

### 8.2.6.3 用語



すべてのICHコントロールド・ボキャブラリはgenericcodeファイルおよびスプレッドシートファイルに記載される。<sup>15</sup>



コードには、規制当局によってさらに制約が課せられる場合がある。該当する『地域ごとの実装ガイド』を参照すること。

### 8.2.6.4 除外要素

*contextOfUse* 要素で除外されるクラス要素はない。

### 8.2.7 Related Context of Use (Context of Use のライフサイクル)

*relatedContextOfUse* 要素を使用して、送信者は *contextOfUse* 要素を 1 つまたは複数の *relatedContextOfUse* 要素に関連付けることができる。*replacementOf* の関係性を使用して、Context of Use 要素のライフサイクルが追跡される。

#### 8.2.7.1 XML 上の配置

XML メッセージの *relatedContextOfUse* 要素は次のように配置する：

- `controlActProcess>>subject>>submissionUnit>>component>>priorityNumber>>contextOfUse>>replacementOf>>relatedContextOfUse`

XML の記述方法については、表 7:v4.0 XML メッセージ構造を参照のこと。

#### 8.2.7.2 XML 要素

次の表に、*relatedContextOfUse* 要素に必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



*classCode* と *moodCode* は eCTD v4.0 XML メッセージでは不要である。*classCode* は「DOC」、*moodCode* は「EVN」に固定されている。これらの属性に別の値が含まれる場合、その XML メッセージはスキーマに対して妥当でない。

*relatedContextOfUse* 要素に適用される条件は以下の通り：

- 新しい *contextOfUse* によって置換される時、1 つまたは複数の *relatedContextOfUse* 要素を XML で提出できる。

#### 8.2.7.2.1 *relatedContextOfUse.id*

要素	属性	多重度	値の型例	説明指示
<i>id</i>		[1..1]		本要素は <i>relatedContextOfUse</i> のコンテナ要素であり、識別子によって参照される。
	<i>root</i>	[1..1]	妥当な UUID	本属性は <i>id</i> 要素の <i>root</i> 属性であり、置換される <i>relatedContextOfUse</i> 要素のグローバル一意識別子を提供する。
<b>準拠事項</b>		<i>id@root</i> は必須の属性である。		

<sup>15</sup> 最終的な実装用語集は ESTRI のウェブサイトに掲載される。

要素	属性	多重度	値の型 例	説明 指示
運用規則				1つの <i>contextOfUse</i> 要素に1つまたは複数の <i>relatedContextOfUse</i> 要素を含めることができる。
除外要素/属性				eCTD v4.0 で不要な属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>id@extension</i></li> <li>• <i>id@identifierName</i></li> <li>• <i>id@scope</i></li> <li>• <i>id@reliability</i></li> <li>• <i>id@displayable</i></li> <li>• <i>id@validTimeLow</i></li> <li>• <i>id@validTimeHigh</i></li> <li>• <i>id@controlInformationRoot</i></li> <li>• <i>id@controlInformationExtension</i></li> <li>• <i>id@nullFlavor</i></li> <li>• <i>id@flavorId</i></li> <li>• <i>id@updateMode</i></li> </ul>

### 8.2.7.3 用語



すべての ICH コントロールド・ボキャブラリは *genericcode* ファイルおよびスプレッドシートファイルに記載される。<sup>16</sup>

### 8.2.7.4 除外要素

*relatedContextOfUse* 要素で除外されるクラス要素はない。

## 8.2.8 Document Reference

Document は複数の Context of Use から参照されることができ、*documentReference* 要素を使用してそれぞれの *contextOfUse* に対応する Document を指定する。同じ Submission Unit の中で Document が使用される際に、Document の使用先 (CTD 見出し) がそれぞれ異なってもよい。これに従い、新たな Context of Use (すなわち、ステータス・コードが active なもの) はそれぞれ1つの Document への参照を示していなければならない。

### 8.2.8.1 XML 上の配置

XML メッセージの *documentReference* 要素は次のように配置する：

- *controlActProcess*>>*subject*>>*submissionUnit*>>*component*>>*priorityNumber*>>*contextOfUse*>>*derivedFrom*>>*documentReference*

*derivedFrom* 要素の前に1つまたは複数の *replacementOf* 要素を記述することができる。

XML の記述方法については、表 7: v4.0 XML メッセージ構造を参照のこと。

### 8.2.8.2 XML 要素

次の表に、*documentReference* 要素に必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



*classCode* と *moodCode* は eCTD v4.0 XML メッセージでは不要である。*classCode* は「DOC」、*moodCode* は「EVN」に固定されている。これらの要素に別の値が含ま

<sup>16</sup> 最終的な実装用語集は ESTRI のウェブサイトに掲載される。

まれる場合、そのXMLメッセージはスキーマに対して妥当でない。

*documentReference* 要素に適用される条件は以下の通り：

- 各 *contextOfUse* につき 0~1 つの *documentReference* 要素を送信できる。注：Context of Use 要素を削除する場合、Document への参照は不要である。
- *contextOfUse.statusCode@code*="active" のとき、*documentReference* 要素は必須である。
- *contextOfUse.statusCode@code*="suspended" のとき、*documentReference* 要素を指定してはならない。

#### 8.2.8.2.1 *documentReference.id*

要素	属性	多重度	値の型例	説明指示
<i>id</i>		[1..1]		本要素は Document への参照情報を格納するコンテナ要素である。
	<i>root</i>	[1..1]	妥当な UUID	<i>root</i> 属性または <i>id</i> 要素は、参照先 <i>document</i> 要素のグローバル意識別子を提供する。
準拠事項 運用規則	<i>documentReference</i> 要素が指定されている場合、 <i>id@root</i> 属性は必須である。			
	<i>id@root</i> は Submission Unit で送信される Document、または以前の Submission Unit で送信された Document への参照である。 承認申請に関する文書保管の詳細については、該当する『地域ごとの実装ガイド』を参照のこと。			
除外要素/属性	eCTD v4.0 で不要な属性は以下の通り：			
	<ul style="list-style-type: none"> <li>• <i>id@extension</i></li> <li>• <i>id@identifierName</i></li> <li>• <i>id@scope</i></li> <li>• <i>id@reliability</i></li> <li>• <i>id@displayable</i></li> <li>• <i>id@validTimeLow</i></li> <li>• <i>id@validTimeHigh</i></li> <li>• <i>id@controlInformationRoot</i></li> <li>• <i>id@controlInformationExtension</i></li> <li>• <i>id@nullFlavor</i></li> <li>• <i>id@flavorId</i></li> <li>• <i>id@updateMode</i></li> </ul>			

#### 8.2.8.3 用語



この要素に対応するコントロールド・ボキャブラリはない。

#### 8.2.8.4 除外要素

*documentReference* 要素で除外されるクラス要素はない。

#### 8.2.9 Context of Use の Keyword

*keyword* 要素は、*contextOfUse* に関する追加情報を送信するために使用される。*keyword* は外部のコントロールド・ボキャブラリで定義されるか、メッセージの中で *keywordDefinition* として定義される。

### 8.2.9.1 XML上の配置

Context ofUse の場合、XMLメッセージの keyword 要素は次のように配置する：

- `controlActProcess>>subject>>submissionUnit>>component>>priorityNumber>contextOfUse>>referencedBy>>keyword`

`referencedBy` 要素の前に、`primaryInformationRecipient`、`replacementOf`、`derivedFrom`、または `subjectOf` 要素を記述することができる。

XMLの記述方法については、表 7:v4.0 XMLメッセージ構造を参照のこと。

### 8.2.9.2 XML要素

次の表に、keyword 要素に必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



`classCode` と `moodCode` は eCTD v4.0 XML メッセージでは不要である。`classCode` は「ACT」、`moodCode` は「EVN」に固定されている。これらの属性に別の値が含まれる場合、そのXMLメッセージはスキーマに対して妥当でない。



eCTD v4.0 XMLメッセージには `typeCode` が必要である。`typeCode` には「REFR」を指定すること。XMLメッセージに、この属性に別の値が含まれる場合、そのメッセージはスキーマに対して妥当でない。

keyword 要素に適用される条件は以下の通り：

- 各 `contextOfUse` 要素につき 0 から複数の `keyword` 要素を送信できる。
- `contextOfUse` 要素で使用すべき個々の Keyword の種類については、『地域ごとの実装ガイド』を参照のこと。

#### 8.2.9.2.1 keyword.code

要素	属性	多重度	値の型例	説明指示
<code>code</code>		[1..1]		本要素は Keyword を識別するコードのコンテナ要素である。
	<code>code</code>	[1..1]	テキスト  <i>ich_route_1, MANU001 または MFR_001 など (製造業者サイトの場合)</i>	本属性は Keyword のコード値を示す <code>code</code> 属性である。
	<code>codeSystem</code>	[1..1]	テキスト  <i>例：OID 値または送信者が定義したテキスト</i>	<code>codeSystem</code> 属性はコントロール・ボキャブラリ・システムを一意に識別する識別子を指定する。
<b>準拠事項</b>	<code>code</code> と <code>codeSystem</code> 属性は必須である。  Keyword に指定できるコードは1つのみである。			

