

JDMA EPX 仕様書 改定 A 版

(社) 日本金型工業会東部支部 技術委員会 CAD/CAM分科会「3D-CADデータと放電加工の連携について研究・開発WG」(以下放電WG)では、3D-CAD/CAMで作成した放電加工機用電極の設計情報を、スムーズに放電加工用自動プログラミング装置もしくは放電加工機に伝達することを目的として活動してきた。

放電WGでは目的を実現するために、情報伝達のための共通フォーマットとして「JDMA EPXフォーマット」の策定を行った。このフォーマットを広く普及させることにより、各社各様の方法で情報伝達を行った際に想定されるユーザ側の負担や混乱を軽減することができる。

JDMA EPXフォーマットは日本の多くのCAD/CAMベンダ、加工機メーカーに支持されている。様々なCAD/CAM、加工機が混在したユーザにおいても、JDMA EPXフォーマットの利用により、効率的な作業環境を構築できる。

JDMA EPXフォーマットは、XMLの仕様に準拠している。本仕様書では、JDMA EPXフォーマットの構造とタグに関して説明する。以下の説明では、【変数名】がXMLにおけるタグとなっている。それぞれの【変数名】に対して、その説明とサンプル(利用方法)を記述する。

[書き出し]

説明

XML 準拠にするための記述

全体の構造は、header, work_piece_information, edm_group という項目に分かれる。

1 ファイルに、header, work_piece_information が1つずつ存在する。

edm_group は、放電加工部位の数に応じて複数指定可能である。

サンプル

```
<?xml version="1.0" encoding="Shift_JIS" ?>
```

```
<jdma_epx>
```

```
</jdma_epx> . . . (最終行となる)
```

で全ての記述を囲む。

[header]

説明

ファイル全般に関する情報を記述する。

サンプル

```
<header>  
</header>  
でデータを囲む。
```

【変数名】 major_version

説明

EPX のメジャーバージョンを記述する。
伝達項目が増えた場合にバージョンを UP する。互換性の保証なし。

タイプ

整数型

サンプル

```
<major_version>2</major_version >
```

【変数名】 minor_version

説明

EPX のマイナーバージョンを記述する。
不具合の修正に対応した場合にバージョンを UP する。

タイプ

整数型

サンプル

```
<minor_version>0</minor_version >
```

【変数名】 title

説明

出力されるファイルに対して、加工内容を特定するためのタイトルを記述する。

タイプ

文字列型

サンプル

```
<title>キャビティ 部品 1 </title>
```

【変数名】 subtitle

説明

title に補足する情報を記述する。

タイプ

文字列型

サンプル

```
<subtitle>側面 X 側</subtitle>
```

【変数名】 sender

説明

出力するシステムの名称、バージョン情報、会社名を記述する。
ベースCADにアドインして開発したシステムの場合は、ベースCADおよび開発したアドインシステムの名称を記述する。

タイプ

文字列型

サンプル

<sender> (送り側の) ベースCAD名 アドインシステム名 V1.0 会社名</sender>

【変数名】 machine_name

説明

放電加工機の製造会社名、機種名を記述する。

タイプ

文字列型

サンプル

<machine_name>放電加工機メーカー 機種1</machine_name>

【変数名】 date

説明

ファイルを作成した日付を記述する。

タイプ

文字列型

サンプル

<date>20040120</date>

【変数名】 time

説明

ファイルを作成した時刻を記述する。

タイプ

文字列型

サンプル

<time>15:30:35</time>

【変数名】 author_name

説明

作成者が所属する会社名および作成者名を記述する。

タイプ

文字列型

サンプル

<author_name>金型メーカー 金型 太郎</author_name>

【変数名】 unit

説明

放電加工の単位系を指定する。

タイプ

ブール型 (0 = mm, 1 = inch)

サンプル

<unit>0</unit>

【変数名】 comment

説明

コメントを記述する。

タイプ

文字列型

サンプル

<comment> 加工点の総数2 加工点1は電極が2個(荒、仕上げ)、加工点2は電極1個</comment>

[work_piece_information]

