3.2. 相対方向と大きさ .....	30
3.3. 3D 変換行列 .....	32
3.4. モデル .....	32
3.4.1. メタデータ .....	33
3.4.2. リソース .....	34
3.4.3. ビルド インストラクション .....	35
第 4 章. オブジェクト リソース .....	36
4.1. メッシュ .....	43
4.1.1. 塗りつぶしルー ル .....	43
4.1.2. 頂点群 .....	44
4.1.3. 三角形群 .....	
4.2. コンポーネント 群 .....	
4.2.1. コンポーネン ト .....	
第 5 章. 素材リソース .....	
5.1. ベース素材 .....	
5.1.1. sRGB Color .....	

## 第 6 章. 3MF パッケージの機能

能	.....
6.1. サムネイ	
ル	.....
6.1.1. JPEG 画像	.....
6.1.2. PNG 画像	.....
6.2. コア・プロパティ	
イ	.....
6.3. 電子署名	.....
6.3.1. 正規化	.....
6.4. 保護されたコンテンツ	
ツ	.....
パートⅡ. 付録	.....
付録 A. 用語	.....
付録 B. 3MF XSD スキーマ	.....
マ	.....
付録 C. 標準名前空間とコンテンツ タイプ	
プ	.....
C.1 コンテンツ タイプ	
プ	.....
C.2 リレーションシップのタイプ	
プ	.....
C.3 名前空間	.....
参考文献	.....

## 序文

---

### この仕様について

[3MF](#) ( 3D Manufacturing Format ) は XML や他の広く利用されている技術を用いて、1 つ以上の 3D モデルのコンテンツおよび外見を表現する一連の規則を説明する。この仕様は 3MF のコンテンツを処理するシステムを構築する開発者に向けて書かれている。

この仕様の主な目的は、3MF コンテンツを生成もしくは利用する独自に作られたソフトウェアやハードウェアシステムの相互運用性を確保することである。この仕様ではプロデューサーとコンシューマーが相互運用性を達成するために満たさなければならない公式の要件を定義する。

この仕様は [3MF ドキュメント](#) と呼ばれる 3D モデルを含むフォーマットについて説明するものである。このフォーマットの要件は Open Packaging Conventions ( OPC ) が定めるそれを拡張したものである。その仕様はあらゆるモデルを構成するコンテンツとリソースを整理するために XML や Unicode、ZIP および他の技術や仕様を使いパッケージングと物理フォーマット規則を説明している。それらは 3MF 仕様のなくてはならない一部である。

この仕様を理解するには、XML および XML 名前空間の仕様の実用的な知識が必要となる。できる限り、依存関係を小さくしているが、それでも本仕様を完全に理解するには 3D 製造部門の一般用語や手順などの知識が必要になる場合がある。

3MF コンソーシアムは 3D コンテンツを扱うアプリケーションが簡単に本フォーマットを導入できるよう、本仕様のオープンソース実装の無料使用を提供している。

パートⅠの「3MF ドキュメント」では主に XML ベースの 3MF ドキュメントフォーマットの詳細を記載している。このセクションでは 3D ドキュメントとドキュメント内のそれぞれのモデルの外見の構成を定義する XML マークアップについて説明している。

パートⅡ「付録」には本文を十分に補足する追加の技術詳細と役に立つ参考情報が含まれている。

この仕様に含まれる情報は変更されることがあるが、発行時点でその正確さを保証するよう最大限の努力が払われている。

## 表記規則

特別な記載がない限り、構文規則は RFC4234 に定義されている ABNF フォーマットで記述される。

用語集に出てくる言葉は[次](#)の様な書式で表示される。構文の説明とコードは等幅フォントで体裁が整えられる。

置き換え可能な項目、つまり、とある値に置き換えられることを意図した項目は筆記体の等幅フォントで体裁が整えられる。メモは以下のように体裁が整えられる。

メモ： これはメモである

---

## 言葉に関する覚書

この仕様では各要件の重要性を定義する語句は大文字で書かれる。これらの語句は RFC2119 の定義にしたがって使用されており、それらの各々の意味を以下に示す。

- 「しなければならない ( MUST )」。この語句、もしくは「要求されている ( REQUIRED )」は、当該項目の絶対的な要求事項であることを意味します。
  - 「する必要がある ( SHOULD )」。この語句、もしくは「推奨される ( RECOMMENDED )」は、特定の条件下では、特定の項目を無視する正当な利用が存在するかもしれないが、異なる選択をする前に、当該項目の示唆するところを十分に理解し、慎重に重要性を判断しなければならない、ということを意味する。
  - 「してもよい ( MAY )」。この語句、もしくは「選択できる ( OPTIONAL )」は、ある要素が、まさに選択的であることを意味する。例えば、特定の市場や状況下で製品機能を拡張する要求があり、とある実装がその要素を選択的に内包するかもしれない。他の実装ではそのオプションが提供する機能を除外するかもしれない。
- 

## ソフトウェア適合性

ほとんどの要件は実装要件というよりも、フォーマットまたはパッケージ要件として記述される。

コンシューマーが適合性を検討するにあたり、次の規則を守らなければならない。

- リソースの枯渇によって強制的に表示される場合を除いて、それらは適合性のあるドキュメント フォーマットを処理するときはエラーを表示してはならない。
- 過剰な処理やパフォーマンスの負担にならない限り、それらは適合性のないドキュメント フォーマットを処理するときはエラーを表示する必要がある。

プロデューサーが適合性を検討するにあたり、次の規則を守らなければならない。

- それらは如何なる適合性のないドキュメント フォーマットも、新しく生成してはならない。
- それらは規定のフォーマットに沿ったドキュメントを修正するとき、いかなる不適合性も持ち込んではならない。

編集アプリケーション ( エディター ) は上記すべてのルールの対象になる。

## パートⅡ. 3MF ドキュメント

## 第 1 章. 3MF ドキュメント フォーマット

この仕様は、3MF ドキュメント フォーマット内部がどのように構成されているか、そしてそれが外部に 3D オブジェクトとしてどのように表現されるのかを説明している。これは単独のファイルフォーマットとして、もしくは、印刷パイプラインのペイロードとして使用することができる。これは、OPC(Open Package Conventions)の規則にしたがって構築されている。

3MF ドキュメントは本仕様に特に示されている場合を除き、その仕様のすべての要件と推奨事項を遵守しなければならない。ここで提示された情報は、[プロデューサー](#) ( 3MF ドキュメント フォーマットのコンテンツを生成する側 ) と [コンシューマー](#) ( 3MF ドキュメントにアクセスもしくは、その内容を 3D オブジェクトに変換する側 ) の両方を対象としている。[エディター](#)は 3MF ドキュメント フォーマットの内容に対し、プロデューサーとコンシューマーの両方の役割を果たす。製造機器は物理的な部品を作り出すコンシューマーとなる。

3MF ドキュメント フォーマットはマークアップ形式で [3D モデル](#)または 1 つ以上の物理的なオブジェクトの詳細を表す。このフォーマットを実装したファイルはコンシューマーが付加製造装置 ( Additive Manufacturing ) や基本的な除去加工装置 ( Subtractive Manufacturing ) の技術を通して物理的なオブジェクトを生成するために必要な基本情報を含んでいる。これは正確に求められる色を表現するためのテクスチャや素材面における内部構造のようなリソースを含んでいる。

このフォーマットは、物理的なオブジェクトを生成するために必要な最小セットのコンポーネントを活用する付属的なコンポーネントも含む。これにより特に印刷ジョブの制御命令の指定や、インターロック構造 ( オブジェクト同士が互いに組み合わせたりバラバラにできない構造 ) であれ分離可能な形であれ、同時に生成されることを想定された複数のオブジェクトの[組立て](#)の記述を含めることができる。

最後に、3MF ドキュメント フォーマットはコア プロパティや電子署名をサポートする OPC の仕様による共通のパッケージ機能を提供する。

---

### 1.1. 本仕様の構成

章	説明
3MF ドキュメント フォーマ	3MF ドキュメント ファイル フォーマットと基本的なパッケ



ット	ージ要件の概要を紹介する。
パーツとリレーションシップ	3MF ドキュメント内の Open Packaging Convention パッケージのパーツと、それらのリレーションシップについて説明する。
3D モデル	3MF ドキュメントの重要な構成要素であるリソースと造形指示を紹介する。
オブジェクト	この章では三角形メッシュでオブジェクトをどのように記述するのかを説明する。
素材	この章では素材リソースがどのように定義されるのかを提示する。これらのリソースは完全な 3D オブジェクトを定義するために使われる。
パッケージの特長	OPC に由来する 3MF ドキュメントの特長を説明する。サムネイルや電子署名、コア プロパティやコンテンツの保護などを含む。
用語	全体のまとめとこの仕様の中で紹介されたすべての用語を定義する。
XSD スキーマ	3D Model パーツのスキーマ。
標準名前空間とコンテンツタイプ	3MF ドキュメントのための標準的な名前空間とコンテンツタイプの参考情報

## 1.2. パッケージ

3MF ドキュメント フォーマットはその物理モデルに対して ZIP アーカイブを利用しなければならない。OPC の仕様には、どのようにしてパッケージが内部のパーツとそれらのリレーションシップ ( 関連性 ) を表すか、すなわちパッケージ モデルについて書かれている。

3MF ドキュメント フォーマットはそれぞれがドキュメント内で特定の目的を果たす明確に定義されたパーツやそれらのリレーションシップを含んでいる。このフォーマットは電子署名やサムネイルを含めるようなパッケージの特長も拡張する。

## 第 2 章. パーツとリレーションシップ

OPC の仕様に書かれたパッケージング規則は任意のペイロードを送信するために利用できる。[ペイロード](#)は、パッケージ内の相互依存するパーツとそのリレーションシップの完全な集合である。この仕様では 3D オブジェクトの定義およびそれをサポートするファイルを含む特定のペイロードである 3D ペイロードを定義する。

3D ペイロードを保持し、この仕様で記述されるルールに沿った OPC ( Open Package Convention ) パッケージが 3MF ドキュメントである。3MF ドキュメントのプロデューサーとコンシューマーはこの仕様に基づいた独自のパーサーや製造機器を実装することができる。

---

### 2.1. 3D ペイロード

3D ペイロードは 3D Model のルート要素を持つペイロードである。1 つ以上の 3D ペイロードが 3MF ドキュメント内に存在しえるが、プライマリ 3D ペイロードは 1 つだけである。