要素	属性	多重度	値の型 例	説明 指示
運用規則				ICH 指定のコントロールド・ボキャブラリ記載の Keyword を提供する場合、 <i>codeSystem</i> 属性値は OID でなければならない。送信者が定義する keyword は送信者によって割り当てられた OID またはテキストを提供しなければならない。 <i>code</i> の表示名は、対応するコード・システムから取得される。
除外要素/属性				eCTD v4.0 で不要な要素と属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>code.displayName</i></li> <li>• <i>code.originalText</i></li> <li>• <i>code.translation</i></li> <li>• <i>code.source</i></li> <li>• <i>code@codeSystemName</i></li> <li>• <i>code@codeSystemVersion</i></li> <li>• <i>code@valueSet</i></li> <li>• <i>code@valueSetVersion</i></li> <li>• <i>code@codingRationale</i></li> <li>• <i>code@validTimeLow</i></li> <li>• <i>code@validTimeHigh</i></li> <li>• <i>code@controlInformationRoot</i></li> <li>• <i>code@controlInformationExtension</i></li> <li>• <i>code@nullFlavor</i></li> <li>• <i>code@flavorId</i></li> <li>• <i>code@updateMode</i></li> </ul>

### 8.2.9.3 用語



すべての ICH コントロールド・ボキャブラリは genericcode ファイルおよびスプレッドシートファイルに記載される。<sup>17</sup>

### 8.2.9.4 除外要素

*keyword* 要素で除外されるクラス要素はない。

### 8.2.10 XML サンプル : Context of Use

#### 8.2.10.1 Context of Use 要素 | Context of Use の Keyword

次に Context of Use の XML の例を示す。*contextOfUse* は *submissionUnit* 要素の *component* として記述される。各 Component には Priority Number 要素が含まれていること。

```
<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="1f080afd-f5d4-4cec-8d09-2bf0ea6bec66"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    ...
    [追加の要素または属性を記述する場合は
     contextOfUse.primaryInformationRecipient の後に記述される。]
    ...
  <replacementOf typeCode="RPLC">
```

<sup>17</sup> 最終的な実装用語集は ESTRI のウェブサイトに掲載される。

```

    <relatedContextOfUse>
      <id root="25fdfdbcb-a2a2-4f2b-a2aa-9ccb4c096acb"/>
    </relatedContextOfUse>
  </replacementOf >
  <derivedFrom>
    <documentReference>
      <id root="8dc27e78-41ef-4b8d-960d-2626b743f194"/>
    </documentReference>
  </derivedFrom>
  ...
  [追加の要素または属性を記述する場合は subjectOf.submissionReference の後に記述される。本要素については『地域ごとの実装ガイド』を参照すること。]
  ...
  <referencedBy typeCode="REFR">
    <keyword>
      <code code="MANU001" codeSystem="2.16.840.1.113883.X.Y.Z"/>
    </keyword>
  </referencedBy>
  <referencedBy typeCode="REFR">
    <keyword>
      <code code="SUB001" codeSystem="2.16.840.1.113883.X.Y.Z"/>
    </keyword>
  </referencedBy>
</contextOfUse>
</component>

```



Context of Use の keyword はすべて例示の目的でのみ示す。使用可能な組み合わせについては、コントロールド・ボキャブラリおよびICHM4「医薬品の承認申請のためのコモン・テクニカル・ドキュメント (CTD) の構成」ファイルを参照のこと。



色の使用方法については XML での色の使用例 を参照のこと。

複数の Context of Use 要素が同じ *contextOfUse.code@code* と *keyword@code* の組み合わせを持つ場合、Priority Number を使用して Context of Use 要素を順序付けし、表示する順序を指定することができる。次の XML サンプルに、Context of Use に対する Priority Number と Keyword の使用例を示す。

```

<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="27c069e1-8fec-4b07-907e-cf691543cf66"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <!-- "Controls for Material YYY" というタイトルの文書 -->
      <documentReference>
        <id root="26a7e20a-b7b6-4729-adcf-75fb90097d68"/>
      </documentReference>
    </derivedFrom>
    <referencedBy typeCode="REFR">
      <keyword>
        <code code="MANU001" codeSystem="2.16.840.1.113883.X.Y.Z"/>
      </keyword>
    </referencedBy>
  </contextOfUse>
</component>

```

```

    </keyword>
  </referencedBy>
  <referencedBy typeCode="REFR">
    <keyword>
      <code code="SUB001" codeSystem="2.16.840.1.113883.X.Y.Z"/>
    </keyword>
  </referencedBy>
</contextOfUse>
</component>
<component>
  <priorityNumber value="2000"/>
  <contextOfUse>
    <id root="749e6f91-797b-4aeb-89c6-7cf7b9402c15"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <!--"Analytical Method #234"というタイトルの文書-->
      <documentReference>
        <id root="57e00a6f-5425-4c0e-98ad-ca4b2e0befea"/>
      </documentReference>
    </derivedFrom>
    <referencedBy typeCode="REFR">
      <keyword>
        <code code="MANU001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>
      </keyword>
    </referencedBy>
    <referencedBy typeCode="REFR">
      <keyword>
        <code code="SUB001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>
      </keyword>
    </referencedBy>
  </contextOfUse>
</component>

```



Context of Use の keyword はすべて例示の目的でのみ示す。使用可能な組み合わせについては、コントロールド・ボキャブラリおよびICHM4「医薬品の承認申請のためのコモン・テクニカル・ドキュメント (CTD) の構成」ファイルを参照のこと。



色の使用方法についてはXMLでの色の使用例を参照のこと。

### 8.2.10.2 Context of Use 要素の管理

本セクションでは、contextOfUse のライフサイクル管理について説明する。contextOfUse が id と共に提出されると、その contextOfUse のライフサイクルが始まる。ICH において調和されている規則は以下の通り：

- Context of Use を置換する場合、置換する 2 つのインスタンスには同じ Context of Use コードと Keyword コードの組み合わせが使用されなければならない（つまり、置換後に目次内での提出物の表示位置が変わらないこと）。
- Context of Use を置換すると、過去に提出された contextOfUse 要素（すなわち relatedContextOfUse 要素）は無効となる。

contextOfUse が変更される理由は以下の通り：

- Context of Use の置換：異なる Document または documentReference 要素中で過去に参照した Document を参照する新たな contextOfUse を提出する。

- **Context of Use の削除** : Submission のライフサイクル中に Context of Use を削除する必要が生じた場合、Submission Unit で *statusCode* 要素を変更することにより、Context of Use の削除を通知する。
- **新しい Keyword** : Context of Use の Keyword を変更する必要がある場合（すなわち、Context of Use コードと Keyword コードの組み合わせの変更）、現在の Context of Use を削除して新しい Context of Use を提出する。送信者が Keyword Definition の Display Name のみを更新する場合、コード値は同一のままであり、新しい Keyword とはみなされない。

#### 8.2.10.2.1 新しい Context of Use 要素の挿入

*submissionUnit* に同じ *contextOfUse* コードと *Keyword* コードを持つ複数の Context of Use が含まれる場合、*component* 要素に Priority Number を設定し、関連する *contextOfUse* 要素間における相対的な表示位置を指定する。

```
<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="fd28ce84-651a-437f-b7f0-5171ad21057d"/>
    <code code="ich_3.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <!-- 参考文献#1-->
      <documentReference>
        <id root="0ac0295e-766f-4567-9d63-40b8180de0c0"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>
<component>
  <priorityNumber value="2000"/>
  <contextOfUse>
    <id root="d27a4269-eebc-449f-9f33-645907f964984"/>
    <code code="ich_3.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <!-- 参考文献#2-->
      <documentReference>
        <id root="839235d5-1409-46c6-a144-e4fc3988e313"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>
```

改訂時に提出する Submission Unit では、前回のシーケンスと同じ *contextOfUse.code* を使用して Context of Use を追加しなければならない場合がある。

同じ *contextOfUse.code@code* と *keyword.code@code* の組み合わせを持つ新しい Context of Use を追加する例を、前掲の例を用いて以下に示す。時間の経過に伴いコンテンツは同じ見出しに対して追加されるので、Priority Number が既存のコンテンツ内での配置を示す。この Context of Use は、以前に記述された 2 つの Context of Use 要素の間に挿入される。

#### Context of Use の挿入

```
<component>
```



```

<priorityNumber value="1500"/>
<contextOfUse>
  <id root="d5528cfc-15f8-479e-ab59-562c0aa3a5d8"/>
  <code code="ich_3.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
  <statusCode code="active"/>
  <derivedFrom>
    <!-- 参考文献#3-->
    <documentReference>
      <id root="1982f2bf-bd82-45c6-83d7-8838598c971f"/>
    </documentReference>
  </derivedFrom>
</contextOfUse>
</component>

```



色の使用方法については XML での色の使用例を参照のこと。

#### 8.2.10.2.2 Context of Use の並べ替え

*contextOfUse* 要素が間違った表示順序で送信され、送信者がその順序を修正したい場合がある。

並べ替えは、既存の Context of Use の間に新しい Context of Use を挿入する際にも必要になることがある。*contextOfUse* 要素の並べ替えが必要な場合は、以下の基本規則に従うこと：

- 並べ替えを行うときに新しい Context Of Use を追加する場合、当該 *contextOfUse* 要素に *contextOfUse.priorityNumber@updateMode* 属性を使用しない。
- Context of Use を並び替える必要がある場合、更新範囲が順序に対するもののみであることを示すため、当該 *contextOfUse* 要素に *contextOfUse.priorityNumber@updateMode* を用いる。

過去に提出した Context of Use の順序が間違っていた際に順序を更新する例を以下に示す。Context of Use の並べ替えには *contextOfUse.code*、*documentReference.id* および *keyword.code* の要素は含まれないことに注意すること。コンテンツは正しい順序で配置される（例：過去に Priority Number 5000 として提出した場合）。注：申請者は *contextOfUse* 要素の並べ替えのみを目的として Submission Unit を送信しないことが推奨される。

```

<component>
  <priorityNumber value="900" updateMode="R"/>
  <contextOfUse>
    <id root="d5528cfc-15f8-479e-ab59-562c0aa3a5d8"/>
    <statusCode code="active"/>
  </contextOfUse>
</component>

