

お手本データ変換検証 STEP AP242XML 書式標準書

I. 背景

JAMA/JAPIA DE データ流通改革タスク 3D 図面普及促進チームでは、各 OEM から仕入先への DE(デジタルエンジニアリング)データ流通を想定した CAD データの変換流通検証を、タスクチーム参加各社からデータ提供をうけて実施。CAD データを流通フォーマットに変換する検証とともに、STEP AP242 XML の作成検証も実施している。2022 年度の変換流通検証において、STEP AP242XML の記述の多くが、ユーザ定義として記述される等、課題が多くあることがわかった。

23 年度変換流通検証では、今後、各 OEM/仕入先が実運用する上でのお手本となるべき STEP AP242 XML データを作成・公開することで、各社の DE データ流通改善の一助としていただく。

II. 目的

ベンチマークモデル変換検証において、変換する CAD のデータの CAD プロパティに入力されている情報を、ISO 10303-4442、PDM-IF Recommended Practices に準拠した XML 形式に変換する。本書は、それらに記載されている事項をまとめたものである。一部、Recommended Practices で out of scope とされた項目についても、実施案として記載している。

今後、各社で STEP AP242XML を活用する際、基準となるべき XML の書式となることを期待している。その活用により、XML の標準化を推進し今後一層の普及・活用を目指す。

III. 変更履歴

修正日時	Version	修正・検討内容
24/3/31	1.0	正式発行

IV. 検証対象

22 年度変換流通検証で使用した各社の CAD データのプロパティに入力されていた値を参考に、下記の XML 変換書式の標準を示す(表 1)。

表 1. 変換検証する値

番号	項目	他の呼び名
1	部品番号	品番、部番
2	部品名称	
3	イベント	
4	Version	Revision
5	ステイタス	
6	設計者	
7	設計部署	
8	設計者コンタクト先	電話番号等
9	出図日時	
10	設変番号	
11	設変日時	
12	材質	
13	密度	
14	板厚	
15	質量	
16	体積	
17	寸法公差	
18	質量公差	
19	モデルグレード	
20	図表	
21	組立/単品	
22	CAD データの座標	

参考

- ✓ 今回の XML 書式標準では、PDM 連携に関連する XML 書式については触れていない。
- ✓ JAMA/JAPIA DE 流通流通改革タスク3D図面普及促進チームでは、組立部品情報や、CADデータ内の幾何公差等の PMI 情報出力等の検討も実施しており、今後、こうした情報の XML 標準化にも取り組む。
- ✓ PDM 情報からの情報流通も考えられるが、具体的な運用方法は未検討。本書では、情報を取得できたことを前提に、その XML 書式を検討する。

V. XML 書式

- ✓ 下記、「RP」とは、CAx Interoperability Forum(CAx-IF)で発行された「PDM-IF Recommended Practices for STEP AP242 Edition 2 Minor Revision Domain Model XML Product & Assembly Structure Release 3.0」(2021 年 11 月 23 日発行)の略である。現在、2021 年 11 月 23 日発行より、更に新しい版が発行されている。
- ✓ Recommended Practices は、out of scope とされて詳細言及されていない項目も多数あり、検討途中の項目も多い。ISO10303-4442、Recommended Practices の内容だけでは、まだ、現実的な運用が困難だと思

われる項目もある。そのような項目も、言及されている ISO10303-4442、Recommended Practices の内容を記述した。今後の、Recommended Practices の開発が待たれる。

1. 部品番号

① 用語定義

- ✓ 「部品番号」(品番)とは、その部品に付与された番号。会社内で、BOM、PDM や、様々な業務に応じたリスト同士を紐づけるキーとなる番号である。PDM などシステム内で、ユーザの目のつかない場所で、もっぱらシステム内の管理のために、部品に付与する番号(インスタンス番号)ではない。
- ✓ 部品番号は、一つの会社内では共通の番号である。一般には、会社が異なると、異なる部品番号が付与される。

② XML 書式

<Part>テンプレート

└<Id>属性

└└<Identifier>テンプレート

└└└<id>属性 (A 社の品番)

└└└<idRoleRef>属性 (A 社の開示範囲)

└└└<idContextRef>属性" (A 社の定義)

└└<Identifier>テンプレート

└└└<id>属性 (B 社の品番)

└└└<idRoleRef>属性 (B 社の開示範囲)

└└└<idContextRef>属性" (B 社の定義)

- ✓ RP の用例は下記(RP 5.1.1 Part)

```
<Part uid="p--0000000017086CB0">
  <Id>
    <Identifier uid="pid--0000000017086CB0--id1" id="as1"
      idRoleRef="rl--ii" idContextRef="o--000000178"/>
  </Id>
</Part>
```

③ XML 書式解説

- ✓ 「部品番号」は、<part> <id> <Identifier> <id> <Identifier>で記述される(RP 5.1.1 Part)。<Identifier>の用法は、<Identifier>配下に、<idRoleRef>と、<idContextRef>を持つ(RP4.4.6 Identifier)
- ✓ 会社によって異なる複数の品番を記述する際には、<Identifier>を 2 つ定義し、<idContextRef>で会社を指示する。
- ✓ <idRoleRef>は、役割を示す。<idRoleRef> が参照する先には文字列には、2 つの表示がある。それぞれの「役割」の意味は次の通り。
 - "identification information": ID が Part をユニークに特定するための役割を持つ。
 - "exchange identification information": ID が組織間データ交換の文脈で Part をユニークに特定するための役割を持つ。
- ✓ <idContextRef>は、会社など、所属先が記載される。

2. 部品名称

① 用語定義

- ✓ 「部品名称」とは、その部品の名称。
- ✓ 「正式な」部品名称は部品番号と、一対一対応する。しかし、実運用上、しばしば長い文字列であるために、部品名称は略される場合がある。そのため、実データ上では、必ずしも一対一対応するとは限らない。マシンリーダブルで活用されることは少なく、人間がその部品を確認しやすくするために用いられることが多い。

② 仕様要望

- ✓ 部品番号と同様、会社ごとに異なる複数の部品名称を定義できるようにする

23 年 6 月、CAx-IF に対し要望。RP に追加するとの回答あり。

③ XML 書式

RP5.1.1 における部品名称の定義は下記

<Part>テンプレート

└<Name>属性

└<Description>テンプレート

部品番号の情報を参照するため、下記のような記述を可能としたい。

<Part>テンプレート

└<Name>属性

└<Descriptor> (A 社の部品名称と関連付ける)

└<Name>属性

└<Descriptor> (B 社の部品名称と関連付ける)

④ XML 書式解説

- ✓ RP 5.1.1 Part では、「部品名称」は、<part> <id> <Identifier> <Name> <Description>で記述される。
- ✓ 上記例では、<Descriptor>を使用して、品番と紐づけている。
- ✓ JAMA/JAPIA の部品名称の運用としては、英語表記が望ましい。

3. イベント

① 用語定義

- ✓ 車両開発進捗の期間に応じて、「開発フェーズ」が設定される。その開発フェーズの設計検討成果物として、フェーズの最後に、設計出図がなされる。多くの場合、車両全体が把握できるレベルの規模で、複数の設計部署がタイミングを調整し、辻褄のあったデータを出力する。この行為を「イベント」定義する。
- ✓ DMU に使用するデータのための「DMU イベント」、部品を仕入先に発注するための「正式図イベント」等が存在する。
- ✓ 「正式図イベント」後に、設計が、幾つかの部品の設計変更で CAD データが出図される場合は、車両全体が把握出来るほどの辻褄のあうデータが出図されないため、規模の点で、通常、イベントとは呼ばない。