説明

加工対象部品に関する情報を記述する。

サンプル

```
<work_piece_information>  
</work_piece_information>  
でデータを囲む。1ファイルに1つだけ存在する。
```

【変数名】 work_piece_name

説明

加工対象部品の名称を記述する。

アセンブリにより加工対象部品が複数の部品で構成されている場合は、メインとなる部品の名称を記述する。

その他の部品名はコメントで記述する。

タイプ

文字列型

サンプル

```
<work_piece_name>対象部品の名前</work_piece_name>
```

【変数名】 work_piece_material

説明

加工対象部品の材質を指定する。

加工対象部品が複数の部品で構成され、それぞれ材質が異なる場合、別々のファイルで出力する必要がある。

タイプ

整数型 (1:鉄 2:アルミ 3:銅合金 4:超硬 99:その他)

サンプル

【変数名】 work_piece_geometry_group

説明

加工対象部品の最終形状の情報をグループにまとめて記述する。

最終形状の情報は段取り作業時の参照用データとして使用する。

count でデータ数を記述する。

サンプル

```
<work_piece_geometry_group count = "2">  
</work_piece_geometry_group>
```

でデータを囲む。

【変数名】 work_piece_geometry_info

説明

加工対象部品の最終形状データとコメントをまとめた情報を記述する。

サンプル

```
<work_piece_geometory_info>  
</work_piece_geometory_info>  
でデータを囲む。
```

【変数名】 work_piece_geometry

説明

加工対象部品の最終形状データのファイル名を記述する。
加工対象部品が複数の部品で構成される場合、最初に定義されたファイルがメイン部品のデータとなる。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<work_piece_geometory_group count = "2">  
  <work_piece_geometory_info>  
    <work_piece_geometry>xyz1.stl</work_piece_geometry>  
    <comment>部品 xyz1 の最終形状データ</comment>  
  </work_piece_geometory_info>  
  <work_piece_geometory_info>  
    <work_piece_geometry>xyz2.stl</work_piece_geometry>  
    <comment>部品 xyz2 の最終形状データ</comment>  
  </work_piece_geometory_info>  
</work_piece_geometory_group>
```

【変数名】 comment

説明

加工対象部品の最終形状データに関するコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>部品 xyz2 の最終形状データ</comment>
```

【変数名】 work_piece_stock_geometry_group

説明

加工対象部品の放電開始前の形状データをグループにまとめて記述する。
放電加工時間の見積りに使用する。
count でデータ数を記述する

サンプル

```
<work_piece_stock_geometry_group count = "2">  
</work_piece_stock_geometry_group>
```

でデータを囲む。

【変数名】 work_piece_stock_geometry_info

説明

加工対象部品の放電加工前の形状データとコメントをまとめた情報を記述する。

サンプル

```
<work_piece_stock_geometry_info>  
</work_piece_stock_geometry_info>
```

でデータを囲む。

【変数名】 work_piece_stock_geometry

説明

加工対象部品の放電開始前の形状データのファイル名を記述する。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<work_piece_stock_geometry_group count = "2">  
  <work_piece_stock_geometry_info>  
    <work_piece_stock_geometry>xyz_stock 1.stl</work_piece_stock_geometry>  
    <comment>部品 xyz_stock 1 の放電加工前形状データ</comment>  
  </work_piece_stock_geometry_info>  
  <work_piece_stock_geometry_info>  
    <work_piece_stock_geometry>xyz_stock 2.stl</work_piece_stock_geometry>  
    <comment>部品 xyz_stock 2 の放電加工前形状データ</comment>  
  </work_piece_stock_geometry_info>  
</work_piece_stock_geometry_group>
```

【変数名】 comment

説明

加工対象部品の放電加工前の形状データに関してコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>部品 xyz_stock 2 の放電加工前形状データ</comment>
```