```

以下に、同じ Context of Use コードと Keyword の組み合わせを有するコンテンツ群の中で、Priority Number が小さい（例：Priority Number が 1000）Context of Use の後に別のコンテンツを配置する（すなわち、並べ替える）例を示す（例：過去の提出で製造者および成分の Keyword とともに m3.2.s.2.3 を提出した場合）。

```

<component>
  <priorityNumber value="1500" updateMode="R"/>
  <contextOfUse>
    <id root="1f080afd-f5d4-4cec-8d09-2bf0ea6bec66"/>

```

```
<statusCode code="active"/>
</contextOfUse>
</component>
```



色の使用方法については [XML での色の使用例](#) を参照のこと。

### 8.2.10.2.3 Context of Use 要素の削除

改訂時の Submission Unit では、既存の Context of Use を削除する必要がある（つまり、既存の Context of Use が他の Context of Use に置換されない）ことがある。その場合、既存の Context of Use は active として表示されなくなる。

#### Context of Use の削除

```
<component>
  <priorityNumber value="900"/>
  <contextOfUse>
    <id root="d5528cfc-15f8-479e-ab59-562c0aa3a5d8"/>
    <statusCode code="suspended"/>
  </contextOfUse>
</component>
```



v3.2.2 から v4.0 に移行した Application については、コンテンツが使用されなくなった場合、または新しい v4.0 の Context of Use コードと Keyword コードの組み合わせの下に配置される必要がある場合、コンテンツが削除されることがある。詳細については、セクション 8.2.15.4 を参照のこと。



色の使用方法については [XML での色の使用例](#) を参照のこと。

### 8.2.10.2.4 Context of Use 要素の置換（バージョン更新）

改訂時の Submission Unit では、新しい contextOfUse 要素によって既提出の contextOfUse 要素を置換する必要があることがある。置換の提出には以下の 2 通りの理由がある：

1. 申請資料内容（含まれている文書）が変更された。
2. 削除した申請資料を再提出する必要がある\*。

\*コンテンツを再提出しなければならないとき、新規の Context of Use のみをメッセージに含めて Document の識別子を参照するが、物理的なファイルは再提出する必要はない。

新しい contextOfUse 要素には、新しい一意の識別子が与えられ、付随するすべての属性が指定される。さらに、relatedContextOfUse 要素を使用して、置換される Context of Use を指定する。これは単なる関係性を示しており、relatedContextOfUse の一意の識別子を参照することのみを目的としている。relatedContextOfUse はシステムにより「無効 (obsolete)」とマークされる（詳細についてはセクション 8.2.7 を参照のこと）。注：一度置換された無効 (obsolete) な Context of Use は、以降、再び置換できない（すなわち、「無効 (obsolete)」とマークされた Context of Use を relatedContextOfUse 要素内で参照してはならない）。要素の priorityNumber を使用して、以前に提出された提出物内での適切な位置に基づき、コンテンツを正しい順序で配置すること。

（番号が既存の値の前または後にあるかを問わず）Priority Number を再度または新たに割り当ててもよい。

```
<component>
```

```

<priorityNumber value="1000"/>
<contextOfUse>
  <id root="b205bb7c-a222-4557-a954-0363dc122ca8"/>
  <code code="ich_2.7.1" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
  <statusCode code="active"/>
  <replacementOf typeCode="RPLC">
  <relatedContextOfUse>
    <id root="78b2f721-25f0-474d-914b-5efb026cc7f7"/>
    </relatedContextOfUse>
  </replacementOf>
  <derivedFrom>
  <!--Document-->
    <documentReference>
      <id root="6ee97feb-8cd1-4991-8c38-002f16102fca"/>
    </documentReference>
  </derivedFrom>
</contextOfUse>
</component>

```

注：上の例は、以前の Context of Use（すなわち、*relatedContextOfUse* 要素により参照される Context of Use）で用いた Priority Number を再度割り当てる場合。



v3.2.2からv4.0に移行したApplicationについても、置換は同じContext of UseコードとKeywordコードの組み合わせのContext of Useの間でのみ実行可能である。詳細については、セクション8.2.15.4を参照のこと。



色の使用方法についてはXMLでの色の使用例を参照のこと。

## 8.2.11 Sequence Number

*sequenceNumber@value* は通常（ただし、常にではない）、段階的に増加する数値で、1つまたは複数の Submission で順序と時系列を維持するために使用される。これは 1つの Application において一意の値である。申請者の Sequence Number の運用方法は v4.0 でも同じである。



**実装者への注意事項：**双方向通信の一部として発行される Sequence Number については、『地域ごとの実装ガイド』を参照のこと。これらはメッセージの送信者のみが識別できる独立した一連の値をもつ（すなわち、この値は送受信者間で共用されない）。

### 8.2.11.1 XML 上の配置

XML メッセージの *sequenceNumber* 要素は次のように配置する：

- *controlActProcess* >> *subject* >> *submissionUnit* >> *componentOfI* >> *sequenceNumber*

*componentOf* 要素の前に *subject* および *component* 要素を（この順序通りに）記述することができる。

XML の記述方法については、表 7: v4.0 XML メッセージ構造を参照のこと。

### 8.2.11.2 XML 要素

次の表に、*componentOfI.sequenceNumber* 要素で必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



*typeCode*はeCTD v4.0 XMLメッセージでは不要である。  
*typeCode*は「COMP」に固定されている。XMLメッセージに、この属性に別の値が含まれる場合、そのメッセージはスキーマに対して妥当でない。

### 8.2.11.2.1 *sequenceNumber*

要素	属性	多重度	値の型 例	説明 指示
<i>sequenceNumber</i>		[1..1]		本要素は <i>SequenceNumber</i> とその値のコンテナ要素である。
	<i>value</i>	[1..1]	数値 1、2、3 など	<i>sequenceNumber</i> 要素の <i>value</i> 属性は、 <i>submission</i> 要素内で Submission Unit を順序付けするために使用される整数である。
準拠事項 運用規則	<i>sequenceNumber@value</i> 属性は必須である。			
	<i>sequenceNumber</i> は正の整数。この値は「1」から始まり、1つずつ増加する。この値は「999999」を超えてはならない。			
除外要素/属性	eCTD v4.0 で不要な要素と属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>sequenceNumber@controlInformationExtension</i></li> <li>• <i>sequenceNumber@controlInformationRoot</i></li> <li>• <i>sequenceNumber@flavorId</i></li> <li>• <i>sequenceNumber@nullFlavor</i></li> <li>• <i>sequenceNumber@uncertaintyType</i></li> <li>• <i>sequenceNumber@validTimeHigh</i></li> <li>• <i>sequenceNumber@validTimeLow</i></li> <li>• <i>sequenceNumber.expression</i></li> <li>• <i>sequenceNumber.originalText</i></li> <li>• <i>sequenceNumber.uncertainty</i></li> <li>• <i>sequenceNumber.uncertainRange</i></li> </ul>			

### 8.2.11.3 用語



この要素に対応するコントロールド・ボキャブラリはない。

### 8.2.11.4 除外要素

*sequenceNumber* 要素で除外されるクラス要素はない。

### 8.2.12 XML サンプル : Submission Unit

次の例に Submission Unit 要素と、すべての Submission Unit に対応する各属性を示す。

```
<subject typeCode="SUBJ">
  <submissionUnit>
    <id root="0d84467e-f20b-42ad-a69a-63e61a4f7ea7"/>
    <code code="regional_sub_unit_type_1" codeSystem="2.16.840.1.113883.3.989.x.x.x"/>
    <title value="Original Submission for pain medication - acetylsalicylic acid tablets"/>
    <statusCode code="active"/>.....
    [追加の要素または属性がある場合は statusCode、title または code 要素の後に記述
     される。例えば、Submission Unit の種類によっては、subject あるいは component
     要素を選択できる。]
    ...
  </componentOf1>
</subject>
```

```

<sequenceNumber value="1"/>
<submission>
...
  [submission 要素に関する追加の要素または属性が記述される。詳細は『地域ごとの実装ガイド』を参照のこと。]
...
  <componentOf>
...
    [application 要素に関する追加の要素または属性が記述される。詳細はセクション 8.2.12 または『地域ごとの実装ガイド』を参照のこと。]
...
    <componentOf>
  </submission>
</componentOf1>
<componentOf2>
...
  [categoryEvent 要素に関する追加の要素または属性が記述される。詳細は『地域ごとの実装ガイド』を参照のこと。]
...
</componentOf2>
</submissionUnit>
</subject>

```



色の使用方法については XML での色の使用例 を参照のこと。



上記のサンプルで使用されている `codeSystem` OID は、地域別のコントロールド・ボキャブラリ OID を表すサンプル値である。



`Sequence Number` の詳細については『地域ごとの実装ガイド』を参照のこと（特に、1 つの `Submission Unit` に複数の `Submission` が含まれる場合）

### 8.2.13 Application

`application` 要素は XML メッセージにおける `document` 要素と `keywordDefinition` 要素を結びつけるものであるため、本セクションで説明する。`application` 要素の概念は地域により異なる。



注：本来、`Application` は地域固有の概念であるため、『地域ごとの実装ガイド』でも説明されている。

#### 8.2.13.1 XML 上の配置

XML メッセージの `application` 要素は次のように配置する：

- `controlActProcess>>subject>>submissionUnit>>componentOf1>>submission>>componentOf>>application`

XML の記述方法については、表 7:v4.0 XML メッセージ構造を参照のこと。

#### 8.2.13.2 XML 要素

次の例に `Application` 情報の XML を示す。この `Application` は、`submission` 要素とその属性の後の `componentOf` 要素として記述されている。

...