② 仕様要望

要件：車両開発プロセスは、各社それぞれ異なっている上に、プロセスは進化しつづける。各 OEM 内でも、車種によって開発プロセスは異なるかもしれない。イベント設定は柔軟に設定できるようにする必要がある。

- ✓ イベントは、各社の開発プロセスそのものであり、各社ごと異なる。
そのため、「〇〇社の△△イベント」と会社名とセットでの定義できるようにする。

23 年 6 月、CAx-IF に対し要望。9 月に回答をいただいている。今後、ユースケース等詳細検討の必要あり。

③ XML 表記

ISO 10303-4442 4.2.4.2.22 Event より作成。

```
<DateTimeAssignmentSelect>  
└<GeneralDateTimeAssignmentSelect>  
  └<DateTimeAssignment>  
    └<DateAndPersonAssignment>  
      └<DateAndPersonOrganization>  
        └<EventAssignment>  
          └<Event>  
            └<EventRelationship>  
              └<Duration> ...間隔
```

上記<Event>は、部品ごとに定義されるのではなく、部品とは別の場所に定義される。< EventRelationship >により、部品と紐づいている。部品側の記述は下記と思われる。

```
<Part>  
└<Versions>  
  └<PartVersion>  
    └<EventAssignment>
```

④ XML 書式解説

- ✓ ISO 10303-4442 4.2.4.2.22 Event のマップによると、<DateTimeAssignmentSelect >がトップで、<Event>が定義される。

今後の検討項目

イベント表示のユースケースの検討が必要。また、下記の管理目的で使用が可能か、確認が必要だと思われる。

- (ア) 仕入先・後工程が、データを検討する上で、検討しても良いデータか、判断できることが必要。そのために、開発フェーズが判るイベント表示が必要である。例えば、「「DMU イベント」のデータであれば、正式図前のデータなので型製作は見合わせよう」等の判断ができる。
- (イ) 正式図前後で、version 管理が異なる可能性がある。version 管理のリセットが可能。

4. Version

① 用語定義

CAD の version 付与は、様々なレベルが想定される。ユースケースの検討が必要。大日程レベルの管理は、<Event>とともに

(1) 大日程レベル version 管理

車両開発フェーズ毎に、異なる version の定義が必要になる可能性あり。

正式図出図後に、設変とともに、CAD version 管理が必要になる。更なる管理が必要か？ → 検討
必要項目参照

(2) 日常の作業レベルでの version 管理

今後の検討項目

version 管理のユースケース検討。

- ・ 正式図前のヴァーチャル検討時の version 管理の必要性
- ・ 先行試作等での version 管理の必要性
- ・ 日常の作業レベル管理。日常の作業レベルでの version を XML に明示する必要があるか？ PDM システム内では、version 管理されているが、DE データ流通として、その version 管理情報を管理し、XML 出力するニーズの確認。

② XML 表記

<Part>

└<Versions>

└└<PartVersion>

└└└<Id>属性

└└└└<Identifier>テンプレート

└└└└└<id>属性 (A 社の version)

└└└└└└<Descriptor>テンプレート (A 社の部品名称と関連付ける)

└└└└└└└<フェーズ定義のタグ>属性 (仕様要望)

└└└└└└└└<Identifier>テンプレート

└└└└└└└└└<id>属性 (B 社の version)

└└└└└└└└└└<Descriptor>テンプレート (B 社の部品名称と関連付ける)

└└└└└└└└└└└<フェーズ定義のタグ>属性 (仕様要望)

③ XML 書式解説

- ✓ Part の version 番号は、<part> <PartVersion> <id> <Identifier> <id> で記述され、<Id>は、<Identifier>(4.6.6 Identifier)を参照(RP 5.1.2 PartVersion)。
- ✓ 部品番号は、会社ごとに定義される。Version は部品番号に付与される情報であるので、必然的に、version 情報も会社ごとに定義できるようになっている必要がある。部品名称と同様に、Part と紐づく。
- ✓ <PartVersion>には、<PartVersionRelationship>が定義されている(RP5.1.2 PartVersion table25)。
<PartVersionRelationship>は、(RP5.1.5 PartVersionRelatioinship)で定義されており、ふたつの <PartViersion>間の関連付けを行う。(RP5.1.5 PartVersionRelatioinship)の表によると、<

PartVersionRelationship>の用例は 2 つ挙げる。

- 一つの部品が、複数の version に分かれた時。(A 案/B 案で検討が進められるような場合か)
- OEM では単品扱いであるが、仕入先では ASSY 部品である場合、それぞれの部品構成にリンクを持たせる。

5. ステータス

① 用語定義

「ステータス」とは、そのデータの「承認」状態を示す。

② XML 表記

<Part>

└<Versions>

└└<PartVersion>

└└└<ApprovalAssignment>

└└└└<AssignedApproval>

└└└└└<id>属性 (承認情報の uid)

<Approval>テンプレート

└<Status>

└└<ClassString>

└└└<OrganizationOrPersonInOrganizationAssignment>

└└└└<Role> (承認するという役割表記)

└└└└└<AssignedPersonOrOrganization>

└└└└└└<PersonInOrganization>

└└└└└└└<AssociatedPerson>

└└└└└└└└<Person> (承認者)

Status には、"in progress", "approved", "approved with comments", "not approved"のいずれかが推奨されている。その他の値の場合は Class テンプレートを使用(RP4.6.16. Approval table18)

③ RP 解説

- ✓ (RP 5.1.2 PartView)では、「ステータス」は、<Part> <PartVersion> <ApprovalAssignment> で定義されている。
- ✓ <Approval>(4.6.16 項)には、<AssignedApproval> <Status>を取る。

6. 設計者

① 用語定義

- ✓ 設計作業に携わる人を定義する。設計者に限らず、下記の定義が必要と思われる。
 - 「設計者」：部品を設計した担当者。設計とは、その部品に期待される性能が確保できるように部品の形状、仕様等を決めること。必ずしも、CAD 機を操作しているとは限らない。
 - 「CAD モデラ」：CAD を操作し、具体的な形状をモデリングする。

- 「確認者」：設計者の上司。承認行為が3段階である場合、その2階層目に当たる。
- 「承認者」：設計部署として、そのCADデータの最終責任を取る人。

② XML 表記

```

<Part>
└<Versions>
└<PartVersion>
└< OrganizationOrPersonInOrganizationAssignment >
└<Role> (定義する人の役割。設計者等)
└<AssignedPersonOrOrganization>
└<PersonInOrganization>
└<AssociatedPerson>
└<Person>テンプレート
└<id>属性
└<FirstName>属性
└<LastName>属性
└<id>
└<PersonRole>
└<Class> (役割が言葉で記述される)
└<AssociateOrganization>
└<Organization>
└<id>
└<Name> (具体的な会社名)
└<OrganizationTypes> 'company' (どの会社に属しているか)
└<Organization> (複数の Organization も使用可)
└<id>
└<Name> (具体的な所属部署名)
└<OrganizationTypes> 'department' (どの部署か)

```

③ RP 解説

- ✓ <PartVersion>は、< OrganizationOrPersonInOrganizationAssignment >を定義(RP5.1.2 PartView)。< OrganizationOrPersonInOrganizationAssignment >配下の定義は、(RP4.6.18 PersonInOrganization Figure17)に記載されている。
- ✓ (RP4.6.18 PersonInOrganization Table20)には、PersonInOrganization が、<AssignedPersonOrOrganization>、<Role>を取ることが、記載されている。
- ✓ (RP4.6.18 PersonInOrganization Table21)には、具体的な Role 役割が記述されている(下記)。

Role	Explanation
'assignee'	The assigned person or organization to complete, review, comment or update the referenced object 参照オブジェクトを完成、レビュー、コメント、または更新するために割り当てられた個人または組織