【変数名】 work_piece_image_group

説明

加工対象部品の画像データをグループにまとめて記述する。
count でデータ数を記述する

サンプル

```
<work_piece_image_group count = "2">  
</work_piece_image_group>
```

でデータを囲む。

【変数名】 work_piece_image_info

説明

加工対象部品の画像データとコメントをまとめた情報を記述する。

サンプル

```
<work_piece_image_info>  
</work_piece_image_info>  
でデータを囲む。
```

【変数名】 work_piece_image

説明

加工対象部品の画像データのファイル名を記述する。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<work_piece_image_group count = "2">  
  <work_piece_image_info>  
    <work_piece_image>aaa1.bmp</work_piece_image>  
    <comment>部品 aaa1 の画像データ</comment>  
  </work_piece_image_info>  
  <work_piece_image_info>  
    <work_piece_image>aaa2.bmp</work_piece_image>  
    <comment>部品 aaa2 の画像データ</comment>  
  </work_piece_image_info>  
</work_piece_image_group>
```

【変数名】 comment

説明

加工対象部品の画像データに関してコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>アセンブリによる加工対象部品です。構成 部品 xyz1 部品 xyz2</comment>
```

【変数名】 work_piece_drawing_group

説明

加工対象部品の図面データをグループにまとめて記述する。
count でデータ数を記述する

サンプル

```
<work_piece_drawing_group count = "1">  
</work_piece_drawing_group>
```


でデータを囲む。

【変数名】 work_piece_drawing_info

説明

加工対象部品の図面データとコメントをまとめた情報を記述する。

サンプル

```
<work_piece_drawing_info>  
</work_piece_drawing_info>  
でデータを囲む。
```

【変数名】 work_piece_drawing

説明

加工対象部品の図面データのファイル名を記述する。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<work_piece_drawing_group count = "1">  
  <work_piece_drawing_info>  
    <work_drawing_image>bbb.dxf</work_drawing_image>  
    <comment>部品 bbb の図面データ</comment>  
  </work_piece_drawing_info>  
</work_piece_drawing_group>
```

【変数名】 comment

説明

加工対象部品の図面データに関してコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>アセンブリによる加工対象部品です。構成 部品 xyz1 部品 xyz2</comment>
```

[edm_group]

説明

同じ放電加工形状単位にグループにまとめて記述する。

サンプル

```
<edm_group>  
</edm_group>
```

でデータを囲む。

【変数名】 edm_detail_info

説明

放電加工形状単位の情報記述する。
edm_group の中に複数定義が可能。

サンプル

```
<edm_detail_info>  
</edm_detail_info>  
でデータを囲む。
```

【変数名】 projection_area

説明

放電加工の進行方向から見た放電加工範囲の投影面積を記述する。
加工点が複数の場合、1箇所あたりの面積を記述する。
単位は<header> <unit>を参照する。 inch², mm²

タイプ

実数型

サンプル

```
<projection_area>15.56</projection_area>
```

【変数名】 finish_mode

説明

加工面の仕上げ方法を指定する。
ファイルの出力段階で特定できない場合、0がデフォルトになる。

タイプ

整数型

(1:標準 2:油鏡面(光沢面) 3:油梨地 4:粉末鏡面(粉末光沢面) 5:粉末梨地 99: その他 0:指定なし)

サンプル

```
<finish_mode>1</finish_mode>
```

【変数名】 spark_mode

説明

加工終了地点に到達までの加工で優先すべき項目を指定する。
ファイルの出力段階で特定できない場合、0がデフォルトになる

タイプ

整数型 (1:標準 2:面品質重視 3:低消耗重視 4:速度重視 0:指定なし)

サンプル

```
<rough_mode>1</rough_mode>
```

【変数名】 roughness_unit

説明

面あらしの表記方法を指定する。
ファイルの出力段階で特定できない場合、0がデフォルトになる。
値が0の場合、roughness_valueの値は意味を持たない。

タイプ

整数型 (1:Rz(μ m) 2:Ra 3:Rmax(μ m) 4:VDI 0:指定なし)