#### 2.1.1. 3D パーツとペイロードのリレーションシップ

3MF ドキュメント内には 3D ペイロードのルート要素を特定するための特別なリレーションシップのタイプ「[3MF ドキュメント StartPart リレーションシップ](#)」が定義されている。[プライマリ 3D ペイロードルート](#)は 3D Model パーツで、パッケージの中からプライマリ 3D ペイロードを探すため、3MF ドキュメント StartPart リレーションシップによって参照される。3MF ドキュメント StartPart リレーションシップは 3D ペイロードのルートを特定する 3D モデル要素を参照しなければならない。

ペイロードは 3D Model パーツを処理するために必要なすべての複数パーツセットを含んでいる。3D ペイロードに記述された 1 つのオブジェクトを製造するために使用するすべてのコンテンツは 3MF ドキュメントの中に含まれていなければならない。

3MF ドキュメントの中に含まれるパーツを表 2-1 に示す。これらのパーツのリレーションシップとコンテンツタイプは付録 C 「標準名前空間とコンテンツ タイプ」に定義されている。各パーツは付録 C に指定された適切なコンテンツ タイプのみを使用しなければならない。

3D ペイロードに含まれるパーツは 3D ペイロードのルート要素と明示的に参照関係を持っている。エクステンションの中で指定されない限り、3MF ドキュメントは 3MF ドキュメントパッケージ外のリソースを参照してはならない。参照関係に関するさらなる情報は OPC の仕様が参考になる。

3D ペイロード内の複数パーツは、次に記載されるペイロードの中の 2 つのパーツ間に構築される適切な参照関係の内の 1 つを使わなければならない。あるパーツから別のパーツへの同じタイプのリレーションシップが複数あってはならない。リレーションシップのタイプは付録 C の「標準名前空間とコンテンツ タイプ」に定義されている。

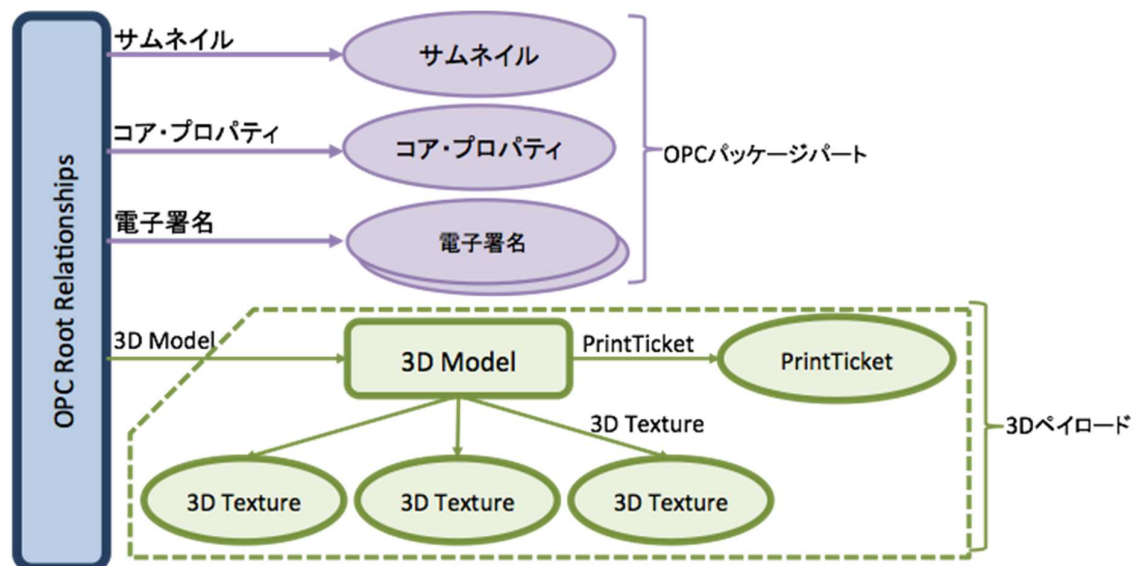
リレーションシップを作るプロデューサーは次のリレーションシップのタイプ、PrintTicket、StartPart および Thumbnail、の対象となるすべてのパーツを 3MF ドキュメント内に含まなければならない。コンシューマーはそれらのタイプを持つリレーションシップの対象となるパーツが 3MF ドキュメントに含まれていなければエラーを出さなければならない。

表 2-1 3MF ドキュメント パーツ

パーツ名	説明	関連ソース	必須/オプション
3D Model	製造のための 1 つ以上の 3D オブジェクトの記述。	Package	必須
コア・プロパティ	様々なドキュメント プロパティを含む OPC のパーツ。	Package	オプション
電子署名基点	パッケージ内の電子署名のルート要素となる OPC のパーツ。	Package	オプション
電子署名	それぞれの電子署名を含む OPC のパーツ。	Digital Signature Origin	オプション
電子署名証明書	電子署名の証明書を含む OPC のパーツ。	Digital Signature	オプション
PrintTicket	3D Model パーツ内の 3D オブジェクトを出力する際に使われる設定。	Package	オプション
Thumbnail	パッケージ内の 3D オブジェクトを表す小さな JPEG か PNG 画像。	Package	オプション
3D Texture	3D Model パーツ内の 3D オブジェクトに複雑な情報を適用するために利用されるファイル。(サムネイルやエクステ	3D Model	オプション

ンションに使用されうる )

図 2-1 典型的な 3MF ドキュメント



### 2.1.2. 3D Model パーツ

[3D Model パーツ](#)は 3D 製造工程によって製造される 1 つ以上のオブジェクトの定義を含んでいる。3D Model パーツは 3D ペイロードの唯一の妥当なルート要素となる。

3D Model パーツは 2 つのセクションを持っている。1 つはオブジェクトや素材を含むリソースの定義、もう 1 つは実際に構築するための特定の項目である。3D Model パーツのコンテンツ タイプは付録 C「標準名前空間とコンテンツ タイプ」に定義されている。

### 2.1.3. Thumbnail パーツ

[サムネイル](#)は 3MF ドキュメントの全体を表現する小さな画像である。サムネイルは外部に対して 3MF ドキュメントの内容を簡単に表示する事ができる。

OPC の仕様ではどのパーツにもサムネイルを付け加える事ができるが、3MF ドキュメントのコンシューマーはパッケージ全体からのリレーションシップを持つサムネイルだけを処理する必要がある。これらの Thumbnail パーツは JPEG もしくは PNG 形式でなければならない。3MF ドキュメントのコンシューマーは他の各パーツに付け加えられたサムネイルを無視する必要がある。Thumbnail パーツのタイプは C.2 章「コンテンツ タイプ」で詳細が指定されている。

C.3 章「リレーションシップのタイプ」にサムネイルのリレーションシップのタイプに関するさらなる情報が記載されている。

### 2.1.4. PrintTicket パーツ

[PrintTicket パーツ](#)は 3D プリントするコンシューマーにユーザの意図や機器の構成情報を提供する。PrintTicket パーツは 3D Model パーツだけに、そして各 3D Model パーツに [PrintTicket](#) を最大 1 つだけ付け加えなければならない。PrintTicket のフォーマットはコンシューマーの特定の環境によって規定される。例えば、Microsoft Windows 上でプリントするには、正しい PrintTicket の設定が 3D Manufacturing の仕様に沿った Print Schema Keywords に記述されている。

もし PrintTicket が存在しない場合やコンシューマーが PrintTicket をサポートしていない場合、コンシューマーが自身のデフォルト値を適用する。

---

## 2.2. 推奨パーツ名

3MF ドキュメントのプロデューサーとコンシューマーは名前でパーツを参照し、リレーションシップの名前で関連するパーツの目的を特定する。OPC の仕様はパーツ名の文法を記述

している。しかし、これらのルールだけにしがっていると、ユーザーがパッケージの中身を理解することが難しくなる。例えば、ユーザーはどのパーツが正確に製造するために必要かを知るために 3MF ドキュメントのすべてのリレーションシップのパーツを開く必要があるかもしれない。

明確に定義された、人に読みやすい規則にしたがったパーツ名にすることで、結果として閲覧しやすく、特定のパーツの配置が容易になる。パーツ名は OPC 仕様に記された文法に沿っていることを確認しなければならない。

3MF ドキュメントのプロデューサーは次のパーツの命名規則を使うことが推奨される。

- 3D Model パーツ名は"/3D/"で始まり、".model"という拡張子で終わる 2 つのセグメントを含んでいる必要がある。(例) /3D/3dModel.model
- PrintTicket のパーツ名は 3D Model パーツとのリレーションシップを通して結びついていて、"/3D/Metadata/"という始めの 2 つのセグメントに".xml"という拡張子を付けた 3 つのセグメントを含んでいる必要がある。(例) /3D/Metadata/Model\_PT.xml
- 3D Texture パーツ名は"/3D/Textures/"という始めの 2 つのセグメントを付与した 3 つのセグメントを含んでいる必要がある。3D Texture パーツはそのリレーションシップを通して 3D Model パーツと結び付けられていなければならない。(例) /3D/Textures/coloring.png
- 3D ペイロードに関連する標準ではないパーツ名は"/3D/Other/"という 2 つのセグメントをはじめに付与した 3 つのセグメントを含む必要がある。

パーツ名は"/"で始まる絶対パスを使わなければならない。パーツ名は空文字もしくはピリオドで始まるはならない (例: "/3D/.png"もしくは"/3D/").