'author'	The author holds the copyright 著作権は作者が保有
'classification officer'	The assigned person or organization is formally responsible for the classification of the referenced object 割り当てられた個人または組織は、参照されたオブジェクトの分類に対して正式に責任を負います
'creator'	The referenced object has been created by the assigned person or organization 参照されたオブジェクトは、割り当てられた個人または組織によって作成されました
'custodian'	The assigned person or organization is responsible for the existence and integrity of the referenced object 割り当てられた個人または組織は、参照されたオブジェクトの存在と完全性に責任があります
'customer'	The assigned person or organization acts as a purchaser or consumer of the referenced object 割り当てられた個人または組織は、参照されたオブジェクトの購入者または消費者として行動します
'design supplier'	The assigned person or organization is the one who delivers the data describing the referenced object 割り当てられた人または組織は、参照されたオブジェクトを説明するデータを提供する人です
'editor'	One or more attributes have been modified by the assigned person or organization 割り当てられた個人または組織によって 1 つ以上の属性が変更されました
'id owner'	The assigned person or organization is the one responsible for the designation of an identifier 割り当てられた個人または組織は、識別子の指定に責任を負います
'location'	The assigned organization is the place where the referenced object can be found or where it takes place 割り当てられた組織は、参照されたオブジェクトを見つけることができる場所、またはそれが行われる場所です
'locked by'	The assigned person that currently locks the associated object in the underlying legacy system 基礎となるレガシー システムで関連付けられたオブジェクトを現在ロックしている担当者
'manufacturer'	The assigned person or organization is the one who produces the actual (physical) object 割り当てられた人または組織は、実際の (物理的な) オブジェクトを作成する人です。
'owner'	The assigned person or organization owns the referenced object, and has final say over its disposition and any changes to it 割り当てられた個人または組織は、参照されたオブジェクトを所有し、その処分と変更について最終的な決定権を持ちます。
'read access'	The assigned person or organization neither has the right to modify any attributes of the referenced object, nor to modify, create or delete objects that are attached directly or indirectly to the referenced object 割り当てられた個人または組織には、参照されたオブジェクトの属性を変更する権

	利も、参照されたオブジェクトに直接的または間接的に添付されたオブジェクトを変更、作成、または削除する権利也没有せん。
'supplier'	The assigned person or organization is the one who delivers the actual (physical) object (e.g., a dealer) 割り当てられた人または組織は、実際の（物理的な）オブジェクトを提供する人です（例：ディーラー）
'wholesaler'	The assigned person or organization the one who is in the sales chain between the manufacturer and the supplier 割り当てられた人または組織 メーカーとサプライヤーの間の販売チェーンにいる人
'write access'	The assigned person or organization has the right to modify the attributes of the referenced object, as well as to modify, create, or delete objects that are attached directly or indirectly to the referenced object 割り当てられた個人または組織は、参照されたオブジェクトの属性を変更する権利、および参照されたオブジェクトに直接的または間接的に添付されたオブジェクトを変更、作成、または削除する権利を有します

- ✓ 更に、(RP4.6.18 Table22)に、<AssignedPersonOrOrganization>は、<AssociatedOrganization><Organization>と、<AssociatedPerson> <Person>を取ることが記載されている。<Organization>は、その役割の人の所属部署、<person>は、その役割の人の名前。
- ✓ <Person>は、<id>、<FirstName>、<LastName>を定義する。(RP4.6.17 Person)

今後の検討項目

Table21 には、<Role>が定義されているが、似たような Role があまりに多く、より詳細な説明、具体的な用例調査が必要。そのうえで、JAMA/JAPIA メンバ各社が、どの Role を使用するか、推奨案が必要と思われる。

7. 設計者部署

① 用語定義

「設計者部署」は、設計者の所属する部署。

② 今後の検討項目

- ✓ 今後の検討課題として、「組織変更時の対応」があるかもしれない。「部品ごとに、設計者・所属名が記載され、このタグは膨大な数があると思われる。一方で、組織変更は頻繁に実施される。
<AssignedPersonOrOrganization>の修正は工数が掛かるかもしれない。

③ XML 表記

「設計者」の項に記載

④ RP 解説

- ✓ 設計者部署は、<AssignedPersonOrOrganization>は、<AssociatedOrganization> <Organization>が定義されている(RP 4.6.8 Table22)。
- ✓ (RP4.6.2 Organization)で、<Organization>を定義。その配下には、<Id>、<Name>、

<OrganizationTypes>が定義される。

- ✓ 更に、(RP4.6.2 Organization)によると、<OrganizationTypes>配下には、<company>、<department>、<plant>が定義される。ひとつの<Organization>に対して、複数の<OrganizationTypes>が定義され、所属会社、所属部署が表現される。
- ✓ 設計者と設計者部署が、別の場所に記載される場合には、<OrganizationRelationShip>を使用する。

8. 設計者コンタクト先

① 用語定義

- ✓ 設計者に連絡する場合の電話番号。
- ✓ 設計者の電話番号ばかりではなく、メールアドレスなど、コンタクト先一般も、この項目の検討課題。

② XML 表記

不明。RP には、適切な表現が見当たらない。

③ この項目の考え方

- ✓ タスクチーム内議論の結果、設計者のコンタクト先情報は、CAD データとは別の情報であり、CAD データ属性情報の XML としては、定義しない。
- ✓ 部品の担当者だけ判ればよく、実業務上、担当者の連絡先は事前に把握できていると考えられるため。

④ RP 解説

- ✓ <Person>に紐づく情報のはずであるが、RP4.6.17 には、直接の意味付けがあるタグが見当たらない。強いて言えば、(RP4.6.17 person) Table19 での、<PropertyValueAssignment>または、<Description>で記述する方法が考えられる。

9. 出図日時

① 用語定義

- ✓ その部品が設計から評価部署・生技に公開された(出図)日時。

② XML 表記

<Part>テンプレート

└<Views>

└└<PartVersion>テンプレート

└└└<DatetimeAssignment>属性

└└└└<DateTime>テンプレート

└└└└└<AssignedDate>属性 (時間そのもの)

└└└└└└<Role>属性 (どんな時間かの説明)

(RP4.6.15 DateTime)に掲載されている用例

```
<DateTimeAssignment uid="dta--0000000017D374A0--id1">  
  <AssignedDate>2014-10-16T09:08:07</AssignedDate>
```

```

<Role>
  <ClassString>creation</ClassString>
</Role>
</DateTimeAssignment>

```

③ RP 解説

- ✓ RP の改訂が予定されている。
- ✓ (RP5.1.2 PartVersion) table20 に、<DatetimeAssignment>が定義され、<DateTime>を取る。
<DateTime>は、<AssignedDate>、<Role>をとる(RP4.6.15 DateTime)。
- ✓ <DateTimeAssignment>は、<PartVersion>の他にも、<PartView>にも定義されている(RP5.1.3 table26)。
<PartVersion>は out of scope ではないが、<PartView>では out of scope の扱い。
- ✓ <DateTime>、<Role>には、下記の値が定義されている(RP4.6.15 DateTime)。

Role	Explanation
'classification date'	the specified object is classified at the given date and time 指定されたオブジェクトは、指定された日時に分類されます
'creation'	the referenced object was created at the given date and time 参照されたオブジェクトは指定された日時に作成されました
'installation'	the referenced object was mounted in a product at the given date and time 参照されたオブジェクトは、指定された日時に製品にマウントされました
'lock'	the specified object is locked in the underlying legacy system since the given date and time 指定されたオブジェクトは、指定された日時以降、基になるレガシー システムでロックされています
'production'	the referenced object was produced at the given date and time 参照されたオブジェクトは、指定された日時に作成されました
'registration'	the referenced object was determined at the given date and time 参照されたオブジェクトは、指定された日時に決定されました
'update'	the referenced object was altered at the given date and time 参照されたオブジェクトは指定された日時に変更されました