サンプル

```
<roughness_unit>1</roughness_unit>
```

【変数名】 roughness_value

説明

最終放電仕上げ面の表面あらしの値を記述する。
roughness_unitの値が1(Rz)、2(Ra)、3(Rmax)の場合は実数だが、4(VDI)の場合は整数となる。

タイプ

実数型

サンプル

```
<roughness_value>0.9</roughness_value>
```

【変数名】 comment

説明

放電加工形状単位に関するコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>コメント</comment>
```

【変数名】 electrode_group

説明

電極の情報をグループにまとめて記述する。
同種の放電加工条件が使用できる電極を同一電極グループとして記述する。
荒、中仕上げ、仕上げの区別は、同一電極グループの中で記述する。
countでデータ数を記述する。

サンプル

```
<electrode_group count = "2">  
  <electrode_info>  
  </electrode_info>  
  <electrode_info>
```

```
</electrode_info>  
</electrode_group>  
でデータを囲む。
```

【変数名】 electrode_info

説明

電極に関する情報を記述する。
電極の本数に応じて記述する。

サンプル

```
<electrode_info>  
</electrode_info>
```

でデータを囲む。

【変数名】 electrode_name

説明

電極を特定するための記述をする。
ユニークに管理した記述は不要。
記述方式の指定はなし。

タイプ

文字列型

サンプル

```
<electrode_name>103M03-01</electrode_name>
```

【変数名】 electrode_id

説明

1つのEPXファイルの範囲で、個々の電極に対して1つの固有のIDを付与する。
電極の同一性はこの番号によって保証できるものとする。

タイプ

正の整数型

サンプル

```
<electrode_id>13</electrode_id>
```

【変数名】 fluid_mode

説明

液処理の種類を指定する。
ファイルの出力段階で特定できない場合、0がデフォルトになる。

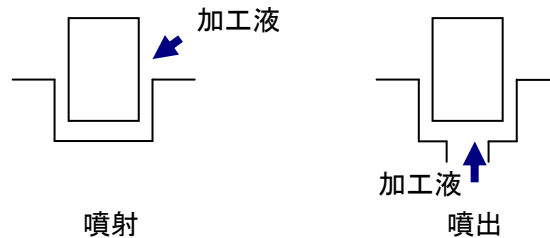
タイプ

整数型 (1:無噴流 2:噴射 3:噴出 4:吸引 0:指定なし)

※噴射 電極周辺部から加工液を噴射する。

※噴出 電極または加工対象部品の中の密閉状態の噴出口から加工液を出す。

※吸引 電極または加工対象部品の中の密閉状態の吸引口から加工液を吸引する。



サンプル

```
<fluid_mode>1</fluid_mode>
```

【変数名】 edm_process_type

説明

荒、中仕上げ、仕上げなどの放電加工工程の区分を指定する。

タイプ

整数型 (1: 荒 2: 中 3: 仕上げ 99: その他 0: 指定なし)

99 (その他) の場合、コメントに取り決りを記述する。

ファイルの出力段階で特定できない場合、0 がデフォルトになる。

サンプル

```
<edm_process_type>1</edm_process_type>
```

【変数名】 electrode_material

説明

電極の材質を指定する。

タイプ

整数型 (1: Cu 2: Gr(L) 3: Gr(M) 4: Gr(H) 5: CuW 6: AgW 99: その他 0: 指定なし)

99 (その他) の場合、コメントに取り決りを記述する。

ファイルの出力段階で特定できない場合、0 がデフォルトになる。

サンプル

```
<electrode_material>1</electrode_material>
```

【変数名】 orbit_pattern

説明

揺動の運動形状を指定する。

タイプ

整数型 (1: 円 2: 四角 3: 球 4: 揺動なし 99: その他 0: 指定なし)

99 (その他) の場合、コメントに取り決りを記述する。

ファイルの出力段階で特定できない場合、0 がデフォルトになる。

サンプル