---

## 2.3. 3MF ドキュメント マークアップ

3MF ドキュメント マークアップは 3MF ドキュメントを製作・利用する互換システムの独立した開発を円滑にするために設計されている。

3MF ドキュメントのマークアップは (競合することもある) 2 つのゴールのトレードオフを反映したものである。

1. 3MF ドキュメント マークアップはできる限り無駄を省く必要がある。すなわち、最小限の基本的な操作群を含み、一般的な 3D オブジェクトを忠実に製造するために必要なマークアップを構築する。仕様に冗長性を含むと、その実装が意図しない非互

換性問題を増加させる危険性がある。冗長性は実装とテストのコストをも増加する要因になり、そして、概してメモリを必要とする。

2. 3MF ドキュメント マークアップはコンパクトにする必要がある。すなわち、最も一般的で基本的な要素はコンパクトな表現にする必要がある。肥大化した表現は 3MF ドキュメントを扱うシステムの性能を損なう。バイト数が増加すると、通信時間をさらに要する。圧縮は通信時間を改善するが、冗長表現による性能劣化を回避することはできない。

### 2.3.1. バージョンとエクステンションのサポート

3MF ドキュメント マークアップはそれ自体の仕様の進化を見越した設計になっており、サードパーティによる拡張マークアップを許可している。

エクステンションは 3MF の重要な部分であり、その前提で、この仕様のコアは可能な限り狭められている。高度な機能にはア・ラ・カルトモデルが採用され、プロデューサーは ( <model>要素内で所定の XML 名前空間を宣言することによって ) 明確にどのエクステンションが使われているかを記載し、コンシューマーはどのエクステンションをサポートするかを明確に記載することで、他のツールがその一連のどのパーツが無視されうのかを知る。バージョンは変更があるたびに名前空間を更新することで逐次管理される。したがって、バージョン管理はコア仕様とエクステンションそれぞれ独立して行い、それぞれのバージョンはその名前空間を確認することで解決されうる。

エクステンション仕様は構成可能な数を制限するため、1 つ以上のコア仕様の対象バージョンを 含まなければならない。プロデューサーは特定のエクステンションが必要であることを 3MF ドキュメント内に指定することができる。コンシューマーがそのエクステンションをサポートしていない場合に、その 3MF ドキュメントの編集または製造を試みると、そのエクステンションの名前空間を無視するのではなく、編集や製造に失敗しなければならない。

エクステンション ポイントは XSD スキーマ ( 付録 B: 3MF XSD スキーマを参照 ) 内において明示的に <any> もしくは <anyAttribute> 要素で囲む ( さらに、本仕様に沿った要素図で可視化される )。これらは他の名前空間でも求められる。そしてそれは適切な仕様と付随する XSD スキーマを探す方法を示している 必要がある。

### 2.3.2. XML の使用

この仕様内で定義されるパーツのすべての XML コンテンツは以下のルールにしたがってなければならない。

1. XML コンテンツは UTF-8 でエンコードされなければならない。4.3.3 章の XML 仕様で定義されているような、XML のエンコード宣言が含まれている如何なる要素があったとしても、UTF-8 か UTF-16 以外を指定してはならない。
2. XML1.0 の仕様では、典型的には、内部実体の拡張技術を利用したサービス拒否 (DoS) 攻撃を可能とする、DTD(Data Type Definitions)の仕様を許可している。この潜在的な脅威を軽減する策として、この仕様内で定義された XML マークアップ内で DTD コンテンツを使用してはならない。そして DTD が存在した場合、コンシューマーはそれをエラーにする必要がある。
3. XML コンテンツはこの仕様で定義されている付属 XSD スキーマに対して有効でなければならない。特に、XSD が XML マークアップ内の特定の位置に存在する任意の名前空間から要素や属性を引き出すことを許可していない限り、XML コンテンツは付属 XSD で明示的に定義されてない名前空間に由来する要素や属性を含んではならない。
4. XSD スキーマ内もしくは仕様内の他の方法で明確に定義されていないなら、XML コンテンツは xml や xsi 名前空間に由来する要素や属性を含んではならない。
5. XML コンテンツは、特に 10 進数データを含む値に関して、en-us ロケールで生成され、パースされなければならない。

### 2.3.3. マークアップ モデル

3MF ドキュメント マークアップは要素や属性、名前空間を使用する XML ベースのマークアップ言語である。3MF ドキュメント マークアップのためのスキーマは要素とその属性、コメント、そしてホワイトスペースのみを含んでいる。

#### 2.3.3.1. XML 名前空間

3MF ドキュメント コアの [XML 名前空間](#)、3D Model パーツのマークアップの要素や属性のために使われる主要な名前空間は付録 C「標準名前空間とコンテンツ タイプ」に記載されている。この仕様に定義されていないすべての要素や属性は、付属する 3MF エクステンションの名前空間の序文で書かれなければならない。

注意として、ある要素上の一般的でない XML 名前空間は自動的に他の名前空間が付与されていない限りその要素の属性値に適用される。しかし、子要素にはその XML 名前空間が適用されないため、子要素全ての属性値には個別に XML 名前空間を付与しなければならない。



エクステンション点で定義されている如何なる属性値も対応する名前空間を付与しなければならない ( すなわち、全てのエクステンション点は XSD スキーマの名前空間では "other" と定義されている )。

#### 2.3.4. ホワイトスペース

3MF ドキュメントはマークアップ内で柔軟なホワイトスペースの使用を許可している。ホワイトスペースを 1 つ入れることが許されている場所ではどこでも複数のホワイトスペースを使用してもよい。3MF ドキュメントは `xml:space` 属性を使用してはならない。さらに、3MF ドキュメント スキーマがホワイトスペースの折りたたみを許可する属性を指定している場所では、属性値の先頭や末尾のホワイトスペースは XML スキーマ仕様で指定されているホワイトスペースの折りたたみの挙動に依存する他のホワイトスペースと共に使用されてもよい。

**メモ：** 厳密に許可されるホワイトスペースについては 3MF スキーマを参照

#### 2.3.5. 言語

3MF ドキュメント コンテンツの言語 ( 典型的にはメタデータ内で提供されるコンテンツに役に立つ ) は `xml:lang` 属性で指定してもよい。その値は子や孫要素に引き継がれる。この属性は W3C XML 仕様に定義されている。コンテンツの言語が不明な場合、"und" ( undetermined ) が使われなければならない。

## 第 3 章. 3D モデル

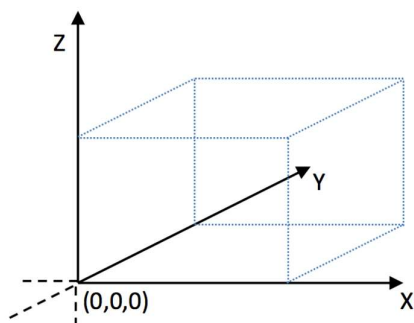
この仕様では、[モデル](#)とは 1 つのオブジェクトまたは 3D 製造行程により一連の操作として生成された複数のオブジェクトのことを指す。つまり、単一のオブジェクト、または、同一形状の複数オブジェクト、形状の異なる複数オブジェクト、他のオブジェクトに完全に囲まれたオブジェクト、そして組み合わされて切り離すことはできない[組立て](#)られた複数オブジェクト、が含まれる。

---

### 3.1. 座標空間

この仕様では右手座標系を基準にする。プロデューサーとコンシューマーは、座標空間の原点を機器の出力領域（トレイ、プラットフォーム、ベッドなど）の手前の左下に置き、x 軸は右方向が正、y 軸は手前から奥に向かう方向が正、z 軸は上方向を正に定義しなければならない。そして<model>要素で示された座標系の単位解像度を使用しなければならない。

図 3-1 座標空間



### 3.2. 相対方向と大きさ

この仕様での相対方向を次のように定義する。[上面](#)が示すのはプリント可能な XY 平面の z 方向の最大値で、[底面](#)はその最小値で z の値が 0 の XY 平面である。これは通常はプリントベッド面となる。[左側面](#)は x の値が 0 で示されるプリント可能な座標空間の YZ 平面の最小とする。[右側面](#)はプリント可能な座標空間の x 座標の値の最大値である YZ 平面を示す。[前面](#)は y 座標が 0 で、プリント可能な座標空間の xz 平面の最小値を示す。[後面](#)はプリント可能な座標空間の y 座標の値が最大値の xz 平面を示す。

これらの用語はモデルのバウンディング ボックスにも適用されうる。その場合、この使用で定義される座標空間に変換されたあとのモデルのバウンディング ボックスに対して相対的なものとして定義される。

プロデューサーとコンシューマーは座標をこの仕様で定義された座標空間に相対的なものとして解釈しなければならない。

---

### 3.3. 3D 変換行列

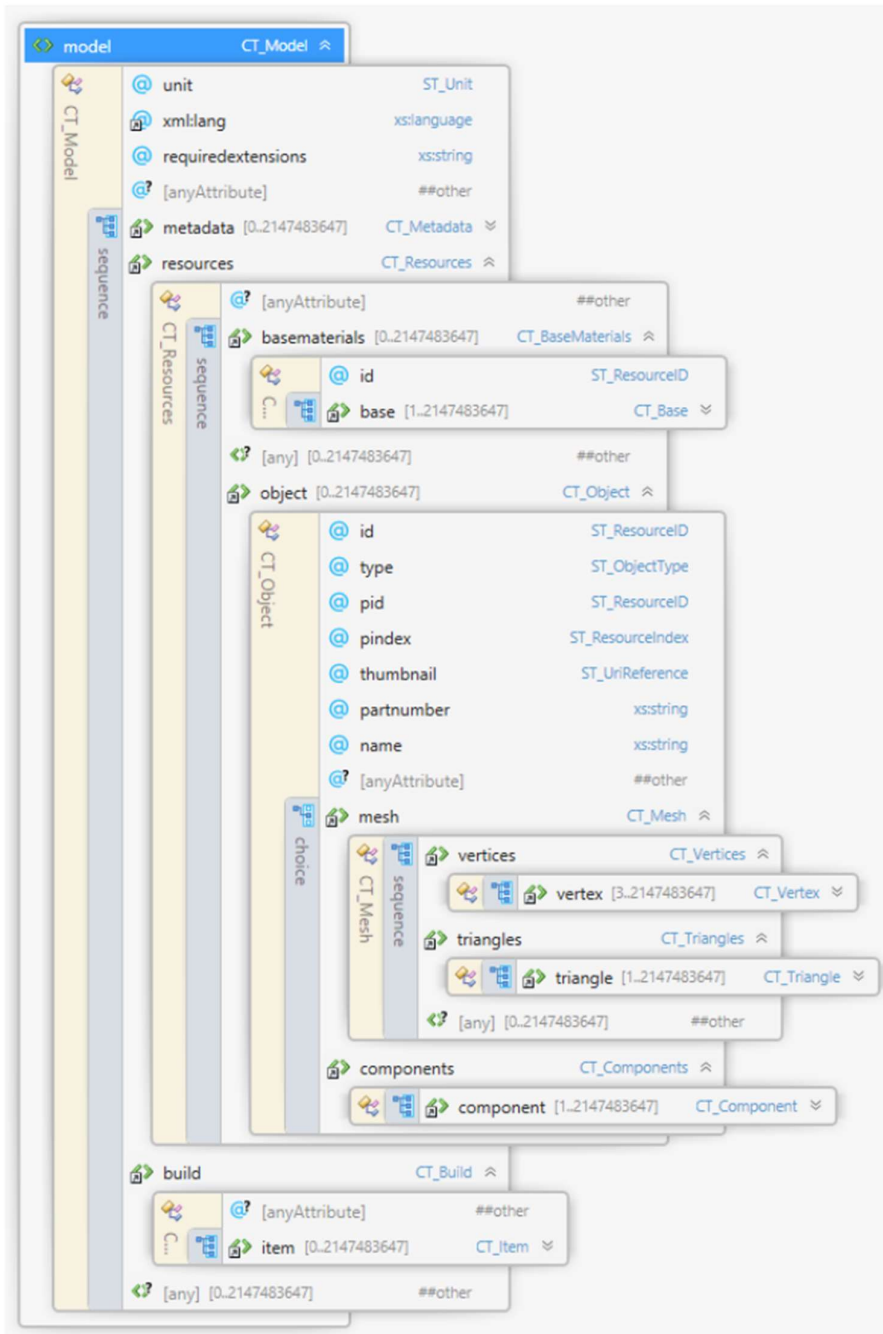
オブジェクトが回転、拡大縮小、移動の変換を必要とする際に、行優先のアフィン [3D 変換行列](#) (4x4) が使用される。行列は特異行列、および特異行列に近いものでないほうがよい。変換行列は最初の 3 列の値だけが指定される形式とする。最後の列は提供されることはなく、0.0, 0.0, 0.0, 1.0 で固定とする。属性として指定された際、変換行列は “m00 m01 m02 m10 m11 m12 m20 m21 m22 m30 m31 m32” の形式で、各値は任意の精度の 10 進数とする。

$$\begin{bmatrix} m00 & m01 & m02 & 0.0 \\ m10 & m11 & m12 & 0.0 \\ m20 & m21 & m22 & 0.0 \\ m30 & m31 & m32 & 1.0 \end{bmatrix}$$