今後の検討項目

下記の懸念のため、ユースケースでの検討が必要と思われる。

- ① <DateTime> <Role>に定義される上記表の、それぞれの意味、使用例等の調査が必要(上記表だけでは、実運用がイメージしにくい)。その上で、JAMA/JAPIA メンバ各社の実情に合っているか、検討が必要。
もし、合っていないのであれば、<Class>で定義検討や、RP 修正への要望の必要がある。
- ② 「出図」は設変が伴うため、DeltaChange タブでの、発行日書式も確認すべき。

10. 設変番号

① 用語定義

- ✓ (設計変更が実施される際には、設計から、設計変更を関係部署に通知する書類「設変書」が発行される。「設変番号」は、その設変書の発行番号。
- ✓ 1 回の設計変更で、複数の部品が、時には車両を跨って実施されることがある。この場合、複数部品が同じ設変書に記載されるため、結果的に、複数部品に同じ設変番号が付与されることになる。

② XML 表記

ISO10303-4442 4.2.6 DelataChange によると、下記となる。ただし、RP での詳細が未定であるため、懸念事項も追記した。

<Part>

└<Versions>

└└<PartVersion>

└└└<PartVersionRelationship> (・・・この位置で良いか確認要)

└└└└<Id> (・・・詳細な書式確認要)

└└└<PartView>

└└└└<ReplacedViewPartVersionRelationship> (・・・この位置で良いか確認要)

└└└└└<Id> (・・・詳細な書式確認要)

└└└<DatetimeAssignment>

└└└└<Date Time> (RP4.6.15)

└└└└└<AssignedDate>

└└└└└└<Role>

<DeltaChange>

└<DeltaChangeRelationshipSelect>

└└<Id> ・・・詳細な書式確認要

└<DeltaChangeActivity>

└└<ChangeElementSequence>

└└└<ModifyElement> ・・・変更点の Element 指示。具体的な指示方法、要確認

└└└<ModifySingleElement> ・・・変更点の Element 指示。ModifyElement との違い不明。
具体的な指示方法、要確認

└└└<DeleteElement> ・・・設変の結果無くなった Element 指示。具体的な指示方法、要確認

└└└<AddElement> ・・・設変で追加された Element 指示。具体的な指示方法、要確認

└└└<FrozenAssignment> ・・・どんなことがあっても変化しない要素。運用・記述書式、要確認

└└└<DatetimeAssignment>

└└└└<Date Time> (RP4.6.15)

└└└└└<AssignedDate>

└└└└└└<Role>

③ RP 解説

- ✓ RP の改訂が予定されている。
- ✓ 下記ポイントについて、ユースケース等の検討を、まず実施し、設変の XML 運用検討すべきと思われる。

- 各社の設変運用における設変書の管理・情報流通
- 設変に伴う CAD の運用方法。
- ✓ 設変時に変わりようがない要素は<FrozenAssignment>に記述される。実際の運用としては、ボルト・ナット等と考えられる。

今後の検討項目

- ① <PartView>の配下に<DeltaChange>が存在したり、<DeltaChange>配下に<PartView>があるのではなく、各々”独立”している。それでは、<DeltaChange>は、どこにあるのか？ <Part>は、CAD データと一緒にある(CAD データごとにある)。<DeltaChange>は設変書単位に存在しているのが自然。XML ファイルも別になると思われる。<ReplacedViewPartVersionRelationship>などでリンク関係が保持されると思うが、具体的には、ファイルが異なるので、意識してリンク関係が保持できる場所にファイルを持つ必要があるかと考える。具体的に、リンク関係を保持するイメージと、運用の要件の確認必要
- ② <DeleteElement>、<AddElement>で変更点が指示できる。CAD データで、Delete された要素と、Add された要素が、外部から XML 書式で紐づけ可能な状態になっていなくてはならない。具体的に、どのように実現可能か、下記のような検討が必要。
 - (ア) 各 CAD アプリケーションが、このような設変に対応しているか？ 現状の CAD 機能で充分か？
 - (イ) Delete と Add を記述することが、設計運用として、各社で、可能か検討が必要。今後、ISO で、削除・追加運用が標準化されるのであれば、各社は、設変前後のデータをキレイに記述しておく運用が必須となる。
- ③ <PartVersionRelationship>、<ReplacedViewPartVersionRelationship>、<DeltaChangeRelationshipSelect>等の、詳細な書式について、調査必要。<Id>は付与されると思うが、Relationship の詳細な書式は不明。
- ④ 設変日時は、<PartVersion>への付与もあり得るし、<DeltaChange>への付与もあり得る。複数の部品に跨った設変書があることを考えると、DeltaChange に日時を付与は一箇所の指示で済む。この運用検討が必要。

- ✓ ISO16949 (IATF 16949)の車両開発標準に、Change Management の記述がある。Change Order と Change Request との違い、工程により設計変更の違い有無等、工程毎に要件を分けて考える必要がある。

11. 設変日時

① 用語定義

- ✓ 設変書が発行された日時
- ✓ 各 CAD の出図日時と同義になるか、どうかは、運用要検討。つまり、設変が発行されるタイミングと、設計者が、その設変書に対応した CAD の出力タイミングは異なるかもしれないため、設変書の発行タイミングでの属性情報が必要となる事がありうる。

② XML 表記

設変の項にて記述。

③ RP 解説

- ✓ 設変書が発行されるタイミングと、その設変の CAD データが発行されるタイミング一致しているのであれば、設変書の発行日時は XML に記述する必要性は薄い。

12. 材質

① 用語定義

- ✓ その部品に用いられている材質。
- ✓ 基本的に、単品に付与される属性情報で、アセンブリ部品には付与されない。
- ✓ 単品部品でも、複数の材質が適宜される場合がありうる。この場合、どの材質が、部品のどこかを示せることが必要。

② XML 表記

```
<Part>
└<Versions>
  └<PartVersion>
    └<Views>
      └<PartView>
        └<MaterialIdentification>
          └<Description> (材質の記述)
            └<Id>
              └<ShapeElement> (材質の複数指示の場合)
                └<MaterialIdentification>
                  └<Description> (材質の記述)
                    └<Id>
                      └<Id>
                        └<ShapeDepenentProperty>
```

③ RP 解説

- ✓ 現在、RP の改訂が予定されている。
- ✓ (RP5.1.3 PartView) Table26 にて定義されている<PartView><MaterialIdentification>を記述。ただし、out of scope で詳細な運用方法は RP には記載されていない。
- ✓ (RP6.3.1 ShapeElement)にも、<MaterialIdetification>があるが、「部品の材質」という意味では、<PartView>直下の方が望ましいと思われる。
- ✓ ISO 10303-4442 4.2.13.2.4.3 MaterialIdetification では、下記の表記方法が示されている。
 <MaterialIdentificationAssginmentSelect>
 └<MaterialIdentification>
 └<MaterialPropertyAssignment>

今後の検討項目

- ① 材質の定義は、各社によって異なるため、<MaterialIdentificaton>で定義した材質は、定義した会社名を記載する必要があるが、<MaterialIdentificaton>が out of scope のため RP には詳細説明が無く、詳細は

不明である。

- ② 複数の材質を指示する場合には、<ShapeElement>を複数用いて(RP5.1.3 解説)、<MaterialIdentification>を指示する方法が考えられる(RP6.3.1 table36)。ただし、<ShapeElement>配下の<MaterialIdentification>も out of scope である。複数の<ShapeElement>を区別するために、<Id>と<ShapeDependentProperty>で、それぞれの<ShapeElement>の定義付けを行う必要があると思われる。この方法が正しいか、確認必要。
- また、その際に、CAD データと<ShapeElement>を紐づける方法の確認が必要。