```
<orbit_pattern>1</orbit_pattern>
```

【変数名】 under_size_model

説明

加工対象部品の最終形状に対して、形状を減寸する方式を指定する。

減寸とは、電極形状とするために加工対象部品の最終形状を縮小することを意味する。

タイプ

整数型 (1: 3D 方向の減寸方式 2: 平面方向の減寸方式 0: 指定なし)
ファイルの出力段階で特定できない場合、0 がデフォルトになる。

サンプル

```
<under_size_model>1</under_size_model>
```

【変数名】 under_size

説明

減寸の量。
減寸方式が「0:指定なし」の場合、意味を持たない。
単位は<header>の<unit>を参照

タイプ

実数型 (正の数値)

サンプル

```
<under_size>0.15</under_size>
```

【変数名】 magazine_pot_number

説明

電極交換装置のマガジンプット番号を指定する。
0 は指定なし。

タイプ

整数型

サンプル

```
<magazine_pot_number>13</magazine_pot_number>
```

【変数名】 electrode_geometry_group

説明

電極形状の情報をグループにまとめて記述する。
count でデータ数を記述する。

サンプル

```
<electrode_geometry_group count = "1">  
  <electrode_geometry_info>  
</electrode_geometry_info>  
  <electrode_geometry_info>  
</electrode_geometry_info>  
</electrode_geometry_group>
```

でデータを囲む。

【変数名】 electrode_geometry_info

説明

電極形状とコメントの情報を記述する。

サンプル

```
<electrode_geometry_info>  
</electrode_geometry_info>
```

でデータを囲む。

【変数名】 electrode_geometry

説明

電極形状データのファイル名を記述する。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<electrode_geometry>t1.stl</electrode_geometry>
```

【変数名】 under_size_electrode_geometry

説明

減寸された電極形状データのファイル名を記述する。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<under_size_electrode_geometry>t1.stl</under_size_electrode_geometry>
```

【変数名】 comment

説明

電極形状データに関するコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>減寸量を考慮済みのモデルデータ</comment>
```

【変数名】 electrode_image_group

説明

電極形状イメージの情報をグループにまとめて記述する。

サンプル

```
<electrode_image_group>  
  <electrode_image_info>  
  </electrode_image_info>  
</electrode_image_group>
```

でデータを囲む。

【変数名】 electrode_image_info

説明

電極形状イメージとコメントの情報を記述する。

サンプル

```
<electrode_image_info>  
</electrode_image_info>
```

【変数名】 electrode_image

説明

電極形状イメージデータのファイル名を記述する。
ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<electrode_image>ttt2.bmp</electrode_image>
```

【変数名】 comment

説明

電極形状イメージデータに関するコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>コメントを記述</comment>
```