[Application 要素毎にXML 部分を繰り返す。Submission 要素は Application 要素の構成要素である。]

```
...
<componentOf>
  <application>
    <id>
      <item root="f23c558f-cd58-41bc-bf6f-c6d230d3d665" extension="NDA987654"/>
      ...
      <!--item 要素が記述されることがある。-->
      ...
    </id>
    <code code="us_application_type_1" codeSystem="2.16.840.1.113883.3.989.5.1.2.2.1.1"/>
    ...
    [追加の要素または属性 (例えば Application に関連する component、referencedBy、
    informationRecipient、reference、subject または holder) を提出する場合は、
    application.code の後に記述される。]
    ...
  </application>
</componentOf>
```



色の使用方法についてはXMLでの色の使用例を参照のこと。

次の表に、application 要素で必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



classCode と moodCode は CTD v4.0 XML メッセージでは不要である。classCode は「ACT」、moodCode は「EVN」に固定されている。これらの属性に別の値が含まれる場合、そのXMLメッセージはスキーマに対して妥当でない。

8.2.13.2.1 *application.id*

要素	属性	多重度	値の型 例	説明 指示
<i>id</i>		[1..1]		本要素は下位要素と属性のコンテナ要素であり、Applicationを一意に識別する情報を格納する。
<i>id.item</i>		[1..*]		本要素は下位属性のコンテナ要素であり、Applicationを一意に識別する情報を格納する。Applicationには地域によって様々な識別子が指定されるが、この一意のApplication 識別子それぞれに対して1つの <i>id.item</i> 要素を使用する必要がある。
	<i>root</i>	[1..1]	妥当な OID or UUID	<i>id.item</i> 要素の <i>root</i> 属性は <i>application</i> 要素のグローバル一意識別子を提供する。
	<i>extension</i>	[0..1]	テキスト  e.g., NDA123456 (Sample U.S. NDA value) NDA123456 (米国 NDA 値 のサンプル) など	<i>id.item</i> 要素の <i>extension</i> 属性は、地域固有の Application 追跡番号を示す場所を指定する。
準拠事項 運用規則	<i>id.item@root</i> は <i>application</i> 要素の必須属性である。			
	Application 識別子の割り当ておよび OID または UUID のいずれの値を <i>id.item@root</i> で提供すべきかについては、『地域ごとの実装ガイド』を参照のこと。			
除外要素/属性	eCTD v4.0 で不要な属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>id.item@identifierName</i></li> <li>• <i>id.item@scope</i></li> <li>• <i>id.item@reliability</i></li> <li>• <i>id.item@displayable</i></li> <li>• <i>id@validTimeLow</i></li> <li>• <i>id@validTimeHigh</i></li> <li>• <i>id@controlInformationRoot</i></li> <li>• <i>id@controlInformationExtension</i></li> <li>• <i>id@nullFlavor</i></li> <li>• <i>id@flavorId</i></li> <li>• <i>id@updateMode</i></li> </ul>			

8.2.13.2 application.code

要素	属性	多重度	値の型例	説明指示
code		[1..1]		本要素は Application のコード値を構成するコンテナ要素である。
	code	[1..1]	テキスト  us_application_type_1 など  『地域ごとの実装ガイド』参照	code 属性は、地域別のコントロールド・ボキャブラリ (NDA、MAA、Art-8-3、Art-10-1 など) に基づいて Application のコンテンツ・タイプを示す一意の値である。
	codeSystem	[1..1]	妥当な OID	codeSystem 属性は、コントロールド・ボキャブラリ・システムを一意に識別する識別子を指定する。  本属性の値はコード・システムに登録された OID でなければならない。
準拠事項 運用規則 除外要素/属性	Application には code@code 属性を必ず1つだけ指定すること。			
	詳細については『地域ごとの実装ガイド』を参照のこと。			
	eCTD v4.0 で不要な要素と属性は以下の通り。 <ul style="list-style-type: none"> <li>• code.displayName</li> <li>• code.originalText</li> <li>• code.translation</li> <li>• code.source</li> <li>• code@codeSystemName</li> <li>• code@codeSystemVersion</li> <li>• code@valueSet</li> <li>• code@valueSetVersion</li> <li>• code@codingRationale</li> <li>• code@validTimeLow</li> <li>• code@validTimeHigh</li> <li>• code@controlInformationRoot</li> <li>• code@controlInformationExtension</li> <li>• code@nullFlavor</li> <li>• code@flavorId</li> <li>• code@updateMode</li> </ul>			

8.2.13.3 用語



すべての ICH コントロールド・ボキャブラリは genericcode ファイルおよびスプレッドシートファイルに記載される。<sup>18</sup>

<sup>18</sup> 最終的な実装用語集は ESTR1 のウェブサイトに掲載される。





*application* タイプ・コードに関する地域固有の情報については、該当する『地域ごとの実装ガイド』を参照のこと。

#### 8.2.13.4 除外要素

*application* 要素で除外されるクラス要素はない。詳細については『地域ごとの実装ガイド』を参照のこと。

#### 8.2.14 Document

*document* 要素は、Application に関連する各 Document に関する情報を送信するために使用する。Document (PDF ファイルなど) は、規制当局による審査のために申請者によって作成される。Document は期間を経て変更される場合がある。1つの Document に複数の *contextOfUse* 要素を関連付けることができる。また、1つの Document が複数の Submission Unit で使用されることもある。

Document とそれに対応する一連の要素/属性を初めて送信したとき、その Document が作成されたと見なされる。Document が受信側システムによって識別されると、以降の使用時には識別子でその Document を参照できる。

##### 8.2.14.1 XML 上の配置

XML メッセージの *document* 要素は次のように配置する：

- `controlActProcess>>subject>>submissionUnit>>componentOf1>>submission>>componentOf>>application>>component >> document`

*component* 要素の前に *holder*、*subject*、または *reference* 要素を記述することができる。

XML の記述方法については、表 7:v4.0 XML メッセージ構造を参照のこと。

##### 8.2.14.2 XML 要素

次の表に、*document* 要素で必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



*classCode* と *moodCode* は eCTD v4.0 XML メッセージでは不要である。*classCode* は「DOC」、*moodCode* は「EVN」に固定されている。これらの属性に別の値が含まれる場合、その XML メッセージはスキーマに対して妥当でない。

*document* 要素に適用される条件は以下の通り：

- 1つまたは複数の *document* 要素を *application* 要素の後に記載することができる。

##### 8.2.14.2.1 document.id

要素	属性	多重度	値の型 例	説明 指示
<i>id</i>		[1..1]		本要素は Document の識別子のコンテナ要素である。
	<i>root</i>	[1..1]	妥当な UUID	<i>id</i> 要素の <i>root</i> 属性は、 <i>document</i> のグローバル一意識別子である。
<b>準拠事項</b>	<i>id@root</i> は必須の属性である。			

要素	属性	多重度	値の型 例	説明 指示
運用規則				<p><i>id@root</i> はすべての <i>document</i> 要素に対して一意であること。つまり、同じ <i>id@root</i> 値を使用して2つの Document を提出してはならない。</p> <p>注：申請者が複数地域にまたがる Document を管理している場合、複数地域にまたがる Document 識別子を用いてもよいが、Document 要素および物理ファイルは各地域または規制当局に提出すること。詳細については『地域ごとの実装ガイド』を参照のこと。</p>
除外要素/属性				<p>eCTD v4.0 で不要な属性は以下の通り：</p> <ul style="list-style-type: none"> <li>• <i>id@extension</i></li> <li>• <i>id@identifierName</i></li> <li>• <i>id@scope</i></li> <li>• <i>id@reliability</i></li> <li>• <i>id@displayable</i></li> <li>• <i>id@validTimeLow</i></li> <li>• <i>id@validTimeHigh</i></li> <li>• <i>id@controlInformationRoot</i></li> <li>• <i>id@controlInformationExtension</i></li> <li>• <i>id@nullFlavor</i></li> <li>• <i>id@flavorId</i></li> <li>• <i>id@updateMode</i></li> </ul>

#### 8.2.14.2.2 document.title

要素	属性	多重度	値の型 例	説明 指示
title		[1..1]		本要素は Document の <i>title</i> 要素のコンテナ要素である。
	<i>value</i>	[1..1]	テキスト  送信者が指定したタイトル  「General Information」 など	<p>本属性は <i>title</i> 要素の <i>value</i> 属性で、Document のタイトルを指定する。</p> <p>本属性の値は送信者が各 <i>document</i> に対して指定する値である。</p>
	<i>updateMode</i>	[0..1]	アルファベット  <i>R = Replace</i> など	<i>document.title</i> 要素を更新する場合は <i>updateMode</i> 属性を使用する。
準拠事項				<i>title@value</i> 属性はすべての Document で必須である。

要素	属性	多重度	値の型 例	説明 指示
運用規則				<p><i>title</i> 要素を使用して、文書ファイルを表示するときに人間が読める値を指定する。</p> <p><i>title</i> 要素の更新を送信する場合は、<i>title@updateMode</i> 属性に「R」の値を指定すること。既存の Document タイトルが更新されていない限り、<i>updateMode</i> は使用しないこと。つまり、タイトル値が同じ場合は <i>updateMode</i> を使用してはならない。</p> <p><i>document.title</i> の更新に関する詳細については、セクション 8.2.16.2 を参照のこと。</p>
除外要素/属性				<p>The following datatype elements and attributes may not be required by eCTD v4.0: eCTD v4.0 で不要な要素と属性は以下の通り：</p> <ul style="list-style-type: none"> <li>• <i>title.translation</i></li> <li>• <i>title@validTimeLow</i></li> <li>• <i>title@validTimeHigh</i></li> <li>• <i>title@controlInformationRoot</i></li> <li>• <i>title@controlInformationExtension</i></li> <li>• <i>title@nullFlavor</i></li> <li>• <i>title@flavorId</i></li> <li>• <i>title@language</i></li> </ul>