13. 密度

① 用語定義

その部品に用いられている材質の密度。

② XML 表記

<MaterialIdentificationAssignmentSelect>

└<MaterialIdentification>

└└<MaterialPropertyAssignment>

<DataEnvironment>

└<MaterialPropertyValue>

└└<MaterialPropertyDefinition>

└└└<密度> : RP には、密度を定義するタグが見当たらない。(そのため、暫定的に日本語で表記)

③ RP 解説・ISO 書式検討途中資料概要

- ✓ RP6.3 で下記記述があるが、RP では、密度の記述は存在しない。
「material properties, such as density and weight, are currently not in scope of this document.
Hence, the terms “center of mass”, “center of geometry” and “centroid” are used synonymously.」
「密度や重量などの材料特性は、現在、このドキュメントの範囲外です。したがって、「質量の中心」、「ジオメトリの中心」、および「重心」という用語は同義語として使用されます」
- ✓ ISO 10303-4442 4.2.13.2.3 MaterialIdentification によると、<MaterialIdentification>、<MaterialPropertyAssignment>は<MaterialPropertyValue>を繋がる。

14. 板厚

① 用語定義

- ✓ その部品に用いられている板厚。
- ✓ シェル要素には必要な属性であるが、ソリッド要素に基本的には必要無い。ソリッド要素への板厚情報は、流通の当事者間でのニーズに依る。

② XML 表記

<Part>

└<Versions>

└└<PartVersion>

↳<Views>
 ↳<PartViews>
 ↳<ShapeElement>
 ↳<ThicknessSizeDimension>
 ↳<NumericalValue>
 ↳<Definition>
 ↳<PropertyDefinition>
 ↳<ShapeDependentProperty>
 ↳<ValueDetermination>
 ↳<ClassString> (板厚情報の由来。次項にて解説)

③ RP 解説

- ✓ ISO 10303-4442 4.2.13.2.8 に ThicknessSizeDimension が記載されている。RP では、まだ未検討な部分があると思われ、今後の改訂が待たれる。
- ✓ <GeneralShapeDependentProperty>、<ShapeElement>(RP6.3.1)に <ThicknessSizeDimension>がある(table36)。Out of scope の項目であり、RP には詳細な記述が無い。「体積」の項目と同様に、<PropertyValue> <NumericalValue>(RP4.6.9)を取ると思われる。<NumericalValue>から<Definition>を取る(RP4.6.9 table10)。<Definition>は<PropertyDefinition>を取る(RP4.6.9 Definition 解説)。しかし、<PropertyDefinition>の<Id>には、厚さの定義が見当たらない(RP12.2)。
- ✓ シェル部品の「板厚」は、「板厚方向」と関連付けの必要性や定義方法は、RP では不明。
- ✓ ISO 10303-4442 4.2.13.2.4.3 MaterialIdentification では、材質特性について、密度同様、特性が記述可能と思われる。
- ✓ ISO 10303-4442 の ShapeDependentProperty には、記載されたデータの由来を記入することが、下記のようにできる。

calculated	The value has been calculated.
designed	The value represents a value intended by the design.
estimated	The value has been estimated.
measured	The value has been measured.
required	The value represents a required.

15. 質量

① 用語定義

- ✓ その部品の質量

② XML 表記

(1)<AssignedPropertyValue>を使う方法

<Part>
 ↳<Versions>
 ↳<PartVersion>
 ↳「PropertyValueAssignment」属性
 ↳「PropertyAssignment」テンプレート

- └「AssignedPropertyValues」属性
 - └「NumericalValue」サブタイプ
 - └<Definition>
 - └<PropertyDefinition>
 - └<id>
 - └<mass property>
 - └<PropertyType>
 - └<DeterminationMethod>
 - └<Name>
 - └<Unit>
 - └<ValueComponent>

(2)<CenterOfMass>を使う方法 ～重心位置に質量を指示

- <Part>
 - └<Versions>
 - └<PartVersion>
 - └<CenterOfMass>
 - └<DefinedIn>
 - └<GeometricCoordinateSpace>
 - └<Unit>
 - └<Id>
 - └<Role>

③ RP 解説

- ✓ RP5.1.2 から、RP6.2 に定義
 - RP5.1.2 から、RP6.2 に定義されている<PropertyAssignment>は、<AssignPropertyValues> <NumericalValue>を取る。<NumericalValue>は、RP4.6.9 で定義される。Figure8 に記載されている通り、<Definition>で質量を定義していることを示す。<Definition>配下には、<PropertyDefinition>を定義する(RP12.2)。<PropertyDefinition>配下に<id>があり、この項が質量であることを宣言する。
 - <Name>で「質量」との言葉を定義、<Unit>で単位を定義する。質量の値そのものは、<ValueComponent>に定義される。
 - <PropertyType>は、この情報が、何処で定義されるのかを示す(RP12.2)。「system property」、「PDM property」、「customized PDM property」、「user defined attribute」、「property group」、が assembly validation property、geometric validation property が定義できる。assembly validation property では、組立部品にも対応しているようだ。
 - <DeterminationMethod>は、計算方法を定義している。「calculated」、「designed」、「estimated」、「measured」、「required」、「set point」が選択可能(RP3.6.9)。
- ✓ <CenterOfMass>を使用した方法
 - <PartView>、または<AssemblyDefinition>から、<CenterOfMass>を取る(RP13.1.2)。
<CenterOfMass>は<DeifinedIn>、<Id>、<Role>を取る(RP13.1.2 table68)。
 - ただし、PartVersion の項(RP5.1.2 table25)、PartView の項(RP5.1.3 table26)には、<CenterOfMass>は、見当たらない。

- ✓ Recommended Practices for Geometric and Assembly Validation Properties release 4.5 (2019/8/22 version)の 4.7 項に、Solid モデルに関しての Volume の記述あり。

16. 体積

① 用語定義

その部品の質量

② XML 表記

```
<Part>
  ↳<Versions>
    ↳<PartVersion>
      ↳<Views>
        ↳<PartViews>
          ↳<ShapeElement>
            ↳<ShapeDependentProperty>
              ↳<PropertyValue>
                ↳<NumericalValue>
                  ↳<Definition>
                    ↳<PropertyDefinition>
                      ↳<id>
                        ↳<Volume>
                          ↳<DeterminationMethod>
                            ↳<Name> 'volume'
                              ↳<Unit>
                                ↳<ValueComponent>
```

下記用例は、(RP6.3.1 GeneralShapeDependedtProperty)による。

```
<Part uid="p--0000000017086CB0">
  <Id>
    <Identifier uid="pid--0000000017086CB0--id1" id="as1"
      idRoleRef="rl--ii" idContextRef="o--000000178"/>
  </Id>
  <Name>
    <CharacterString>as1</CharacterString>
  </Name>
  <PartTypes>
    <ClassString>assembly</ClassString>
  </PartTypes>
  <Versions>
    <PartVersion uid="pv--0000000017086CB0--id1">
```

```

<Id id="/NULL"/>
<Views>
  <PartView xsi:type="n0:AssemblyDefinition"
uid="pvv--0000000017086CB0--id1">
    <InitialContext uidRef="ac--mechanicaldesign--design"/>
    <ShapeDependentProperty uid="sdp--000000782"
xsi:type="n1:General-ShapeDependentProperty">
      <PropertyType>
        <PropertyDefinition uidRef="pd--000000320"/>
      </PropertyType>
      <PropertyValue>
        <NumericalValue>
          <Definition>
            <PropertyDefinition uidRef="pd--000000320/>
          </Definition>
          <Name>
            <CharacterString>volume</CharacterString>
          </Name>
          <Unit uidRef="u--000000000"></Unit>
          <ValueComponent>120</ValueComponent>
        </NumericalValue>
      </PropertyValue>
    </ShapeDependentProperty>
    <ShapeElement uid="se--1">
      <Id id="/NULL"/>
      <ShapeDependentProperty uid="sdp--100000782"
xsi:type="n0:GeneralShapeDependentProperty">
        <PropertyType>
          <PropertyDefinition uidRef="pd--000000320"/>
        </PropertyType>
        <PropertyValue>
          <NumericalValue>
            <Definition>
              <PropertyDefinition uidRef="pd--000000320/>
            </Definition>
            <Name>
              <CharacterString>volume</CharacterString>
            </Name>
            <Unit uidRef="u--000000004"/>