すべての変換がオブジェクトに適用された後、モデルは正の体積である 必要があり、座標空間の正の象限に配置される 必要がある。

### 3.4. モデル

図 3-1 3MF におけるモデル XML 構造の概要



この XML の仕様はシンプルな前方参照専用パーサーを使用する前提で設計されている。そして要素が定義される順番はこれをサポートする。つまりエクステンションで特別に認めら

れた場合を除いて、プロデューサーは各要素をドキュメントのどこかで参照されるより前に定義しなければならない。

<model>要素

属性

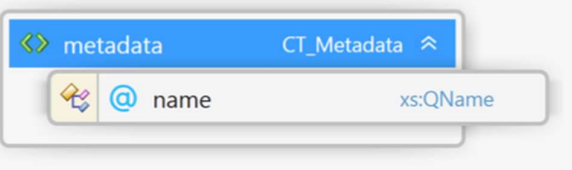
名前	型	使用	デフォルト 値	固定 値	注記
unit	ST_Unit		millimeter		モデル中のすべての頂点、位置、寸法を解釈するために使用される単位を指定する。 有効な値は micron, millimeter, centimeter, inch, foot, meter となる。
requiredextensions	xs:string				空白で区切られた名前空間の接頭辞のリストで、ドキュメントを処理するために必要なエクステンションのセットを表す。必要とされたエクステンションをサポートしていない場合、エディターと製造機器はそのドキュメントを処理してはならない。
xml:lang	xs:language				現在の要素とその派生要素で使用されるデフォルトの言語を指定する。言語は RFC 3066 に準じて指定される。

<model>要素は 3D Model パーツのルート要素である。3D Model パーツには<model>要素が正確に 1 つ存在しなければならない。モデルはゼロ以上の子要素となる<metadata>要素（“メタデータ”の詳細は 6.1 を参照）を持つことができる。モデルは<resources>と<build>の 2 つの追加の子要素を持たなければならない。<resources>要素は 3D オブジェクトを定義するために読み込まれる定義のセットを提供する。<build>要素はジョブのパーツとして実際に製造されるアイテムのセットを提供する。

プロデューサーは、そのエクステンション データなしではドキュメントが意味を失うので  
ない限り、エクステンション データを要求しないほうがよい。コンシューマーがサポート  
されていないエクステンションを無視することを認めると、より適切な代替を提供できる。

3.4.1. メタデータ

<metadata>要素

図						
属性	名前	型	使用	デフォルト	固定	注記
				値	値	
	name	xs:QName	必須			この仕様で定義されたメタデータの周知の名前（表 8-1 を参照）、もしくは<model>要素で XML 名前空間宣言の接頭辞としてつけられなければならないベンダー定義のメタデータを含む。

3MF ドキュメントのプロデューサーは、ドキュメントについての追加情報を<model>要素配下のメタデータのフォームに提供する必要がある。

名前空間名の無い 3MF ドキュメントのメタデータはこの仕様で定義されている名前と値に制限されなければならない。名前がこの仕様で定義されていない場合、デフォルトの名前空間に属していない<model>要素の XML 名前空間宣言の名前空間名の接頭辞として使用されないなければならない。この仕様に含まれる有効なメタデータ名とコンテンツは以下の通りである。

表 8-1 メタデータの値

コンテキスト	名前	コメント
Model	Title	

Designer
Description
Copyright
LicenseTerms
Rating
CreationDate
ModificationDate

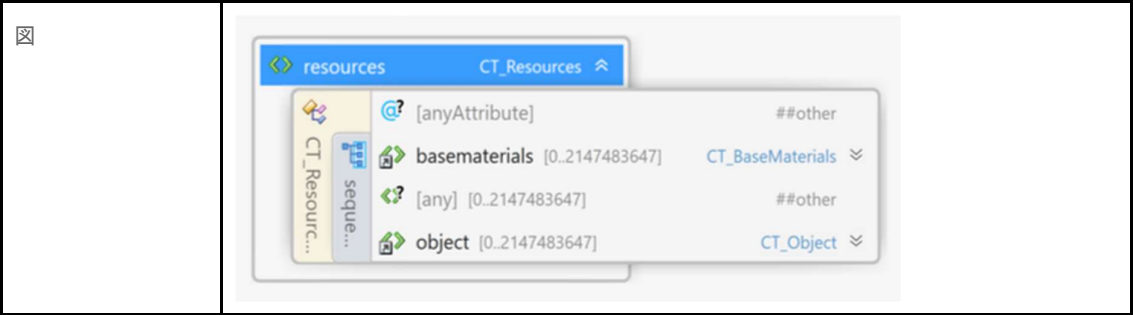
コンシューマーは、この仕様の将来のバージョン、もしくは認識されないプロデューサーやターゲットのコンシューマーに由来する認識できない名前のメタデータをすべて無視しなければならない。<metadata>要素のコンテンツは任意の文字列となる。

この仕組みを使用して追加の情報を受け取ることが希望するコンシューマーは、プロデューサーがコンテンツを期待する形で生成できるように、名前空間の URI と明確に定義されたメタデータ名と期待されるコンテンツを発行する必要がある。

プロデューサーは同じ名前で複数のメタデータ要素を生成してはならない。

3.4.2. リソース

<resources>要素



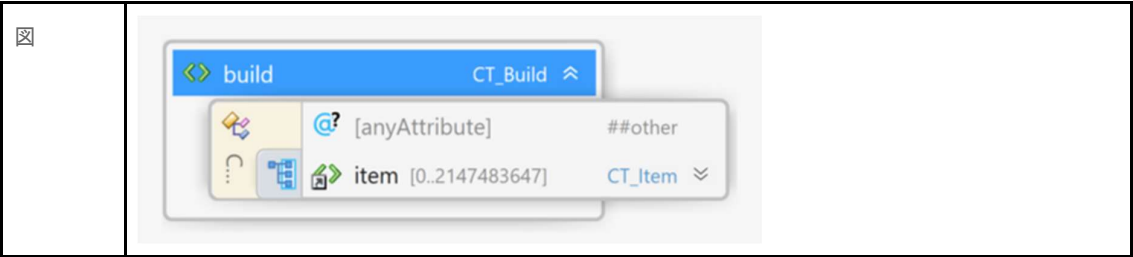
<resources>要素は 3D オブジェクト定義全体の構成要素のライブラリのルート要素の役割を果たす。この仕様では、オブジェクト、プロパティ、および材料はまとめて[リソース](#)として参照される。

各リソースはその完全な定義のために他のリソースに依存する場合がある。例として、完全に 3D オブジェクトを記述するために、あるオブジェクト リソースは材料 リソース、あるいは他のオブジェクト リソースを参照する。

オブジェクト リソースは製造可能な単体の 3D オブジェクトを表すが、必ずしも製造される必要は無い。実際に製造されるオブジェクトは<build>要素の子要素である<item>要素から参照される。モジュール化とコンポーネントの再利用のためにオブジェクトは主にリソースとして定義され、その結果としてマークアップ全体のサイズは小さくなる。

リソースの ID はモデル内で一意でなければならない。

3.4.3. ビルド インストラクション



<build> 要素は処理するジョブのパーツとして 1 つ以上の製造するアイテムを含む。コンシューマーは <item> 要素により参照されていないどの 3D オブジェクトも出力してはならない。

3.4.3.1. アイテム要素

<item>要素

<> item CT\_Item ^

CT\_Item

@ objectid ST\_ResourceID

@ transform ST\_Matrix3D

@ partnumber xs:string

@? [anyAttribute] ##other

属性

名前	型	使用	デフォルト	固定	注記
		値	値	値	
objectid	ST_ResourceID	必須			id に合致する<object>要素への参照



	transform	ST_Matrix	出力される item に適用される変換行列 ( §3.3 の 3D 変換行列を参照 )
	partnumber	xs:string	item への一意の識別子。変換が変更される場合のみ、エディターにより保持される <u>必要がある</u> 。

<item>要素は 3D 製造機器により出力されるべき一つのオブジェクト リソースを示す。コンシューマーはオブジェクトを出力する前に変換を反映しなければならない。

1 つの 3MF ドキュメントは同時に製造される複数のオブジェクトを持つことがある。製造時のこれらのアイテムの配置は初期値とみなされ、コンシューマーはアイテムを製造する際に造形体積を最小化するために再配置してもよい。しばしばオブジェクトは座標空間の中でインターロック構造で製造されるよう配置される場合があるが、これらのオブジェクトのプロデューサーはコンポーネントとしてそれらをまとめる必要があり ( §4.2 “コンポーネント群” を参照 )、3D 製造機器は互いが相対的に配置されたオブジェクトのコンポーネントを変換してはならない。

アイテムが重なる場合は、3D 製造機器は曖昧さを解決するために正の塗りつぶしルール ( 4.1.1 節で説明される ) を使用しなければならない。これを防ぐためには、3D 製造機器は造形体積を最小化する際にアイテムが交差するのを許さないほうがよい。

メモ：アイテムは “other” タイプのオブジェクトを参照してはならない。

第 4 章. オブジェクトリソース

[オブジェクトリソース](#)は再利用可能なオブジェクトを表し、それらのオブジェクトは ( 参照により ) 出力、もしくはより複雑なオブジェクトや組立てを構成する。