#### 8.2.14.2.3 document.text

要素	属性	多重度	値の型 例	説明 指示
<i>text</i>		[0..1]		本要素は Document に関する追加情報を示すコンテナ要素である。
	<i>integrityCheckAlgorithm</i>	[0..1]	英数字 <i>SHA256</i> など	本属性は、 <i>integrityCheck</i> 要素で指定されているチェックサム値に使用された <i>integrityCheckAlgorithm</i> の型である。
	<i>language</i>	[0..1]	アルファベット 2文字言語コードについては ISO639.1 を参照。 『地域ごとの実装ガイド』を参照。	本属性は Document に使用されている言語を示す <i>language</i> 属性である。
	<i>mediaType</i>	[0..1]	テキスト <i>send</i> 、 <i>sdtm</i> または <i>adam</i> など	<i>mediaType</i> 属性は地域固有で求められるファイルの使用法を指定する。

要素	属性	多重度	値の型 例	説明 指示
	<i>updateMode</i>	[0..1]	アルファベット  R = Replace など	<i>updateMode</i> 属性は <i>text</i> 要素の属性の更新が必要であるかどうかを示すコード値を指定する。
<i>text.reference</i>		[0..1]		本要素は Document の <i>text</i> 要素に含まれるコンテナ要素である。
	<i>value</i>	[1..1]	テキスト  Document のファイル・パス 「../m3/32-bodydata/32s-drugsub/32s1-geninfo.pdf」など	これは <i>text</i> 要素の <i>value</i> 属性であり、Document の相対パスとファイル名を使用して Document の場所を指定する。
<i>text.integrityCheck</i>		[0..1]	英数字  「618102bf07065bc c1250594201fe4485 15f0fa61」など	本要素は、チェックサム値が含まれた <i>Integrity Check</i> 要素である。
<i>text.description</i>		[0..1]		本要素は、送信者が自身のシステム内で Document を記述するための <i>description</i> 要素である。
	<i>value</i>	[1..1]	テキスト  「26145c7a-3dc7-404d-91c1-6e0e5c71f8f6」 (UUID) または A1234567 (送信者指定の値) など	本属性は、申請者が自身のシステム内で定義する値を Document に指定するための <i>value</i> 属性である。注：受信者はこの値を使用しない。
<b>準拠事項</b>	Document には以下の要素と属性が必須である： <ul style="list-style-type: none"> <li>• <i>text</i> 要素 <ul style="list-style-type: none"> <li>○ <i>text@IntegrityCheckAlgorithm</i> 属性</li> <li>○ <i>reference@value</i> 属性</li> <li>○ <i>text.integrityCheck</i> 要素</li> </ul> </li> </ul>			

要素	属性	多重度	値の型 例	説明 指示
運用規則				Document を提出する場合は <i>text</i> 要素を使用すること。
				<i>text@language</i> および <i>text@mediaType</i> 属性の使用は任意である。 <i>document.text</i> 要素の更新に関する詳細についてはセクション 8.2.16.2.2 を参照のこと。詳細および使用可能な値の完全な一覧については『地域ごとの実装ガイド』を参照のこと。
除外要素/属性				ファイルを再利用するには、以前に提出された <i>document</i> 要素と同じ <i>reference@value</i> 、 <i>text@IntegrityCheckAlgorithm</i> 、および <i>text.integrityCheck</i> の値を <i>text</i> 要素で提示しなければならない。 <i>reference@value</i> は Document への相対パスとすること。同じ Submission Unit 内で再利用する場合、相対パスは同値となる。複数の Submission Unit にまたがって再利用する場合、相対パスには Sequence Number を示す必要があり、場合によっては Application Number も示す必要がある。 <i>document.reference</i> 要素の更新に関する詳細についてはセクション 8.2.16.3 を参照のこと。
				Document タイトルを更新する場合を除き、 <i>text.integrityCheckAlgorithm</i> および <i>text.integrityCheck</i> は必須である。
				eCTD v4.0 で不要な要素と属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>text.xml</i></li> <li>• <i>text.data</i></li> <li>• <i>text.thumbnail</i></li> <li>• <i>text.translation</i></li> <li>• <i>text@charset</i></li> <li>• <i>text@compressiontext@translation</i></li> <li>• <i>text@validTimeLow</i></li> <li>• <i>text@validTimeHigh</i></li> <li>• <i>text@controlInformationRoot</i></li> <li>• <i>text@controlInformationExtension</i></li> <li>• <i>text@nullFlavor</i></li> <li>• <i>text@flavorId</i></li> </ul>

### 8.2.14.3 用語



すべての ICH コントロールド・ボキャブラリは genericcode ファイルおよびスプレッドシートファイルに記載される。<sup>19</sup>

### 8.2.14.4 除外要素

*document* 要素で除外されるクラス要素はない。

### 8.2.14.5 XML サンプル : Document

以下の XML サンプルに、Application に指定された *document* 要素を示す。

```
<document>
  <id root="973d9293-77b9-4f45-b62e-aae62d7ce814"/>
  <title value="Process and Controls"/>
</document>
```

<sup>19</sup> 最終的な実装用語集は ESTRI のウェブサイトに掲載される。

```

<text integrityCheckAlgorithm="SHA256">
  <reference value="m3/32-prod/manuf-process-and-controls.pdf"/>
  <integrityCheck>c0d5623550c997a70b62717d95fca1cada201754d1ed9fbbbbfa97bfd64c8ea4</integrityCheck>
</text>
</document>

```

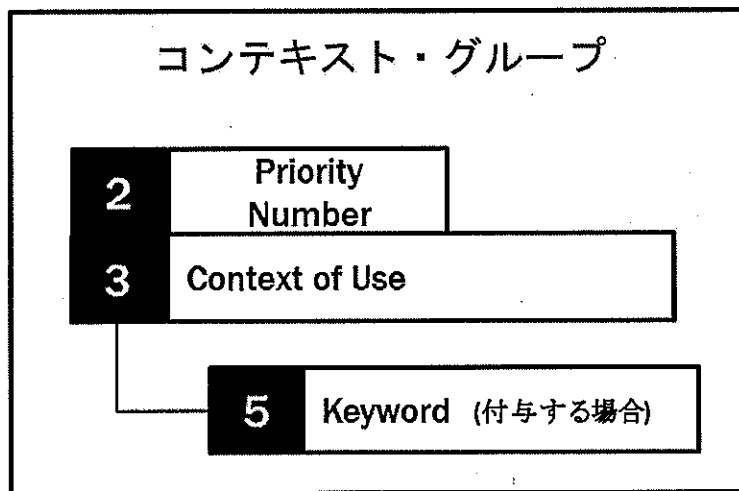


色の使用方法についてはXMLでの色の使用例を参照のこと。

### 8.2.15 コンテキスト・グループの変更への対処方法

ここでは、同一の Context of Use コードおよび Keyword コード (Keyword が存在する場合) の組み合わせを持つ 1 つまたは複数の Context of Use で構成されたコンテキスト・グループが、時間の経過に伴って変更された場合に、*contextOfUse* 要素を管理する方法を説明する。

図 4: コンテキスト・グループ・モデル



次のセクションでは、Keyword を使用して 1 つまたは複数の Context of Use 要素をグループ化する方法と、コンテンツ構成の変更、すなわち 1 つの Context of Use から参照される 1 文書を複数の Context of Use から参照される文書群からなる 1 グループへ、またはその逆へ変更する方法を説明する。

#### 8.2.15.1 グループ・タイトルでの Keyword の使用方法

送信者は Keyword を使用して、グループ・タイトルを Context of Use に追加し、目次の見出しに表示されるコンテンツを整理することができる。送信者によって、CTD 見出しがグループ・タイトルで整理される必要があると判断される場合、*contextOfUse* 要素が 1 つだけであっても、Keyword を用いること。Context of Use に適用するグループ・タイトルの Keyword は 1 つのみとすること。

```

<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="1f271446-8d56-4ddc-b730-eaee208c7053"/>
    <code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <!--参照される Documentは"Analytical Procedure 1"である。-->
    <derivedFrom>
      <documentReference>
        <id root="164af1e4-f625-4621-8d69-ca56b8f7dc7b"/>

```

</documentReference>  
 </derivedFrom>  
 <!--C001はコンテナのKeyword Definitionで"PVDC Blister Pack"を表すコードである。-->

<referencedBy typeCode="REFR">  
   <keyword>  
     <code code="C001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>  
   </keyword>  
 </referencedBy>  
 <!--GT001はグループ・タイトルのKeyword Definitionで"Analytical Procedures"を表すコードである。-->  
 <referencedBy typeCode="REFR">  
   <keyword>  
     <code code="GT001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>  
   </keyword>  
 </referencedBy>  
 </contextOfUse>