```

```

        <ValueComponent>120</ValueComponent>
      </NumericalValue>
    </PropertyValue>
  </ShapeDependentProperty>
</ShapeElement>
</PartView>
</Views>
</PartVersion>
</Versions>
</Part>

```

③ RP 解説

- ✓ <ShapeDependentProperty> <PropertyValue> <NumericalValue> <Definition>で、体積の ID が定義され、<Name>で「体積」という言葉が定義される。<Unit>で単位が定義され、<ValueComponent>で、体積の数値そのものが定義される。
- ✓ <DeterminationMethod>も、体積同様に定義可能。
- ✓ <Name>に「volume」を定義している図が RP13.2.1.1 Figure72 に存在。

17. 寸法公差

① 用語定義

その部品を規定している一般寸法公差

② XML 表記

RP4.6.12 には、下記の用例がある。

```

<Unit uid="U__1">
  <Kind>
    <ClassString>SI system</ClassString>
  </Kind>
  <Name>
    <ClassString>metre</ClassString>
  </Name>
  <Prefix>
    <ClassString>milli</ClassString>
  </Prefix>
  <Quantity>
    <ClassString>length</ClassString>
  </Quantity>
</Unit>
<PropertyDefinition uid="ID_57">
  <Id id="general property"/>
  <PropertyType>
    <ClassString>customized PDM property</ClassString>
  </PropertyType>
</PropertyDefinition>
<PropertyValue uid="ID_159_3" xsi:type="n0:ValueWithTolerances">
  <Definition>
    <PropertyDefinition uidRef="ID_57"/>
  </Definition>

```

```

<Name>
  <CharacterString>PDMIFPvr_Tol</CharacterString>
</Name>
<Unit uidRef="U__1"/>
<LowerLimit>-3.000000000000000E-03</LowerLimit>
<TolerancedValue>8.901000000000000E+00</TolerancedValue>
<UpperLimit>4.000000000000000E-03</UpperLimit>
</PropertyValue>

```

③ RP 解説

- ✓ RP4.6.12 において、<ValueWithTolerance>が規定。
- ✓ 本用例そのもので、一般寸法公差が定義できないか、**要検討**。

18. 質量公差

① 用語定義

その部品を規定している一般質量公差

② XML 表記

前項の「寸法公差」で記述した RP4.6.12 の用例は、単位で長さを規定しているが、単位で質量を定義すれば、質量公差が定義出来るのではないか？ 下記用例は、質量の単位を定義した XML である。

```

<Unit uid="u--100000002">4
  <Kind><ClassString>SI system</ClassString></Kind>
  <Name><ClassString>gram</ClassString></Name>
  <Prefix><ClassString>kilo</ClassString></Prefix>
  <Quantity><ClassString>mass</ClassString></Quantity>
</Unit>

```

③ RP 解説

RP AnnexE に単位を規定があり、質量を定義できる。公差の定義は、RP4.6.12 の用例に依る。

19. モデルグレード(データランク)

① 用語定義

モデルグレード(データランク)は、CAD データ完成度を示す指標である。下記のように、幾つかの指標が存在する。

- ✓ JAMA/JAPIA 3DA モデルガイドライン、Annex B 「設計モデルグレード」
- ✓ JEITA 3DA モデル金型工程連携ガイドライン -「製品設計」と「金型設計・製作」間での 3DA モデルの有効な活用方法- プラスチック部品編 ver1.2 のおける「金型要件盛り込みランク」
- ✓ ASME Y14.47、5.2 項、5.3 項、5.4 項

これら 3 つの指標について記述する。

② JAMA/JAPIA 3DA モデルガイドライン、Annex B

設計モデルを用いて製品の設計及び製造の各業務を進めるためには、製品の形状が、ある程度の設計モデルグレードまで作り込まれている必要がある。以上から、図面タイプごとに必要な設計モデルグレードを規定した（表 Annex B-1 参照）。

分類	判定基準	設計モデルグレード		
		III	II	I
表現方法	ワイヤーフレーム	○		

	サーフェス	○	○	○
	ソリッド	○	○	○
作成範囲	基本形状のみ	○	○	○
	部品を取り付ける形状		○	○
	補強・質量軽減形状		○	○
	製造要件を反映した形状を含む、部品形状全体			○
抜き勾配	あり		△	○
フィレット面	あり		△	○
(参考) 用途	製品形状を定義する程度	レイアウト 形状	基本 製品形状	完成 製品形状

表 Annex B-1

③ JEITA 3DA モデル金型工程連携ガイドライン

JEITA の金型要件盛り込みランクの例。(JEITA 3DA モデル金型工程連携ガイドライン -「製品設計」と「金型設計・製作」間での 3DA モデルの)有効な活用方法- プラスチック部品編 ver1.2)、15 ページより。

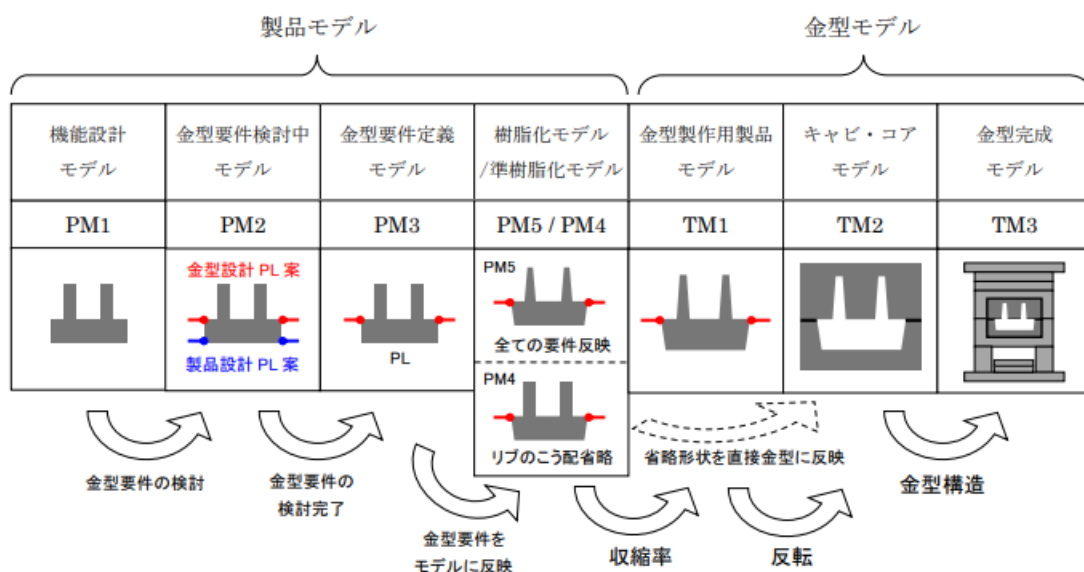


図 5-2 金型要件盛り込みランク

④ ASME Y14.47 Data set completeness

5.2 項 Maturity State、5.3 項 Geometry State、5.4 項 Annotation and Attribute State に記述があり、それぞれ、「設計の成熟度」、「形状の詳細度」、「注記と属性の詳細度」について記述されている。

5.2 Maturity State

The maturity state describes the intended use of a data set or the product life cycle. The maturity state consists of four completeness levels: M1, M2, M3, and M4.