<object>要素

属性	<div><div><div>objectCT_Object</div><div><div>CT_Object</div><div><div>@ idST_ResourceID</div><div>@ typeST_ObjectType</div><div>@ pidST_ResourceID</div><div>@ pindexST_ResourceIndex</div><div>@ thumbnailST_UriReference</div><div>@ partnumberxs:string</div><div>@ namexs:string</div><div>@ [anyAttribute]##other</div><div>meshCT_Mesh</div><div>componentsCT_Components</div></div></div></div></div>					
	名前	型	使用	デフォルト	固定	注記
			値	値		
	id	ST_ResourceID	必須			このオブジェクトの一意的 ID を定義する。
	type	ST_ObjectType		model		モデルの中でのオブジェクトの機能を指定する。有効な値は “model”、 “solidsupport”、 “support” もしくは “other”。サポート オブジェクトが不要な場合、コンシューマーは無視してもよい。コンシューマーは独自の support オブジェクトをサポート オブジェクトが提供されていた場所に生成してもよい。ただし推奨されない。
	pid	ST_ResourceID				id 属性値に合致するプロパティ グループ 要素への参照 ( 例 <basematerials> ) 。 pindex が指定された場合は要求されている。
	pindex	ST_ResourceIndex				pid で示されるプロパティ グループ内を 0 から始まるインデックスで参照する。このプロパティはオブジェクトを構築する

	のに使用される。		
thumbnail	ST_UriReference		オブジェクトをレンダリングしたイメージの JPEG もしくは PNG 形式の 3D テクスチャへのパス。
partnumber	xs:string		エディターがオブジェクトを編集または抽出している間保持する <u>必要がある</u> パーツ番号。
name	xs:string		可読性を向上するためのオブジェクト名。

オブジェクト リソースは<object>要素で定義される。<object>要素は プロパティ グループのための属性とオブジェクト全体 ( ただし子要素の<triangle>要素やコンポーネント参照された<object>要素のように継承した要素で上書きされる箇所を除く ) に適用される特定のプロパティ メンバーを持つ。このオブジェクトにマテリアルが割り当てられた三角形が含まれる場合、そのオブジェクトは明示されていないプロパティを持つ三角形のためのデフォルト値としての役割を持つため pid と pindex を明示しなければならない。プロパティが何も割り当てられていない場合は、コンシューマーにオブジェクトのプロパティの選択が委ねられる。

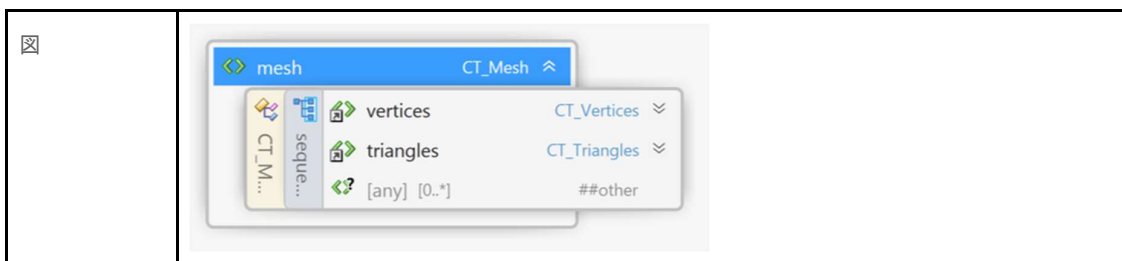
コンポーネントを含むオブジェクトでは、オブジェクト タイプは継承されたオブジェクトで上書きされるため無視される。

オブジェクトのサムネイルは適切な 3D テクスチャのリレーションシップを 2.1.1 節に記載されているようにモデルパーツに持たなければならない。

パーツ番号は ツール チェインの間に変更されるかもしれないオブジェクトを追跡する手段として作られている。3MF ドキュメントを編集、処理する際、これらのパーツ番号は可能な最大の段階まで保存される必要がある。つまり、複数の部分に分割されたオブジェクトに複製し、結合されたオブジェクトからは削除し、変更されたオブジェクトのものは維持する。

4.1. メッシュ

<mesh>要素

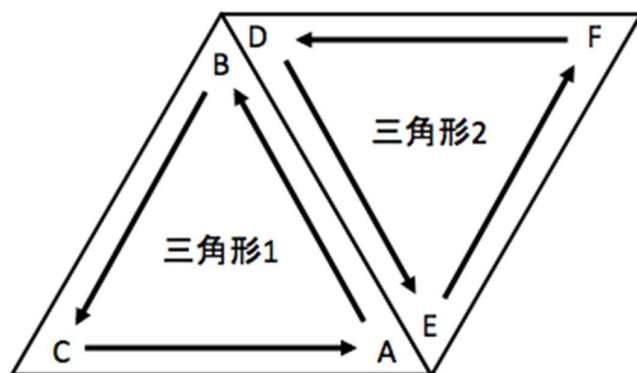


<mesh>要素はオブジェクト ボリユームの三角形 [メッシュ](#) 表現のルートで、頂点群と三角形群を含む。

mesh が “model” タイプまたは “solidsupport” タイプのオブジェクトの配下にある場合、以下のものを持たなければならない。

- マニフォールドな稜線: メッシュの中のすべての三角形の稜線が共通の頂点の終点を確実に別の 1 つの三角形の稜線と共有する。
- 一貫した三角形の向き: メッシュの中のすべての隣接した三角形のペアはメッシュの外向きの面法線が同じ向きでなければならない。つまり、共有する稜線上の頂点の宣言順序は逆順でなければならない。例を挙げると、三角形 1 が ABC という頂点、三角形 2 が DEF という頂点を持ち、三角形 1 と三角形 2 は AB/DE という稜線を共有している場合、A と E は同じ頂点、B と D は同じ頂点 ( 図 4-1 を参照 ) となければならない。この仕様では、三角形の面法線は ( 三角形の頂点の順序は ABC ) はベクトルの外積  $(B - A) \times (C - A)$  で求められる方向を持つ単位ベクトルとして定義される。例として、図 4-1 で示される三角形の法線はページの外へ抜け出す方向となる。負の行列式を含む変換が反映された場合、体積の符号を保つためにこれらの頂点の順序は反転されなければならない。
- 外向きの法線: すべての三角形はオブジェクトの内側から外向きの法線を持たなければならない。次のセクションで定義される正の塗りつぶしルールにしたがって、負の値の体積を持つメッシュはプリントされない ( もしくは空となる ) 。規格にしたがって正の値で埋めるルールは先行する二つのルールの組み合わせで、メッシュとはすなわち、穴やギャップ、開いている稜線、方向付け不可能な面 ( 例: クラインの壺 ) の無い連続した面である。

図 4-1 一貫した三角形の向き



“support” および “other” のタイプのオブジェクトはこれらの条件を外れることが許される。“other” タイプのオブジェクトは Build セクションから参照されてはならない。すべてのメッシュは以下を持つ必要がある。

- 最小限の自己交差: プロデューサーはそれ自身と交差、もしくは重なり合うオブジェクトを作成しないほうがよい。コンシューマーは自分自身と交差するメッシュや重なりあうオブジェクトを受け入れるようにしなければならない。ただし“model”タイプと“solidsupport”タイプは正の塗りつぶしルール ( 次のセクションを参照 ) で埋められているものとして扱われる。
- 非縮退: すべての三角形は法線を求める時に数学的不安定を起こさないように、ゼロでない面積を持つ必要がある。

製造機器は“model”タイプまたは“solidsupport”タイプを持つオブジェクトをそれぞれ固体の素材または内部構造で充填してもよい。充填についてはプリントチケットで指定されていることもある。オブジェクトが“support”タイプであれば、サポートの実際の表現 ( 厚さなど ) についてプリントチケットのパラメータを考慮するかどうかは製造機器に選択が任されているが、いずれにしても製造機器はそのメッシュを充填してはならない。

サポート構造物 ( “solidsupport”タイプ、“support”タイプとも ) は無視、もしくは自動生成されたサポートで置き換えてもよいが、その挙動は推奨されない。

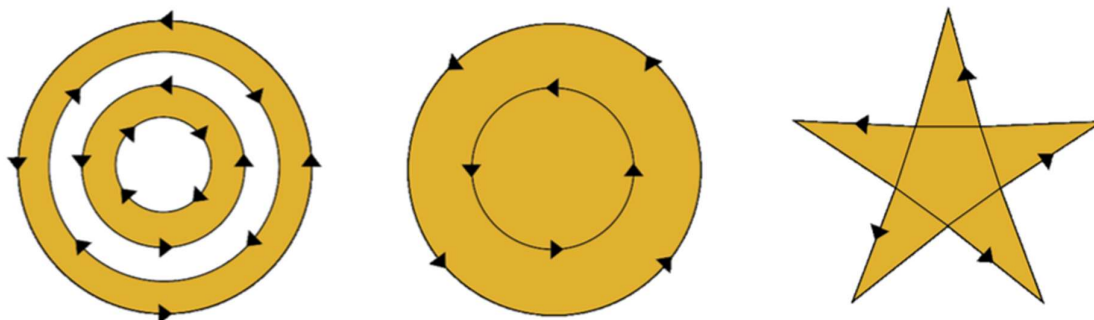
#### 4.1.1. 塗りつぶしルール

メッシュは重なったりそれ自身が交差したりすることがあるため、囲まれた領域の何がポリユームの「内部」に入っているのか、そしてどちらがポリユームの「外部」なのかを示すために塗りつぶしルールが明記されなければならない。メッシュのグループの囲まれたポリユームは塗りつぶしアルゴリズムを反映することにより定義される。塗りつぶしアルゴリズムはどのようにして幾何学的図形の交差している領域(area)を結合して領域(region)が作られるかを決定する。塗りつぶしのルールは 2D においても 3D と完全に同じ形で定義される。そ

ここでこの例は図示を簡単にするために同等の 2D で示す。コンシューマーはこの事実を活用してスライシングの後に塗りつぶしルールを適用してもよい。それにより、2D 上の操作のみでアルゴリズムを簡略化できる。慣例的に 2D の図は反時計回りの時に正の面積を持ち、稜線に沿って見た際にローカルでの「外部」は右側となり、三角形のローカルでの「外部」は法線方向と等価である。