【変数名】 electrode_drawing_group

説明

電極形状図面の情報をグループにまとめて記述する。
count でデータ数を記述する。

サンプル

```
<electrode_drawing_group count = "1">  
</electrode_drawing_group>
```

【変数名】 electrode_drawing_info

説明

電極形状図面とコメントの情報を記述する。

サンプル

```
<electrode_drawing_info>  
</electrode_drawing_info>
```

【変数名】 electrode_drawing

説明

電極形状図面データのファイル名を記述する。

ファイルのタイプは指定なし。

タイプ

文字列型

サンプル

```
<electrode_drawing>ttt3.dxf</electrode_drawing>
```

【変数名】 comment

説明

電極形状図面データに関するコメントを記述する。

タイプ

文字列型

サンプル

```
<comment>コメントを記述</comment>
```

【変数名】 position_group

説明

加工点の情報をグループにまとめて記述する。
count でデータ数を記述する。

サンプル

```
<position_group count = "1">  
<position_group>
```

でデータを囲む。

【変数名】 position_detail_info

説明

加工点に関する情報を記述する。

サンプル

```
<position_detail_info count = "1">  
<position_detail_info.>
```

でデータを囲む。

【変数名】 initial_pos_group

説明

加工初期位置の点座標情報をグループにまとめて記述する。
count でデータ数を記述する。

サンプル

```
<initial_pos_group count = "1">  
< initial_pos_group>
```

でデータを囲む。

【変数名】 initial_position

説明

加工初期位置の点座標情報を記述する。

サンプル

```
<initial_position>  
</initial_position>
```

でデータを囲む。

【変数名】 X

説明

加工初期位置の X 座標値を記述する。

タイプ

実数型

サンプル

```
<X>0.0</X>
```

【変数名】 Y

説明

加工初期位置の Y 軸座標値を記述する。

タイプ

実数型

サンプル

```
<Y>0.0</Y>
```

【変数名】 Z

説明

加工初期位置の Z 軸座標値を記述する。

タイプ

実数型

サンプル

```
<Z>30.0</Z>
```

【変数名】 C

説明

加工初期位置の C 軸座標値を記述する。
電極の基準座標値を基に記述する。
反時計回りが C 軸の + 方向となる。

タイプ

実数型

サンプル

```
<C>30.0</C>
```

【変数名】 start_position

説明

加工開始位置の点座標情報を記述する。

サンプル

```
<start_position>  
</start_position>  
でデータを囲む。
```

【変数名】 X

説明

加工開始位置の X 軸座標値を記述する。

タイプ

実数型