</component>  
 <component>  
   <priorityNumber value="2000"/>  
   <contextOfUse>  
     <id root="4a5c97e1-4448-47e2-90ff-2d6a264167c0"/>  
     <code code="ich\_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>  
     <statusCode code="active"/>  
     <!--参照されるDocumentは"Analytical Procedure 2"である。-->  
     <derivedFrom>  
       <documentReference>  
         <id root="0127b8b6-5510-45c5-93fd-9a3a6e9735aa"/>  
       </documentReference>  
     </derivedFrom>  
     <!--C001はコンテナのKeyword Definitionで"PVDC Blister Pack"を表すコードである。-->

<referencedBy typeCode="REFR">  
   <keyword>  
     <code code="C001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>  
   </keyword>  
 </referencedBy>  
 <!--GT001はグループ・タイトルのKeyword Definitionで"Analytical Procedures"を表すコードである。-->  
 <referencedBy typeCode="REFR">  
   <keyword>  
     <code code="GT001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>  
   </keyword>  
 </referencedBy>  
 </contextOfUse>



Context of Useのkeywordはすべて例示の目的でのみ示す。使用可能な組み合わせについては、コントロールド・ボキャブラリおよびICHM4「医薬品の承認申請のためのコモン・テクニカル・ドキュメント (CTD) の構成」ファイルを参照のこと。



色の使用方法についてはXMLでの色の使用例を参照のこと。

### 8.2.15.2 単一ファイルから複数ファイルへの変更

このシナリオでは、1つのファイル（Document）を多数の Document から成るコンテンツで置換する場合の、コンテンツ構成の変更について説明する。

#### シーケンス 1

初回の Submission Unit では、Context of Use から参照される Document を表示するために、以下の Context of Use を提出する。

```
<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="1f271446-8d56-4ddc-b730-eaee208c7053"/>
    <code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <!--参照される Document は "Analytical Procedure 1" である。-->
    <derivedFrom>
      <documentReference>
        <id root="164af1e4-f625-4621-8d69-ca56b8f7dc7b"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>
```

#### シーケンス 2

次の例では、前のシーケンスで提出された Context of Use を 2つの新しい Context of Use 要素で置換する。各 Context of Use 要素はそれぞれ別の Document を参照している（すなわち、コンテンツが 2つの新しい Document に分割されるが、一方の Document が改訂であり、もう一方の Document が追加である場合、あるいは両方の Document が新たなコンテンツを含む場合）。Related Context of Use は、前のシーケンスで提出された Context of Use の識別子を参照する。これは同じ見出しのまま、1つの Document が 2つの Document で置換されることを表す。注：以下の例では、置換後の 2つの Context of Use 要素のうち、最初の方に Priority Number が再度割り当てられる。送信者が新しい Priority Number を割り当てている場合もある。

```
<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="0c0abab8-cbfa-4d2f-9793-2b30ea51b8f5"/>
    <code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <replacementOf typeCode="RPLC">
      <relatedContextOfUse>
        <id root="1f271446-8d56-4ddc-b730-eaee208c7053"/>
      </relatedContextOfUse>
    </replacementOf>
    <!--参照される Document は "Analytical Procedure 1" である。-->
    <derivedFrom>
      <documentReference>
        <id root="164af1e4-f625-4621-8d69-ca56b8f7dc7b"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>
<component>
  <priorityNumber value="2000"/>
  <contextOfUse>
```



```

<id root="4a5c97e1-4448-47e2-90ff-2d6a264167c0"/>
<code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
<statusCode code="active"/>
<replacementOf typeCode="RPLC">
  <relatedContextOfUse>
    <id root="1f271446-8d56-4ddc-b730-eaee208c7053"/>
  </relatedContextOfUse>
</replacementOf>
<!--参照される Documentは"Analytical Procedure 2"である。-->
<derivedFrom>
  <documentReference>
    <id root="0127b8b6-5510-45c5-93fd-9a3a6e9735aa"/>
  </documentReference>
</derivedFrom>
</contextOfUse>
</component>

```



色の使用方法についてはXMLでの色の使用例を参照のこと。

### 8.2.15.3 複数ファイルから単一ファイルへの変更

このシナリオでは、複数のファイルで構成されたコンテンツを、1つのファイルで構成されるコンテンツで置換する方法を説明する。最初のシーケンスで多数の *contextOfUse* 要素（すなわち複数の Document）を送信し、後続のシーケンスでは1つのファイルを参照したい場合、コンテンツを1つの物理ファイルに結合する。

シーケンス 1-複数の document がそれぞれ参照される

次のサンプルに、それぞれが別の Document を参照する2つの *contextOfUse* 要素を示す。

```

<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="0c0abab8-cbfa-4d2f-9793-2b30ea51b8f5"/>
    <code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <!--参照される Documentは"Analytical Procedure 1"である。-->
    <derivedFrom>
      <documentReference>
        <id root="164af1e4-f625-4621-8d69-ca56b8f7dc7b"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>
<component>
  <priorityNumber value="2000"/>
  <contextOfUse>
    <id root="4a5c97e1-4448-47e2-90ff-2d6a264167c0"/>
    <code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <!--参照される Documentは"Analytical Procedure 2"である。-->
    <derivedFrom>
      <documentReference>
        <id root="0127b8b6-5510-45c5-93fd-9a3a6e9735aa"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>

```

```
</component>
```

### シーケンス 2-参照される単一の Document

次の例では、前のシーケンスで提出された 2 つの Context of Use 要素を、1 つの Document を参照する 1 つの Context of Use で置換する（したがってこの Document には、前のシーケンスで提出された 2 つの異なる Document で提出されたコンテンツがすべて含まれる）。注：以下の例では、新しい *contextOfUse* 要素に Priority Number が再度割り当てられる。送信者が新しい Priority Number を割り当てている場合もある。

```
<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="49e18e35-fe1b-4929-bf30-ea58c81ec30f"/>
    <code code="ich_3.2.p.7" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <replacementOf typeCode="RPLC">
      <relatedContextOfUse>
        <id root="0c0abab8-cbfa-4d2f-9793-2b30ea51b8f5"/>
      </relatedContextOfUse>
    </replacementOf>
    <replacementOf typeCode="RPLC">
      <relatedContextOfUse>
        <id root="4a5c97e1-4448-47e2-90ff-2d6a264167c0"/>
      </relatedContextOfUse>
    </replacementOf>
    <!--参照されるDocumentは"Analytical Procedure Consolidated"である-->
    <derivedFrom>
      <documentReference>
        <id root="e8e44446-de99-4324-ba9c-502fe8d729ba"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>
```



色の使用方法についてはXMLでの色の使用例を参照のこと。

### 8.2.15.4 階層構造の変更

提出物に変更された場合、または最新の eCTD 実装では当該見出しが無効となった場合、CTD 見出しの階層構造の変更が必要となることがある。最新の eCTD 実装における CTD 見出しの使用法は ICH 文書「医薬品の承認申請のためのコモン・テクニカル・ドキュメント (CTD) の構成」で定められている。提出物の将来の変更は、各 CTD 見出しで使用可能な Context of Use コードまたは Keyword に影響する可能性がある。階層構造の変更に対処する際の指針を以下に示す。

#### 8.2.15.4.1 以前は使用可能であった CTD 見出しの削除

(以前は使用可能であった) CTD 見出しが使用不能となった場合、以下のシナリオが生じる：

- 提出物は既存の見出しの下でも引き続き使用されるが、追加の情報の提供が必要な場合。
  - 新しい提出物を新しい妥当な Context of Use コードおよび Keyword とともに提出すること。
  - 提出物は各 CTD 見出しおよび Keyword の下で表示される。

- 既存の提出物と新しい提出物の間の関係性（すなわちライフサイクル）はなくなる。
- 既存の提出物は使用されなくなり、新しい提出物の提出が必要な場合：
  - 既存の提出物を削除する必要がある（セクション 8.2.10.2.3 参照）。
  - 新しい妥当な Context of Use コードおよび Keyword とともに、新しい提出物を提出する。
  - 削除された提出物は active でなくなり、新しい提出物が新しい CTD 見出しおよび Keyword の下で active となる。
  - 既存の提出物と新しい提出物の間の関係性（すなわちライフサイクル）はなくなる。
- 提出物を置き換える必要があるが、CTD 見出しおよび/または Keyword の変更が先に必要な場合。
  - 既存の提出物を削除する必要がある（セクション 8.2.10.2.3 参照）。
  - 新しい CTD 見出し、Keyword および既存の提出物の Document 識別子への参照とともに、新しい Context of Use が提出される。
  - 将来のシーケンスでは、同じ CTD 見出し、Keyword および既存のコンテンツへの参照（すなわち、related Context of Use）を伴う新しい Context of Use の下で、新しい提出物を指定する。
  - 提出物は、新しい CTD 見出しおよび Keyword の下で、既存の提出物と新しい提出物の間の関係を示す。

#### 8.2.15.4.2 以前は使用不能であった CTD 見出しの追加

（以前は使用不能であった）CTD 見出しが使用可能となり、新しい CTD 見出しの下ではより高位または低位の提出物の方が適している場合。

- 提出物は既存の見出しの下でも引き続き使用されるが、新しい CTD 見出しの下では追加の情報の提供が必要な場合。
  - 新しい提出物を新しい妥当な Context of Use コードおよび Keyword とともに提出すること。
  - 提出物は各 CTD 見出しおよび Keyword の下で表示される。
  - 既存の提出物と新しい提出物の間に関係（すなわちライフサイクル）はなくなる。
- 既存の提出物は使用されないが、新しい提出物の提出が必要な場合。
  - 既存の提出物を削除する必要がある（セクション 8.2.10.2.3 参照）。
  - 新しい妥当な Context of Use コードおよび Keyword とともに、新しい提出物を提出する。
  - 削除された提出物は active でなくなり、新しい提出物が新しい CTD 見出しおよび Keyword の下で active となる。
  - 既存の提出物と新しい提出物の間の関係性（すなわちライフサイクル）はなくなる。