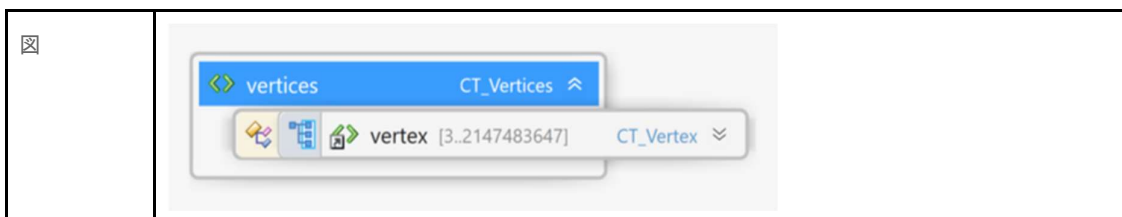
3MF では、塗りつぶしルールは常に正の塗りつぶしルールが適用される。それはその点から任意の方向へ無限遠に線を引き、形状のどこと交わるかによって点の「内部性」を決定する。ゼロから数え始めて、線がローカルの「内部」からローカルの「外部」へ通過する度 1 を加算し、その反対の際は 1 を減算する。通過をカウントした後、結果が 1 以上ならば、そのポイントはグローバルでボリュームの内部となり、そうでない場合は外部ということになる。

図 4-3 ポジティブアルゴリズムを用いた塗りつぶし



#### 4.1.2. 頂点群

##### <vertices>要素



<vertices>要素はオブジェクトのすべての<vertex>要素を含む。頂点はメッシュの各三角形の角を意味する。これらの要素の順番は<triangle>のような他の要素から参照される 0 から始まるインデックスを暗黙的に定義する。プロデューサーは重複を融合することによって非多様体となる稜線が生成される場合を除き、重複する vertices を含めないほうがよい。さらに、プロデューサーは適切な時は極めて近接する頂点は 1 つの頂点に畳み込む必要がある。整数の桁あふれを防ぐために、vertex 配列が含む頂点の数は  $2^{31}$  未満にしなければならない。

4.1.2.1. 頂点

<vertex>要素

図

<> vertex CT\_Vertex ^

CT\_Vertex

@ x ST\_Double

@ y ST\_Double

@ z ST\_Double

@? [anyAttribute] ##other

属性	名前	型	使用	デフォルト値	固定値	注記
	x	ST_Number	必須			x 軸の頂点座標
	y	ST_Number	必須			y 軸の頂点座標
	z	ST_Number	必須			z 軸の頂点座標

<vertex>要素はメッシュの三角形から参照される3次元空間の点を表す。座標を表す小数の値は任意の有効桁数で記録することが可能である。プロデューサーはプロデューサー自身の計算精度やコンシューマー側で想定される計算精度を上回る有効桁数を使わないほうが良い。可変の有効桁数を持つことができるというASCIIエンコーディングの性質は固定幅のバイナリフォーマットに対して、記憶領域の効率の差を補って余りある大きな利点がある。

4.1.3. 三角形群

<triangles>要素



<triangles>要素は 3D オブジェクトのメッシュ全体を表すための 1 つ以上の<triangle>要素を含む。オブジェクトタイプが “model” の場合、ソリッドなボディを形成するためには mesh は最低 4 つの三角形を含む必要がある。整数値の桁あふれを防ぐために triangle 配列が含む三角形の数は  $2^{31}$  未満にしなければならない。

4.1.3.1. 三角形

<triangle>要素

属性

名前 型 使用 デフォルト値 固定値 注記

v1	ST_ResourceIndex	必須			このメッシュのための頂点群への 0 から始まるインデックス。三角形の最初の頂点を定義する。
v2	ST_ResourceIndex	必須			このメッシュのための頂点群への 0 から始まるインデックス。三角形の 2 番目の頂点



				を定義する。
v3	ST_ResourceIndex	必須		このメッシュのための頂点群への 0 から始まるインデックス。三角形の 3 番目の頂点を定義する。
p1	ST_ResourceIndex			三角形の最初の頂点の object レベルの pindex を上書きする

**メモ：** 三角形の向きは 4.1 節のとおり、変換の行列式の符号に影響される。

p2	ST_ResourceIndex			三角形の 2 番目の頂点の object レベルの pindex を上書きする。
p3	ST_ResourceIndex			三角形の 3 番目の頂点の object レベルの pindex を上書きする。
pid	ST_ResourceID			三角形の object レベルの pid を上書きする。

<triangle>要素はメッシュの 1 つの面を表す。頂点 (v1, v2, v3) の順番は、三角形の面の法線をオブジェクトの外側に向けるために反時計回りで示されなければならない。v1, v2, および v3 のインデックスはそれぞれ異なるものでなければならない。

頂点(p1, p2, p3)のそれぞれにプロパティを適用することで、三角形全体に渡って徐々に変化するプロパティを定義できる。プロパティの補間は線形凸結合で定義される。プロパティ勾配を生成出来ないコンシューマーは p1 プロパティを三角形全体に適用しなければならない。各三角形にプロパティを使用出来ないコンシューマーは三角形のプロパティを無視しなければならない、<object>レベルのプロパティを代わりに使用しなければならない。p1 が指定されていない場合は三角形にプロパティは適用されない。p2 もしくは p3 が指定されていない場合は p1 が三角形全体に使用される。

プロパティグループが object レベルで定義されたプロパティグループと異なる場合、それは pid 属性で指定される。これは三角形全体に適用されるため、3 つのプロパティには暗黙的に同じグループ、同じタイプが適用されることを強制する。

もし三角形を定義しているプロパティが<basematerials>グループ ( 第 5 章を参照 ) の場合、基本素材の補完が本コア仕様に定義されていないため、それらは勾配を形成してはならない。

したがって、p1、p2 そして p3 は等価もしくは未指定でなければならない。素材の勾配や補完方法は拡張仕様の中で定義される。

## 4.2. コンポーネント群

### <comonents>要素



<comonents>要素は、現在のオブジェクトを構成するすべてのコンポーネントのコンテナの役割を果たす。[コンポーネント](#)は別のオブジェクトの定義で使われるオブジェクト リソースである。コンポーネントの使用によって、プロデューサーは 3MF ドキュメント全体のサイズを削減できる。例えば、自動車の 3D オブジェクトを表す 3MF ドキュメントがあるとすると、そこには自動車本体を表すコンポーネントのインスタンスが 1 つと、タイヤのオブジェクト リソースの 4 つのコンポーネントのインスタンスを使用する自動車オブジェクトがあり、車輪は ( transform により ) 必要な回転と移動を適用されたあとで自動車本体上に配置されているだろう。

3D 製造機器はコンポーネント オブジェクトの相対位置を配慮しなければならない。そしてそれらをドキュメントで明示された場合を除きお互いの相対位置を変換してはならない。

整数値の桁あふれを防ぐために、components 要素が含むコンポーネントの数は  $2^{31}$  未満でなければならない。

4.2.1. コンポーネント

<component>要素

図

componentCT\_Component

CT\_Co...

@ objectidST\_ResourceID

@ transformST\_Matrix3D

@? [anyAttribute]##other

属性

名前	型	使用	デフォルト値	固定値	注記
objectid	ST_ResourceID	必須			id 属性値と合致するオブジェクト リソースの参照。
transform	ST_Matrix3D				出力されるアイテムに適用する変換行列（ §3.3 3D 変換行列を参照 ）。

component はその属性で指定されたオブジェクト リソースに、属性で与えられた行列変換を適用し、現在のオブジェクトの定義に追加する。この複数の基本コンポーネントからオブジェクト定義を合成することで、非常に複雑なモデルのファイルサイズをかなり小さくすることができる。シンプルなパーサーを使うことを踏まえて、プロデューサーはオブジェクトをそれらがコンポーネントとして参照されるより前に定義しなければならない。

第 5 章. 素材リソース

<basematerials>要素

図						
属性	名前	型	使用	デフォルト値	固定値	注記
	id	ST_ResourceID	必須			(この仕様のエクステンション内の要素も含めて) 全素材グループ内で一意な ID。

<basematerials>要素は基本の素材群のコンテナとして振る舞う素材グループである。これらの要素はその順序が暗黙的に 0 から始まるインデックスを構成し、<object>要素や<triangle>要素などから参照する際に使用される。

リソース以下の<any>要素の位置に、プロパティグループ要素の他の型をこの仕様のエクステンションとして追加できる。そのようにして定義されるプロパティグループはそれぞれ ID を持たなければならず、またその ID は Model 配下のすべてのリソース内で一意でなければならない。これらのグループは、多くの利用可能なエクステンションが異なる型のプロパティを分離したり組み合わせることを可能にする。

整数の桁あふれを避けるため、素材グループに含まれる素材の数は 2^31 個以下でなければならない。

5.1. ベース素材

<base>要素

						
属性	名前	型	使用	デフォルト値	固定値	注記

	name	xs:string	必須	素材名を指定することで設計の意図を伝え、ユーザーがプリント素材との対応付けを容易にできるようにする。
	displaycolor	ST_ColorValue	必須	素材を描画するための sRGB カラーを指定する。

ベース素材はモデル内のあるオブジェクトを製造するのに使用する特定の**素材**を定義するために指定される。特に、サポートオブジェクトはしばしばモデルのそれ以外の部分とは異なる素材で構築される。

これらの素材は<object>レベルと<triangle>レベルの両方に適用できるように、技術的にはオブジェクト表面の素材だけを指定する。ただし表面が指定した素材であるかぎりにおいて、コンシューマーは素材が体積全体にわたって使用されることを選択してもよい。

ベース素材の名前は設計の意図を明確にするものであるため、プロデューサーはマシン固有の名前を付けることを避け、より汎用的な説明になるようにする**必要がある**。プリンタ固有の情報についてはこれらのベース素材と実際のプリント素材の対応付けを指定する PrintTicket 内に保持することが想定されている。ベース素材名は 3MF パッケージ全体にわたって一意である**必要がある**。

#### 5.1.1. sRGB Color

sRGB カラーは 6 桁または 8 桁の 16 進数で値を指定しなければならない。例えば"RRRGGBB"または"RRGGBBAA"などの値をとる。なお、ここで RR は 00 から FF の値を持つ赤チャンネル、GG は 00 から FF の値を持つ緑チャンネル、BB は 00 から FF の値を持つ青チャンネル、AA は 00 (完全に透明) から FF (完全に不透明、指定されていない場合のデフォルト値) の値を持つアルファチャンネルとする。

**メモ** : 3MF では World Wide Web コンソーシアム規定の sRGB を使用する  
( <http://www.w3.org/Graphics/Color/sRGB> )

## 第 6 章. 3MF パッケージの特徴

3MF ドキュメントでは OPC ( Open Packaging Conventions ) で定義されたパーツを利用してパッケージのコンテンツに関する追加情報を指定する。