M1 (conceptual): a data set that is used when there is a need to verify preliminary design and engineering and to confirm that the technology is feasible and that the design concept has the potential to be useful in meeting a specific requirement.

M2 (development): a data set that is used for testing or experimentation and for the analytical evaluation of the inherent ability of the design approach to attain the required performance.

M3 (production): a data set that is used for the commission, operation, service and decommission of a product.

M4 (Archive): a data set that is used to store the product information for preservation.

5.3 Geometry State

The geometry state describes the level of detail of the item in the model. The geometry state consists of three completeness levels: G1, G2, and G3.

G1 (none): the model that is created with no distinctions made for individual features.

G2 (partial): the model that has a portion of the features represented.

G3 (full): the model that has all features represented.

5.4 Annotation and Attribute State

The annotation and attribute state describes the completeness level of annotations and attributes in a data set. The annotation and attribute state consists of three completeness levels.

A1 (none): no annotations or attributes that are in the data set are intended to be used to define the product.

A2 (partial): the data set that contains a subset of all of the annotations or attributes intended to be used to define the product.

A3 (full): the data set that contains all of the annotations or attributes [including items such as general notes and other product and manufacturing information (PMI)] without querying the model geometry intended to be used to define the product.

5.2 成熟度

成熟度は、データ セットまたは製品ライフ サイクルの使用目的を表します。成熟度は、M1、M2、M3、および M4 の 4 つの完全性レベルで構成されます。

M1 (conceptual): 予備設計とエンジニアリングを検証し、技術が実現可能であり、設計コンセプトが特定の要件を満たすのに役立つ可能性があることを確認する必要がある場合に使用されるデータセット。

M2 (開発): テストまたは実験、および必要なパフォーマンスを達成するための設計アプローチの固有の能力の分析評価に使用されるデータセット。

M3 (生産): 製品の試運転、運用、サービス、廃止に使用されるデータセット。

M4 (アーカイブ): 製品情報を保存するために使用されるデータセット。

5.3 ジオメトリの状態

ジオメトリの状態は、モデル内のアイテムの詳細レベルを表します。ジオメトリ状態は、G1、G2、および G3 の 3 つの完全性レベルで構成されます。

G1 (なし): 個々の機能を区別せずに作成されたモデル。

G2 (部分): 機能の一部が表現されているモデル。

G3 (フル) : 全ての機能をそのまま表現したモデル。

5.4 注釈と属性の状態

注釈と属性の状態は、データ セット内の注釈と属性の完全性レベルを表します。注釈と属性の状態は、3 つの完全性レベルで構成されます。

A1 (なし): データ セットにある注釈または属性は、製品を定義するために使用されることを意図していません。

A2 (部分的): 製品を定義することを意図したすべての注釈または属性のサブセットを含むデータ セット。

A3 (フル): 製品を定義するために使用することを意図したモデル ジオメトリを照会せずに、すべての注釈または属性 [一般的な注意事項やその他の製品および製造情報 (PMI) などの項目を含む] を含むデータ セット。

⑤ XML 表記

3 つのモデルグレード(データランク)とも、XML で表現する方法を、RP で見つけることができなかった。

とはいえ、そもそも、XML にモデルグレードを記述する必要性の検討からすべきである。。

20. 図表

① 用語定義

その部品の関しての情報で、3D 図面、または 2D 図面で、図表の形式で表示されている情報。

図表で表示されていることに本項では注目し、図表について一般的な検討を行う。情報の中身によっては、特に異なる運用をする情報があり得る。

② XML 表記

RP4.6.13 に<ValueList>、RP4.6.14 に<ValueSet>が定義されている。RP4.6.13<ValueList>の用例を下記に示す。

```
<PropertyDefinition uid="ID_57">
  <Id id="general property"/>
  <PropertyType>
    <ClassString>customized PDM property</ClassString>
  </PropertyType>
</PropertyDefinition>
<PropertyValue uid="ID_170_3" xsi:type="n0:ValueList">
  <Definition>
    <PropertyDefinition uidRef="ID_57"/>
  </Definition>
  <Name>
    <CharacterString>PDMIFPvr_ValueList</CharacterString>
  </Name>
  <Values>
    <PropertyValue xsi:type="n0:StringValue" uid="ID_170_3_1">
      <Definition>
        <PropertyDefinition uidRef="ID_57"/>
      </Definition>
      <ValueComponent>
        <CharacterString>PDMIFPvr_ValueList#1</CharacterString>
      </ValueComponent>
    </PropertyValue>
    <PropertyValue xsi:type="n0:StringValue" uid="ID_170_3_2">
      <Definition>
        <PropertyDefinition uidRef="ID_57"/>
      </Definition>
      <ValueComponent>
        <CharacterString>PDMIFPvr_ValueList#2</CharacterString>
      </ValueComponent>
    </PropertyValue>
  </Values>
</PropertyValue>
```

```

    </ValueComponent>
  </PropertyValue>
  <PropertyValue xsi:type="n0:StringValue" uid="ID_170_3_3">
    <Definition>
      <PropertyDefinition uidRef="ID_57"/>
    </Definition>
    <ValueComponent>
      <CharacterString>PDMIFPvr_ValueList#3</CharacterString>
    </ValueComponent>
  </PropertyValue>
</Values>
</PropertyValue>

```

③ RP 解説

- ✓ RP4.6.13 に<ValueList>、RP4.6.14 に<ValueSet>の用例は、ValueList と ValueSet の言葉が異なっているだけで、他は同じ。
- ✓ ValueList と ValueSet の違いは、ValueList が順番があり(ordered)、ValueSet は順番が無い(unordered)であること。

21. 組立/単品

① 用語定義

その部品が、組立部品か、単品部品かの表示。

会社によって、組立部品か、単品部品かの定義は異なると思われる。組立部品であるが、部品表上、単品部品扱いの運用もあり得る。なので、必ずしも CAD ファイルの形態に依る訳では無い。例えば、一つの塊として、CATIA の CATProduct ファイルと CATPart ファイルが登録されていれば、その部品は単品扱いになる。

例

OEM A 社

└ドア

└ドアインナパネル …単品

└ドアアウトパネル …単品

└**ドアロック** …単品

└

仕入先 B 社

└**ドアロック** …組立部品

└ドアロックアクチュエータ …組立部品

└ドアロックアクチュエータモータ …単品

└ドアロックアクチュエータギア …単品

└

└ドアロックリンク …単品

└ドアロックケース ……単品

└ ……

この例では、OEM A 社では、ドアロックは単品部品扱い。しかし、仕入先 B 社では、同じドアロックが組立部品扱いとなる。

② XML 表記

<Part> 組立部品側

└<Id>

└<Versions>

└└<PartVersion>

└└└<View>

└└└└<PartView>

└└└└└<ViewOccurrenceRelationship> ここで、NextAssemblyOccurrenceUsage を記述

└└└└└└<Related>

└└└└└└└<RelationType> ここで、NextAssemblyOccurrence を記述。組立部品である表示がされる

<Part> 単品部品側

└<Id>

└<Versions>

└└<PartVersion>

└└└<View>

└└└└<PartView>

└└└└└<Occurrence> ここで、SingleOccurrence が記述され、「子」側の部品であることが示される

└└└└└└<Id>

上記により、組立部品であることが表示される。会社によって異なる場合には、複数の<Part><Id>を

<idContextRef>で区別する。

RP による実例は下記。(RP 7.1 項)