- 提出物の置換が必要であるが、CTD 見出しおよび/または Keyword の変更が先に必要な場合。
  - 既存の提出物を削除する必要がある（セクション 8.2.10.2.3 参照）。
  - 新しい CTD 見出し、Keyword および既存の提出物の Document 識別子への参照とともに、新しい Context of Use が提出される。
  - 将来のシーケンスでは、同じ CTD 見出し、Keyword および既存のコンテンツへの参照（すなわち、related Context of Use）を伴う新しい Context of Use の下で、新しい提出物を指定する。
  - 提出物は、新しい CTD 見出しおよび Keyword の下で、既存の提出物と新しい提出物の間の関係を示す。

## 8.2.16 その他の考慮事項

### 8.2.16.1 Document の再利用

Application のライフサイクルの中で、1 つの Document が何度も参照されることがある。そのため、Document の再利用は eCTD v4.0 の重要な機能である。Document の再利用は、文書のコンテンツとメタデータが、再利用する Context of Use にも適用可能な場合のみ可能である。また、Document の再利用にあたっては、文書内容として記載される参照先や、ハイパーテキスト・リンク先を含む全てのコンテンツが、再利用可能でなければならない。当該提出に関連のない文書への参照を含む結果となる場合は、Document を再利用すべきではない。本セクションでは、Document 再利用の最も一般的な例を示す。

Document を再利用するための Document 保管方法と規則については、『地域ごとの実装ガイド』を参照のこと。

同じ文書を 1 つまたは複数の Submission Unit に含めて送信する場合、当該文書の識別子を設定する *document* 要素は 1 度だけ提出すればよい。その後は、どの *contextOfUse* 要素からでもその Document を参照できる。

次の XML に、文書の識別子によって同じ Document を参照する 2 つの *contextOfUse* 要素を示す。

#### シーケンス 1

##### Context of Use 要素

```

<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="7480bc1a-6486-4714-8d32-a3bd41de9be6"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <documentReference>
        <id root="3d1084fb-56c6-4923-a1e5-8a15e4fdc9c5"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>

```

*Document.id*  
シーケンス 1 で  
指定

##### Document 要素

```
<document>
```

```

<id root="3d1084fb-56c6-4923-a1e5-8a15e4fdc9c5"/>
<title value="Excipients X"/>
<text integrityCheckAlgorithm="SHA256">
  <reference value="m3/32-prod/excipients.pdf"/>
  <integrityCheck>c0d5623550c997a70b62717d95fca1cada201754d1ed9fbbbf97bfd64c8ea4</integrityCheck>
</text>
</document>

```

Document.id

## シーケンス 2

### Context of Use 要素

```

<component>
  <priorityNumber value="2000"/>
  <contextOfUse>
    <id root="47939431-1ac1-4e17-b44d-dcea7ce43050"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <documentReference>
        <id root="3d1084fb-56c6-4923-a1e5-8a15e4fdc9c5"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>

```

Document.id

シーケンス 1 で指定



色の使用方法についてはXMLでの色の使用例を参照のこと。

### Document 要素

*document* 要素はこの Submission Unit では送信されない。この Document はシーケンス 1 ですでに送信済みである。上記の例では、Document は同じ Application 内または複数の Application から参照されている。Document の再利用に関する詳細については『地域ごとの実装ガイド』を参照のこと。

#### 8.2.16.2 Document 要素の更新

*document* 要素は、最初の提出後に更新を要する可能性があるいくつかの属性を含む。更新は、当該 Document の初回提出時に含まれる必須および任意の要素提出に基づいて実行される必要がある。*document* 要素の例をすべての必須および任意の要素および属性とともに以下に示す。

#### document の最初の提出

```

<document>
  <id root="ceb05f3d-ebb0-4547-9734-056efa134a7a"/>
  <title value="Process and Controls"/>
  <text integrityCheckAlgorithm="SHA256" mediaType="sdtm" language="en" >
    <reference value="m3/32-prod/manuf-process-and-controls.pdf"/>
    <integrityCheck>a4c828974a7d177137d69aedfc45379a694611ef317c6c1741a935aa9555c57d</integrityCheck>
  </text>
</document>

```

**Document** 要素の更新については、Document タイトルの更新、**text** 要素の更新および両要素の更新に関する以下のサブセクションで説明する。以下の条件は、**text** 要素およびタイトルの更新の両方に適用される：

- **text@integrityCheckAlgorithm**、**text.integrityCheck** および **reference@value** がタイトルおよび/または **text** 要素の更新とともに提出された場合、それらの値は受信者により無視される。最初の Document の提出後、これらの要素または属性の値は変更できない。

#### 8.2.16.2.1 Document タイトルの更新

**document.title** 要素にエラーが含まれた **document** 要素が送信された場合、送信者は、新しい Document を作成し直す代わりに、**document.title** 要素を更新することができる。本セクションの例に、当該更新に必要な要素を示す。

##### **updateMode** を使用した Document タイトルの更新

**document.title@value** 属性値の誤り（例：誤字）を修正する場合、Document の **id@root** 値は同じままである。Document タイトルを更新すると、当該 Document が参照されているのすべての箇所で更新される。

```
<document>
  <id root="ceb05f3d-ebb0-4547-9734-056efa134a7a"/>
  <title value="Manufacturing Process and Controls" updateMode="R"/>
</document>
```

**document.title@value** が全く異なる場合（すなわち、新しいタイトルの指定を意図する場合）、新しい Document オブジェクトを提出すること。このとき、ファイルが再利用されることとなる（次のセクション参照）。

#### 8.2.16.2.2 Document Text の更新

**mediaType** および **language** 属性が更新され、同じ **updateMode** 属性を共有する。

**document.text@mediaType** または **document.text@language** 属性中のエラーまたは欠損値を修正しても Document の **id@root** 値は同じままである。このセクション中の例では当該更新の必須要素を示す。

```
<component>
  <document>
    <id root="ceb05f3d-ebb0-4547-9734-056efa134a7a"/>
    <text mediaType="sdtm" language="en" updateMode="R"/>
  </document>
</component>
```

Document の **text** 要素を更新（値の追加か置換かを問わない）する場合、**updateMode** に「R」の値を指定し、送信者が以前に提出した情報の上書きを意図したことを示す。

**mediaType** および **language** 要素の使い方に関する指示については『地域ごとの実装ガイド』を参照のこと。

#### 8.2.16.2.3 Document タイトルおよびテキストの更新

Document の **title** および **text** 要素の両方の更新を要する場合、それぞれの更新方法を組み合わせる。

`document.title@value`、`document.text@mediaType` または `document.text@language` 属性中のエラーまたは欠損値を修正しても、Document の `id@root` 値は同じままである。このセクション中の例では当該更新の必須要素を示す。

```
<component>
  <document>
    <id root="ceb05f3d-ebb0-4547-9734-056efal34a7a"/>
    <title value="Manufacturing Process and Controls" updateMode="R"/>
    <text mediaType="sdtm" language="en" updateMode="R"/>
  </document>
</component>
```



色の使用方法については[XMLでの色の使用例](#)を参照のこと。

### 8.2.16.3 ファイルの再利用

ファイルは通常、1つの `document` 要素で記述され、その `document` 要素は複数の `contextOfUse` 要素から参照できる。これについては「Document の再利用」セクションで説明している。特定の状況下では、同じファイルを用途によって異なる方法（異なる Document タイトル）で表さなければならないことがある。このような状況においては、ファイルを新たな別の `document` 要素で記述する必要がある。このため、複数の `document.text` 要素に同じファイルパスが出現する場合がある。このファイルはフォルダ構造に含めて 1 度だけ送信する必要がある。これについては「Document の再利用」で説明している（セクション 8.2.16.1 を参照）。

新しい `document` 要素を定義する際、以前に提出したファイルのパスを指定すると、複数の Submission および Application（下記の注を参照）を越えて同じファイルを再利用できる。ファイルは、当該ファイルが実際に提出されたフォルダの場所から取得される。想定される再利用のシナリオは以下の通り：

- 第一階層のフォルダが異なる（他の Application 内に存在する）ファイルの再利用
- 第二階層のフォルダが異なる（最新の Application の他のシーケンス内に存在する）ファイルの再利用
- 第二階層のフォルダが同じ（同じ Submission Unit 内に存在する）ファイルの再利用

注：ファイルを再利用する場合、つまり、そのファイルをすでに提出済みである場合、フォルダ構造上のファイル位置を頼りに操作するのは難しくなる。なぜなら、Submission Unit、Submission、または Application に含めて提出するすべてのファイルが、同じフォルダ構造内に存在するとは限らないからである。

注：ファイルの再利用に関する地域別の情報については、『地域ごとの実装ガイド』を参照のこと。

次の XML サンプルに、「Report for Study 1」というタイトルの `document` 要素を示す。

```
<component>
  <document>
    <id root="bab246ef-7d8e-4042-bd8b-ad9769f4589b"/>
    <title value="Report for Study 1"/>
    <text integrityCheckAlgorithm="SHA256" language="en">
      <reference value="../m5/531-biopharm/report1.pdf"/>
      <integrityCheck>5b94eb14cd31031a4d4539d0bcfbef028a91c04d2d2575990c4422947a9f437a
    </integrityCheck>
    </text>
  </document>
</component>
```

```
</document>
</component>
```