サンプル

```
<X>0.0</X>
```

【変数名】 Y

説明

加工開始位置の Y 軸座標値を記述する。

タイプ

実数型

サンプル

```
<Y>0.0</Y>
```

【変数名】 Z

説明

加工開始位置の Z 軸座標値を記述する。

タイプ

実数型

サンプル

```
<Z>30.0</Z>
```

【変数名】 C

説明

加工開始位置の C 軸座標値を記述する。

タイプ

実数型

サンプル

```
<C>30.0</C>
```

【変数名】 end_position

説明

加工終了位置の点座標情報を記述する。

サンプル

```
<end_position>  
</end_position>  
でデータを囲む。
```

【変数名】 X

説明

加工終了位置の X 軸座標値を記述する。

タイプ

実数型

サンプル

```
<X>0.0</X>
```

【変数名】 Y

説明

加工開始位置の Y 軸座標値を記述する。

タイプ

実数型

サンプル

```
<Y>0.0</Y>
```

【変数名】 Z

説明

加工終了位置の Z 軸座標値を記述する。

タイプ

実数型

サンプル

```
<Z>30.0</Z>
```

【変数名】 C

説明

加工終了位置の C 軸座標値を記述する。

タイプ

実数型（10進角度、0° 10′ 20″ などの表記は無効）

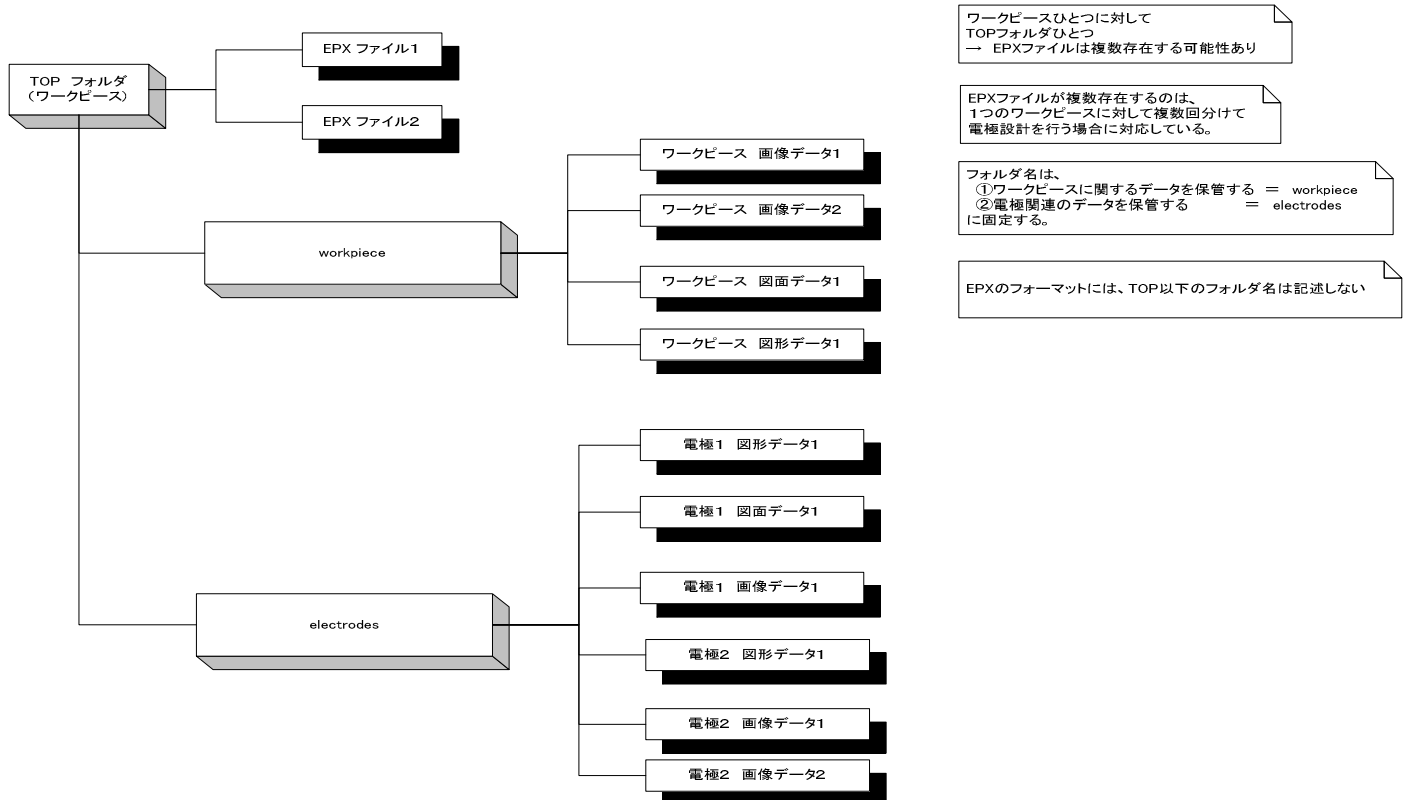
サンプル

```
<C>30.0</C>
```

[出力データのディレクトリ構成について]

「JDMA EPX」ファイルを出力する場合、同時に形状データ、画像データおよび図面データを出力することになる。

これらのファイルと「JDMA EPX」ファイルは、パソコンなどが変更されても参照可能な状態を保つ必要がある。このために、以下のような相対的な位置関係を保つよう出力することとする。



[注記]

- ・「指定なし」は、ファイルの出力時点で決定できないときに利用する。
「指定なし」が存在する変数では、デフォルトとして必ず入力する。
- ・バージョン2では、座標系を絶対座標系とする。
- ・データがない場合、グループの値は0とする。タグは記述し値を空欄とする。
- ・C軸の回転角度と揺動の方向の同期に関しては、放電加工自動プロ側で判断する。

検討事項

- ・加工点ごとに異なる加工対象部品の材質を定義することは、バージョン3以降で検討
- ・座標系の定義に関しては、バージョン3以降で検討
- ・Raの単位および表面あらさに関するインチ系単位指定に関しては、バージョン3以降で検討

[改定記録]

2004.7.7 初版発行

2013.1.10 改定A版発行

- ・本改訂は、初版の記載誤りと文章の不明確点を修正する目的で実施するため、EPX規格のバージョン番号は修正しない。
- ・<work_piece_image_info>タグの解説追加（初版での抜け）
- ・<work_piece_drawing_info>タグの解説追加（初版での抜け）
- ・<electrode_id>タグの説明の変更（意味の明確化）
- ・<under_size>タグのタイプの変更、実数型（正の数値）とする。
- ・<initial_pos_group>タグのタグ名記載誤りを修正
- ・検討事項から、「荒加工と仕上げ加工が一体になった電極はバージョン3以降で検討」という記載を削除。バージョン2でも表現できるという結論に至ったため。