### 6.1. サムネイル

3MF ドキュメントのプロデューサーは 3D ペイロードの内容を表す 2D のサムネイル画像を含めてもよい。この画像はコンテンツ タイプとして image/jpeg または image/png を持ち、次のサブセクションで説明される内部ファイル フォーマットの必要条件にしたがう。

このサムネイルはパッケージ ルートからサムネイル画像へのリレーションシップを持つ。このリレーションシップのタイプは次のとおりでなければならない。

<http://schemas.openxmlformats.org/package/2006/relationships/metadata/thumbnail>

サムネイルはオペレーティング システムのファイル エクスプローラーやモデル ブラウザ、プリント キューのプレビュー ユーティリティなどの外部システムからアクセスして表示できる。

#### 6.1.1. JPEG 画像

JPEG image パーツは JPEG の仕様を満たす画像を含まなければならない。コンシューマーは APP0、APP2、APP13、APP14 マーカーを含む JPEG 画像をサポートする必要がある。コンシューマーは APP1 マーカーを含む JPEG 画像をサポートしなければならず、また EXIF の色空間情報を正しく解釈しなければならない。

表 2-3. サポートされる JPEG APPn マーカー

APPn マーカー	出典
APP0	JFIF 仕様
APP1	JEITA によって定義された EXIF 拡張
APP2	ICC 仕様によって定義された ICC プロファイルマーカー
APP13	Photoshop 3.0 拡張

APP14	PostScript レベル 2 拡張内の Adobe DCT フィルター
-------	---------------------------------------

CMYK JPEG 画像は使用してはならない。

**メモ：** APP2 マーカーと APP13 マーカーの両方が指定されていた場合、APP2 マーカーが優先される。

### 6.1.2. PNG 画像

PNG image パーツは PNG 仕様を満たす画像を含まなければならない。

表 2-4 PNG 補助チャンクのサポート

チャンク	サポートレベル
tRNS	サポートしなければならない
iCCP	サポートしなければならない
sRGB	無視しなければならない
cHRM	無視しなければならない
gAMA	無視しなければならない
sBIT	無視しなければならない

## 6.2. コア・プロパティ

3MF ドキュメント フォーマットは OPC 仕様で定義されている [Core Properties](#) パーツに依存する形で、作成日時、更新日時、作者、検索キーワード、その他のドキュメント中心メタデータに関するメタデータを提供している。さらに詳細が必要な場合は OPC 仕様を参照。

## 6.3. 電子署名

3MF ドキュメントは[電子署名](#)を一つ以上含むことがある。電子署名は署名後にその 3MF ドキュメントが改ざんされていないことを検証する手段を提供する。電子署名がなかったとしても 3MF ドキュメントがかつて署名されたことがないとみなすことはできない。

OPC 仕様には OPC パッケージ内で電子署名がどのように適用されるかに関する情報の詳細がすべて記述されている。Digital Signature Origin パーツは 3MF ドキュメント内の電子署名ペイロードの起点としての機能を果たす。個別の Digital Signature パーツは Digital Signature Origin パーツからのリレーションシップを通じて発見することができる。それぞれの Digital Signature パーツにはインラインのデジタル証明書がパッケージ内の外部 Digital Signature Certificate パーツへの参照がある。

3D Model パーツに適用される電子署名には 3D Model パーツとそれにより参照されるその他のパーツと関係するリレーションシップだけを含む必要がある。Thumbnail パーツや Digital Signature Origin パーツ、Core Properties パーツ、関係するリレーションシップのパーツを含んでもよい。

### 6.3.1. 正規化

3D Model パーツのコンテンツの電子署名を適用または検証する前に、それを含む XML をまず正規化しなければならない。これは XML C14N の要件に準ずる XML 正規化と同等なものである。

---

## 6.4. 保護されたコンテンツ

この仕様自体にはコンテンツ保護の仕組みが一切含まれない。しかし、いくつかのワークフローにおいて（例えば、3ML ドキュメントがスタンドアロンファイルとして保存される場合など）、製作または配布された地点から認可された機器上の対象利用者または認可されたアプリケーションやユーザーの場所まで送信される間にそれを傍受した第三者が 3D オブジェクトの詳細にアクセスすることを防ぐことが重要であることは認められている。

Open Packaging Conventions 仕様は OPC パッケージ内でコンテンツをどのように保護するかについての詳細を提供している。

上記の手順を反転することでコンテンツの保護を解除する権限のあるコンシューマーは、（プロデューサーであるかどうかに関わらず）[保護権者](#)による（ドキュメントまたはプログ



ラムを通じた ) 許可なしに、そのコンテンツを上書き保存してはならず、またそのコンテンツを ( ファイルフォーマットによらず ) 保護されない形で保存できてはならない。

## パートⅡ. 付録

## 付録 A. 用語

**3D Model パーツ**： 3D モデルを含む OPC パーツ

**3D Texture パーツ**： 3D モデル内で 3D オブジェクトに複雑な情報を適用するために使用されるファイル ( この仕様では未定義だが、エクステンションによって利用可能 )

**3D 変換行列**： 3 次元空間でオブジェクトの回転、拡大縮小、平行移動に使用される 4x4 のアフィン行列

**3D モデル**： 出力するモデルを定義したマークアップ

**3MF**： この仕様で記述される 3D Manufacturing Format で、物理的な形で出力することを意図された 1 つ以上の 3D オブジェクトを定義する

**3MF ドキュメント**： 3MF 仕様にしたがった 3D ペイロードを含む OPC パッケージのデジタルな形式

**3MF ドキュメント StartPart リレーションシップ**： パッケージのルートから 3D Model パーツへの OPC リレーションシップ

**PrintTicket**： PrintTicket パーツのコンテンツ。Print Schema 仕様準拠。可能であれば 3D 製造のための標準プリント スキーマ キーワードを使用することが推奨される。

**PrintTicket パーツ**： PrintTicket パーツは 3MF ドキュメントがプリントされる際に使用される設定を提供する

**XML 名前空間**： XML 名前空間仕様にしたがって<model>要素で宣言される名前空間

**エディター**： 3MF ドキュメントを読み込んだり、3MF ドキュメントに書き込んだり、場合によってはそれら両方を使用してコンテンツを変更したりするソフトウェアまたはサービス、機器

**オブジェクト リソース**： 3D 製造機器で出力するか他のオブジェクト リソース内でコンポーネントとして利用できる単一の 3D 図形

**組立て**： 3D 製造プロセス中または完了後に接続されているか、またはバラバラにできない組み合わせられた形で連結される 2 つ以上の独立して定義されたオブジェクトを含むモデル。

組み立ては分解できる場合もあれば、独立した部分は分割不能な形で連結されている場合もある。

**後面：** プリント領域のプリント可能な XZ 平面の最大値か、または出力座標空間に変換された後のモデルのバウンディング ボックスの対応する最大平面

**コアプロパティ：** 作者、最終更新日時などのパッケージ全体に関するメタデータを定義する明確に定義されたパッケージ ( OPC ) プロパティのセット

**コンシューマー：** 3MF ドキュメントを解釈するソフトウェアまたはサービス、機器

**コンポーネント：** 他のオブジェクトの全体的な定義に欠損のない形状として追加されるオブジェクト

**サポート：** 多くの 3D プリンタはモデル内のオーバーハングした領域が崩れたり歪んだりしないようにする土台が必要である。この土台はサポートと呼ばれる

**サムネイル：** JPEG 画像または PNG 画像で表される、エンドユーザーがパッケージのコンテンツを特定する助けになる画像

**上面：** プリント領域のプリント可能な XY 平面の最大値か、または出力座標空間に変換された後のモデルのバウンディング ボックスの対応する最大平面

**前面：** プリント領域のプリント可能な XZ 平面の最小値か、または出力座標空間に変換された後のモデルのバウンディング ボックスの対応する最小平面

**素材：** オブジェクトを出力する際に使用できる物理的な物質に関する記述

**素材リソース：** オブジェクトが何でできているかを記述するためにオブジェクトによって参照される可能性のある潜在的なリソース

**底面：** プリント領域のプリント可能な XY 平面の最小値か、または出力座標空間に変換された後のモデルのバウンディング ボックスの対応する最小平面

**電子署名：** 存在する場合、電子署名が適用されてからパーツまたはペイロードが改ざんまたは変更されていないことを検証するために使用できる機構

**塗りつぶしルール：** 特定の座標が形状の「内側」にあるとみなせるかどうかを決定するために使用されるアルゴリズム

**左側面：** プリント領域のプリント可能な YZ 平面の最小値か、または出力座標空間に変換された後のモデルのバウンディング ボックスの対応する最小平面

**プライマリ 3D ペイロード ルート：** 3MF ドキュメントの StartPart リレーションシップによって発見される 3D ペイロード

**プロデューサー：** 3MF ドキュメントを作成するソフトウェアまたはサービス、機器

**ペイロード：** パッケージ内で相互依存するパーツとリレーションシップの完全な集合

**保護権者：** 保護タイプで記述される保護機構の所有者。これははじめにファイルを保護したプロデューサーであることもあれば、どの保護機構が受け入れ可能かを定義したコンシューマーやデジタル著作権管理サービスなどの第三者保護サービスであることもある。

**保護タイプ：** 3MF ドキュメントのリソースとテクスチャに適用されている保護機構の正式名称。XOR 難読化と同程度に単純なこともあれば完全なデジタル著作権管理と同じように複雑なこともある。妥当な保護タイプはこの仕様では定義されず、実装依存となる。