組立部品側

```
<Part uid="p--00000000017086CB0">
```

```
<Id>
```

```
<Identifier uid="pid--00000000017086CB0--id1" id="as1" idRoleRef="rl--ii"
idContextRef="o--0000000178"/>
```

```
</Id>
```

```
...
```

```
<Versions>
```

```
<PartVersion uid="pv--00000000017086CB0--id1">
```

```
...
```

```
<Views>
```

```
<PartView xsi:type="n0:AssemblyDefinition"
```

```
uid="pvv--00000000017086CB0--id1">
```

```
...
```

```

    <ViewOccurrenceRelationship uid="pvvid--000000001E5A89F0--10"
    xsi:type="n0:NextAssemblyOccurrenceUsage">
        <Related uidRef="pi--000000001E5A89F0--10"/>
        <RelationType>
            <ClassString>next assembly occurrence</ClassString>
        </RelationType>
    ...
</ViewOccurrenceRelationship>
...
</PartView>
</Views>
</PartVersion>
</Versions>
</Part>

```

単品部品側

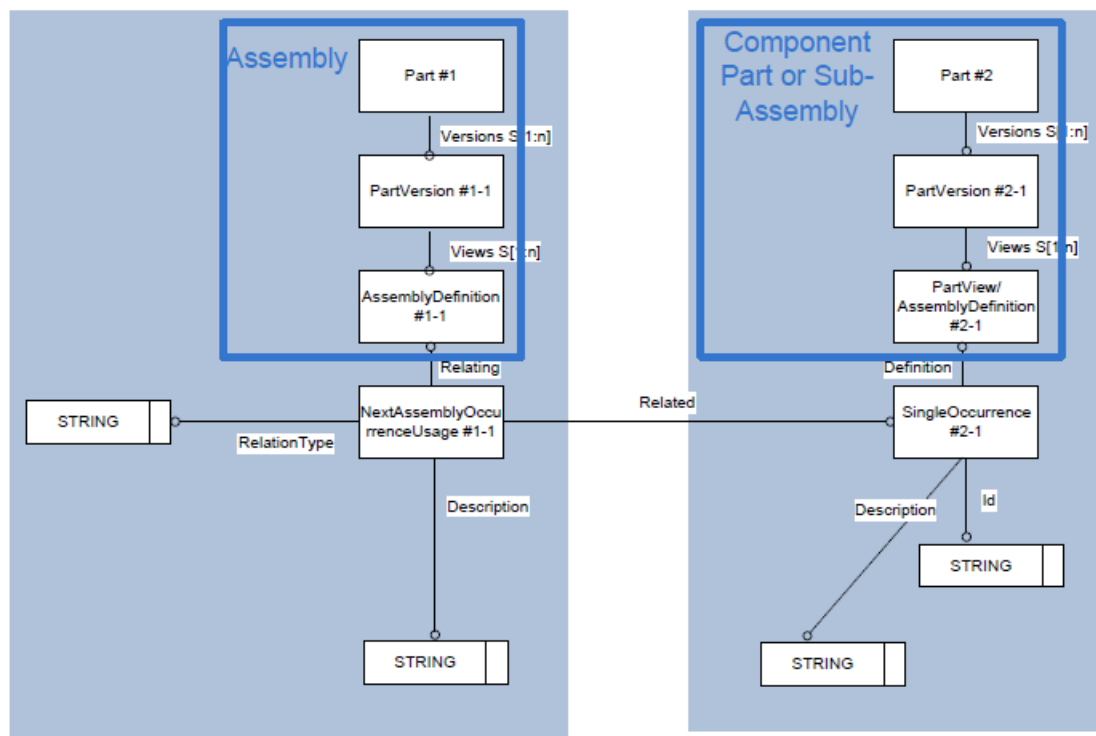
```

<Part uid="p--000000001E5A89F0">
    <Id>
        <Identifier uid="pid--000000001E5A89F0--id2" id="plate" idRoleRef="rl--ii"
        idContextRef="o--000000178"/>
    </Id>
    ...
    <Versions>
        <PartVersion uid="pv--000000001E5A89F0--id2">
            ...
            <Views>
                <PartView uid="pvv--000000001E5A89F0--id2">
                    ...
                    <Occurrence xsi:type="n0:SingleOccurrence"
                    uid="pi--000000001E5A89F0--10">
                        <Id id="plate.1"/>
                    </Occurrence>
                    ...
                </PartView>
            </Views>
        </PartVersion>
    </Versions>
</Part>

```

③ RP 解説

RP Figure25 に示されているように、組立部品と、その配下のサブアセンブリの組立部品または単品部品は、NextAssemblyOccuenceUsage と SingleOccurrence 間で、Related されている。この NextAssemblyOccuenceUsage の存在有無で、組立部品であることを示すことができる。ただし、SingleOccurrence でも、サブアセンブリ部品である可能性があるため、SingleOccurrence だけでは単品部品とは言えない。



RP Figure25: Template SingleOccurrence

22. CAD データの座標

① 用語定義

会社によって、座標軸の方向が異なり、データ流通時に、XML 表記により、座標軸方向が明示され、必要に応じて変換できれば好都合な場合がある。

座標軸の方向は JAMA/JAPIA にて調査されている。調査結果を下記に示す。

CAD データの座標軸の方向の調査結果

会社名	車両前後	車両左右	車両上下
スズキ	Y(+が後)	X(+が左)	Z(+が上)
SUBARU	X(+が後)	Y(+が右)	Z(+が上)
ダイハツ	X(+が後)	Y(+が右)	Z(+が上)
トヨタ	X(+が後)	Y(+が右)	Z(+が上)
日産	X(+が後)	Y(+が右)	Z(+が上)
ホンダ	X(+が後)	Y(+が右)	Z(+が上)

マツダ	X(+が後)	Y(+が右)	Z(+が上)
三菱自動車	X(+が後)	Y(+が右)	Z(+が上)
ふそう	X(+が後)	Y(+が右)	Z(+が上)
ヤマハ	X(+が後)	Z(+が左)	Y(+が上)

② XML 表記

RP7.3.1 項による部品の移動の用例を示す。この例を応用することで、座標軸の変換が可能であると思われる。

<Part>

└<Id>

└<Versions>

└└<PartVersion>

└└└<View>

└└└└<PartView>

└└└└└<ViewOccurrenceRelationship>

└└└└└└<Related>

└└└└└└└<RelationType>

└└└└└└└└<Placement>

└└└└└└└└└<CartesianTransformation>

└└└└└└└└└└<RotationMatrix>

└└└└└└└└└└└<TranslationVector>

③ RP 解説

✓ 座標軸の変換

- 上記用例(RP7.3.1)は、部品ごとに座標系が異なる場合であるが、個々の部品ごとに座標系の方向が異なる場合は、ごく少数だと思われる。会社間のやり取りで、座標軸が異なる場合には、XML のヘッダなどに、座標軸方向が記述されることが望ましい。
- 部品によって、車両番線上に配置される場合と、車両位置との関係が薄い部品が存在する。そのため、この座標軸の表示は、車両番線上に配置される部品でないと表示される意味が無い。表示させる部品は、データ授受する両方で事前調整されるべきである。
- また、本機能を使用すれば、原点配置の部品に対し、車両番線位置を指示して車両番線上の配置することも可能となる。その場合は、関係する会社間での運用調整が必要となると思われる。

✓ バリエーションの部品の配置情報

- RP7.3.1 は、もともとの目的は、部品移動の XML 書式の記述である。組立図で、流用部品の座標が異なる場合の仕様を意図している。例えば、「ホールベース違いで、派生車が元車の〇〇mm 後方に配置される」といったような場合である。
- 車両天井のアシストグリップのような、単純な XYZ 移動で表現できない部品の位置移動に関しては、RP7.3.2 項に記述されている。

以上