**保護名前空間：** 保護タイプと関連するメタデータが定義される XML 名前空間。<model>要素で宣言される。

**右側面：** プリント領域のプリント可能な YZ 平面の最大値か、または出力座標空間に変換された後のモデルのバウンディング ボックスの対応する最大平面

**メッシュ：** オブジェクトの表面を三角形で分割したもの

**メタデータ：** 詳しいプロデューサーまたはコンシューマーがより正確に処理することを可能にするための、モデルのある部分に関する補助的な情報

**モデル：** 単一ジョブの一部として製造されるオブジェクトのセット。単一のオブジェクト、同一オブジェクトの複数インスタンス、異なる複数のオブジェクト、組立てに使用される複数のオブジェクトなどがありえる

**リソース：** オブジェクト、素材、またはそれら以外のもので他のリソースで利用できるものか、造形指示にしたがって物理的な 3D オブジェクトを造形するために必要となりうるエクステンション内で定義されるもの



## 付録 B. 3MF XSD スキーマ

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.microsoft.com/3dmanufacturing/core/2015/02"
targetNamespace="http://schemas.microsoft.com/3dmanufacturing/core/2015/02" 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:
    ]]></xs:documentation>
  </xs:annotation>
```

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\_Model">

<xs:sequence>

<xs:element ref="metadata" minOccurs="0" maxOccurs="2147483647" />

<xs:element ref="resources" />

<xs:element ref="build" />

<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="2147483647" />

</xs:sequence>

<xs:attribute name="unit" type="ST\_Unit" default="millimeter" />

<xs:attribute ref="xml:lang" />

<xs:attribute name="requiredextensions" type="xs:string" />

<xs:anyAttribute namespace="##other" processContents="lax" />

</xs:complexType>

<xs:complexType name="CT\_Resources">

<xs:sequence>

<xs:element ref="basematerials" minOccurs="0" maxOccurs="2147483647" />

<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="2147483647" />

<xs:element ref="object" minOccurs="0" maxOccurs="2147483647" />

</xs:sequence>

<xs:anyAttribute namespace="##other" processContents="lax" />

</xs:complexType>

<xs:complexType name="CT\_Build">

<xs:sequence>

<xs:element ref="item" minOccurs="0" maxOccurs="2147483647" />

</xs:sequence>

<xs:anyAttribute namespace="##other" processContents="lax" />

</xs:complexType>

<xs:complexType name="CT\_BaseMaterials">

<xs:sequence>

<xs:element ref="base" maxOccurs="2147483647" />

</xs:sequence>

<xs:attribute name="id" type="ST\_ResourceID" use="required" />

</xs:complexType>

<xs:complexType name="CT\_Base">

<xs:attribute name="name" type="xs:string" use="required" />

<xs:attribute name="displaycolor" type="ST\_ColorValue" use="required" />

<xs:anyAttribute namespace="##other" processContents="lax" />

</xs:complexType>

<xs:complexType name="CT\_Object">

<xs:choice>

<xs:element ref="mesh" />

<xs:element ref="components" />

```

</xs:choice>
<xs:attribute name="id" type="ST_ResourceID" use="required" />
<xs:attribute name="type" type="ST_ObjectType" default="model" />
<xs:attribute name="matid" type="ST_ResourceID" />
<xs:attribute name="matindex" type="ST_ResourceIndex" />
<xs:attribute name="thumbnail" type="UriReference" />
<xs:attribute name="partnumber" type="xs:string" />
<xs:attribute name="name" type="xs:string" />
<xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>
<xs:complexType name="CT_Mesh">
  <xs:sequence>
    <xs:element ref="vertices" />
    <xs:element ref="triangles" />
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="2147483647" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CT_Vertices">
  <xs:sequence>
    <xs:element ref="vertex" minOccurs="3" maxOccurs="2147483647" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CT_Vertex">
  <xs:attribute name="x" type="ST_Number" use="required" />
  <xs:attribute name="y" type="ST_Number" use="required" />
  <xs:attribute name="z" type="ST_Number" use="required" />
  <xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>
<xs:complexType name="CT_Triangles">
  <xs:sequence>
    <xs:element ref="triangle" minOccurs="1" maxOccurs="2147483647" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CT_Triangle">
  <xs:attribute name="v1" type="ST_ResourceIndex" use="required" />
  <xs:attribute name="v2" type="ST_ResourceIndex" use="required" />
  <xs:attribute name="v3" type="ST_ResourceIndex" use="required" />
  <xs:attribute name="p1" type="ST_ResourceIndex" />
  <xs:attribute name="p2" type="ST_ResourceIndex" />
  <xs:attribute name="p3" type="ST_ResourceIndex" />
  <xs:attribute name="pid" type="ST_ResourceID" />
  <xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>
<xs:complexType name="CT_Components">
  <xs:sequence>
    <xs:element ref="component" maxOccurs="2147483647" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CT_Component">
  <xs:attribute name="objectid" type="ST_ResourceID" use="required" />
  <xs:attribute name="transform" type="ST_Matrix3D" />
  <xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>
<xs:complexType name="CT_Metadata" mixed="true">
  <xs:attribute name="name" type="xs:QName" use="required" />
</xs:complexType>
<xs:complexType name="CT_Item">
  <xs:attribute name="objectid" type="ST_ResourceID" use="required" />
  <xs:attribute name="transform" type="ST_Matrix3D" />
  <xs:attribute name="itemref" type="xs:ID" />
  <xs:anyAttribute namespace="##other" processContents="lax" />

```



```
</xs:complexType>
<!-- Simple Types -->
<xs:simpleType name="ST_Unit">
  <xs:restriction base="xs:string">
    <xs:enumeration value="micron" />
    <xs:enumeration value="millimeter" />
    <xs:enumeration value="centimeter" />
    <xs:enumeration value="inch" />
    <xs:enumeration value="foot" />
    <xs:enumeration value="meter" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_ColorValue">
  <xs:restriction base="xs:string">
    <xs:pattern value="#[0-9|A-F|a-f][0-9|A-F|a-f][0-9|A-F|a-f][0-9|A-F|a-f][0-9|A-F|a-f][0-9|A-F|a-f]([0-9|A-F|a-f][0-9|A-F|a-f])?" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_UriReference">
  <xs:restriction base="xs:anyURI">
    <xs:pattern value="/.*" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_Matrix3D">
  <xs:restriction base="xs:string">
    <xs:pattern value="((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_Number">
  <xs:restriction base="xs:double">
    <xs:whiteSpace value="collapse" />
    <xs:pattern value="((\-\|+)?(((0-9)+(\.[0-9]+)?)(\.[0-9]+))((e|E)(\-\|+)?[0-9]+)?)" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_ResourceID">
  <xs:restriction base="xs:positiveInteger">
    <xs:maxExclusive value="2147483648" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_ResourceIndex">
  <xs:restriction base="xs:nonNegativeInteger">
    <xs:maxExclusive value="2147483648" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ST_ObjectType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="model" />
    <xs:enumeration value="support" />
    <xs:enumeration value="other" />
  </xs:restriction>
</xs:simpleType>
<!-- Elements -->
<xs:element name="model" type="CT_Model" />
<xs:element name="resources" type="CT_Resources" />
<xs:element name="build" type="CT_Build" />
<xs:element name="basematerials" type="CT_BaseMaterials" />
<xs:element name="base" type="CT_Base" />
```

```
<xs:element name="object" type="CT_Object" />
<xs:element name="mesh" type="CT_Mesh" />
<xs:element name="vertices" type="CT_Vertices" />
<xs:element name="vertex" type="CT_Vertex" />
<xs:element name="triangles" type="CT_Triangles" />
<xs:element name="triangle" type="CT_Triangle" />
<xs:element name="components" type="CT_Components" />
<xs:element name="component" type="CT_Component" />
<xs:element name="metadata" type="CT_Metadata" />
<xs:element name="item" type="CT_Item" />
</xs:schema>
```

## 付録 C. 標準名前空間とコンテンツタイプ

---

### C.1 コンテンツタイプ

3D Model      [application/vnd.ms-package.3dmanufacturing-3dmodel+xml](#)

PrintTicket      [application/vnd.ms-printing.printticket+xml](#)

---

### C.2 リレーションシップのタイプ

StartPart      <http://schemas.microsoft.com/3dmanufacturing/2013/01/3dmodel>

Thumbnail      <http://schemas.openxmlformats.org/package/2006/relationships/metadata/thumbnail>

PrintTicket      <http://schemas.microsoft.com/3dmanufacturing/2013/01/printticket>

---

### C.3 名前空間

3D Model      <http://schemas.microsoft.com/3dmanufacturing/core/2015/02>

## 参考文献

### BNF of Generic URI Syntax

"BNF of Generic URI Syntax." World Wide Web Consortium.  
[http://www.w3.org/Addressing/URL/5\\_URI\\_BNF.html](http://www.w3.org/Addressing/URL/5_URI_BNF.html)

### JPEG

Hamilton, Eric. "JPEG File Interchange Format, Version 1.02." World Wide Web Consortium. 1992.  
<http://www.w3.org/Graphics/JPEG/jfif3.pdf>

### Open Packaging Conventions

Ecma International. "Office Open XML Part 2: Open Packaging Conventions." 2006.  
<http://www.ecma-international.org>

### PNG

Duce, David (editor). "Portable Network Graphics (PNG) Specification," Second Edition. World Wide Web Consortium. 2003. <http://www.w3.org/TR/2003/REC-PNG-20031110>

### Print Schema

Microsoft Corporation. "Print Schema." 2006. <http://windowssdk.msdn.microsoft.com/en-us/library/default.aspx>

### RFC 2119

Bradner, S. "Key words for use in RFCs to Indicate Requirement Levels." The Internet Society. 1997.  
<http://www.rfc-editor.org>

### RFC 3066

Alvestrand, H. "Tags for the Identification of Languages." The Internet Society. 2001. <http://www.rfc-editor.org>

### sRGB

Anderson, Matthew, Srinivasan Chandrasekar, Ricardo Motta, and Michael Stokes. "A Standard Default Color Space for the Internet-sRGB, Version 1.10." World Wide Web Consortium. 1996.  
<http://www.w3.org/Graphics/Color/sRGB>

### Unicode

The Unicode Consortium. The Unicode Standard, Version 4.0.0, defined by: The Unicode Standard, Version 4.0. Boston, MA: Addison-Wesley, 2003.

### XML

Bray, Tim, Eve Maler, Jean Paoli, C. M. Sperberg-McQueen, and François Yergeau (editors). "Extensible Markup Language (XML) 1.0 (Fourth Edition)." World Wide Web Consortium. 2006.  
<http://www.w3.org/TR/2006/REC-xml-20060816/>

### XML C14N

Boyer, John. "Canonical XML Version 1.0." World Wide Web Consortium. 2001.  
<http://www.w3.org/TR/xml-c14n>

### XML Namespaces

Bray, Tim, Dave Hollander, Andrew Layman, and Richard Tobin (editors). "Namespaces in XML 1.0 (Second Edition)." World Wide Web Consortium. 2006. <http://www.w3.org/TR/2006/REC-xml-names-20060816/>

**XML Schema**

Beech, David, Murray Maloney, Noah Mendelsohn, and Henry S. Thompson (editors). "XML Schema Part 1: Structures," Second Edition. World Wide Web Consortium. 2004. <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/>

Biron, Paul V. and Ashok Malhotra (editors). "XML Schema Part 2: Datatypes," Second Edition. World Wide Web Consortium. 2004. <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>