次の *document* 要素には、上記の例と同じファイルが若干異なる Document タイトルとともに記述されている。この *document* 要素は、ファイルの初回提出後に提出した後続の Submission Unit によって作成されている。これらの例で指定されているファイルパス情報の違いに注意すること。

```
<component>
  <document>
    <id root="79da2f37-02a8-4dcd-8552-54565b093c08"/>
    <title value="Summary Report for Study 1"/>
    <text integrityCheckAlgorithm="SHA256" language="en">
      <reference value="../sequencefolder/m5/531-biopharm/report1.pdf"/>
      <integrityCheck>5b94eb14cd31031a4d4539d0bcfbef028a91c04d2d2575990c4422947a9f437a
    </integrityCheck>
    </text>
  </document>
</component>
```



色の使用方法についてはXMLでの色の使用例を参照のこと。

## 8.2.17 Keyword Definition

*keywordDefinition* は送信者が定義する Keyword を提供するために使用され、メッセージの他の部分ではコードで参照される。Keyword Definition の主な使用目的は、コントロールド・ボキャブラリで定義されていない Keyword (送信者特有の Keyword など) の値を定義することである。Keyword Definition には名前と値が対になって含まれており、Context of Use で Keyword を指定するために使用される。注：申請者は複数の Application 間で再利用可能な方法で Keyword Definition を定義してもよいが、その場合は Application ごとに同じ Keyword Definition を提出する必要がある。

### 8.2.17.1 XML 上の配置

Keyword Definition の場合、XML メッセージの *keywordDefinition* 要素を次のように配置する。

- *controlActProcess*>>*subject*>>*submissionUnit*>>*componentOf1*>>*submission*>>*componentOf*>>*application*>>*referencedBy*>>*keywordDefinition*

*referencedBy* 要素の前に、*informationRecipient*、*holder*、*reference*、または *subject* 要素を記述することができる。

XML の記述方法については、表 7: v4.0 XML メッセージ構造を参照のこと。

### 8.2.17.2 XML 要素

次の表に、*keywordDefinition* 要素で必要なすべての XML 要素および属性を示し、特別な指示がある場合はこれを付す。



*ClassCode* と *moodCode* は eCTD v4.0 XML メッセージでは不要である。*classCode* は「ACT」、*moodCode* は「EVN」に固定されている。これらの属性に別の値が含まれる場合、その XML メッセージはスキーマに対して妥当でない。



各 *keywordDefinition* は、それ自身の *keywordDefinition* 要素に含めて送信する。スキーマでは、各 *keywordDefinition* に複数の値を記載することが許可されているが、eCTD v4.0 では 1 つの *keywordDefinition* 要素につき 1 つの値のみ記載することが許可される。

*keywordDefinition* 要素に適用される条件は以下の通り：

- 各 *application* 要素につき 0 から複数の *keywordDefinition* 要素を送信できる。
- 送信者が定義する Keyword を使用する場合は、対応する *keywordDefinition* を指定する必要がある。
- *keywordDefinition* は Application のライフサイクルを通して 1 度だけ提出すればよい。つまり、Keyword Definition は 1 度定義すれば、同じ Application 内であれば割り当てられたコード値を使用してそれを参照することができる。注：*keywordDefinition* は Application ごとに定義する必要がある。

#### 8.2.17.2.1 *keywordDefinition.code*

要素	属性	多重度	値の型 例	説明 指示
<i>code</i>		[1..1]		本要素は Keyword Definition の種類を識別する情報を格納するコンテナ要素である。
	<i>code</i>	[1..1]	テキスト <i>ich_keyword_type_1</i> など	本属性は Keyword Definition の種類を示すコード値に対応する <i>code</i> 属性である。
	<i>codeSystem</i>	[1..1]	妥当な OID	<i>codeSystem</i> 属性は、コントロール・ボキャブラリ・システムを一意に識別する識別子を指定する。  本属性の値はコード・システムに登録された OID でなければならない。
準拠事項	<i>code</i> と <i>codeSystem</i> は必須の属性である。			
運用規則	<i>code</i> は妥当な ICH Keyword コードの種類であること。			

要素	属性	多重度	値の型例	説明指示
除外要素/属性	eCTD v4.0 で不要な要素と属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>code.displayName</i></li> <li>• <i>code.originalText</i></li> <li>• <i>code.translation</i></li> <li>• <i>code.source</i></li> <li>• <i>code@codeSystemName</i></li> <li>• <i>code@codeSystemVersion</i></li> <li>• <i>code@valueSet</i></li> <li>• <i>code@valueSetVersion</i></li> <li>• <i>code@codingRationale</i></li> <li>• <i>code@validTimeLow</i></li> <li>• <i>code@validTimeHigh</i></li> <li>• <i>code@controlInformationRoot</i></li> <li>• <i>code@controlInformationExtension</i></li> <li>• <i>code@nullFlavor</i></li> <li>• <i>code@flavorId</i></li> <li>• <i>code@updateMode</i></li> </ul>			

#### 8.2.17.2.2 keywordDefinition.statusCode

要素	属性	多重度	値の型例	説明指示
<i>statusCode</i>		[1..1]		本要素は、 <i>keywordDefinition</i> の状態を示すコンテナ要素である。
	<i>code</i>	[1..1]	アルファベット active など	本属性はステータスを示すコード値である。
準拠事項	<i>statusCode</i> は必須の要素である。			
運用規則	<i>code</i> 属性の値は常に「active」であること。			
除外要素/属性	eCTD v4.0 で不要な要素と属性は以下の通り： <ul style="list-style-type: none"> <li>• <i>statusCode.part</i></li> <li>• <i>statusCode@validTimeLow</i></li> <li>• <i>statusCode@validTimeHigh</i></li> <li>• <i>statusCode@controlInformationRoot</i></li> <li>• <i>statusCode@controlInformationExtension</i></li> <li>• <i>statusCode@nullFlavor</i></li> <li>• <i>statusCode@flavorId</i></li> <li>• <i>statusCode@updateMode</i></li> </ul>			

#### 8.2.17.2.3 keywordDefinition.value

要素	属性	多重度	値の型例	説明指示
<i>value</i>		[1..1]		本要素は <i>keywordDefinition</i> 側で Keyword を識別するためのコードのコンテナ要素である。
<i>value.item</i>		[1..1]		本要素は個々の Keyword 識別子を指定するコンテナ要素である。

要素	属性	多重度	値の型 例	説明 指示
	<i>code</i>	[1..1]	テキスト 送信者が定義した値  MANU001 または MFR_001 など	本属性は定義される Keyword に対応する <i>code</i> 属性である。
	<i>codeSystem</i>	[1..1]	テキスト 送信者が定義した値	本属性はコントロールド・ボキャブラリ・システムを一意に識別する <i>codeSystem</i> 値である。
<i>value.item.displayName</i>		[1..1]		本要素は <i>displayName</i> を指定するコンテナ要素である。 <i>displayName</i> とは、 <i>keywordDefinition</i> コードに対応する表示名である。
	<i>value</i>	[1..1]	テキスト  「Big Manufacturer」など送信者が定義した値	本属性は、定義される Keyword の <i>value</i> 要素の <i>displayName</i> 属性である。
	<i>updateMode</i>	[0..1]	アルファベット  R = Replace など	<i>updateMode</i> は、Keyword Definition の表示名の値を更新するために使用する。
準拠事項	<i>keywordDefinition.value</i> は必須の要素である。			
	<i>value.item@code</i> 、 <i>value.item@codeSystem</i> 、 <i>value.item.displayName@value</i> は必須の属性である。			
	<i>keywordDefinition.value.item@code</i> が指定する <i>keywordDefinition.value.item.displayName</i> の値が一意に定まらないとき（すなわち、 <i>updateMode</i> を用いた適切な更新が行われず、既存のコードに対して新規の <i>displayName@value</i> が提出されたとき）、その提出は却下される。			
運用規則	各 <i>keywordDefinition</i> には送信者が定義した Keyword を 1 つだけ含めることができる。			
	<i>displayName@value</i> は Keyword Definition が持つ情報のうち更新可能な唯一の属性であり、更新する場合は <i>displayName@updateMode</i> に「R」の値を指定すること。 <i>displayName@value</i> が更新されていない限り、 <i>updateMode</i> は使用しないこと。つまり、当該値が更新されていない場合は <i>update mode</i> を使用してはならない。			

要素	属性	多重度	値の型 例	説明 指示
除外要素/属性	eCTD v4.0 で不要な要素と属性は以下の通り： <ul style="list-style-type: none"> <li>• <code>displayName@controlInformationExtension</code></li> <li>• <code>displayName@controlInformationRoot</code></li> <li>• <code>displayName@flavorId</code></li> <li>• <code>displayName@language</code></li> <li>• <code>displayName@nullFlavor</code></li> <li>• <code>displayName@validTimeHigh</code></li> <li>• <code>displayName@validTimeLow</code></li> </ul>			

### 8.2.17.3 用語



すべてのICHコントロールド・ボキャブラリはgenericcodeファイルおよびスプレッドシートファイルに記載される。<sup>20</sup>

### 8.2.17.4 除外要素

`keywordDefinition` 要素で除外されるクラス要素はない。

### 8.2.18 XML サンプル : Keyword Definition

現時点で、Keywordの定義は1つのApplicationの範囲においてのみ有効である。しかし、地域固有の運用シナリオ（例：グループ化されたSubmission）を考慮して、同じKeyword Definition（コードおよび値の対）を複数のApplicationで再提出してもよい。次のセクションでは、Keywordの定義および使用におけるXMLインスタンスのシナリオを概説する。

#### 8.2.18.1 Keyword Definition

次のXMLサンプルに、`keywordDefinition`の種類の一つであるManufacturerを示す。

```
<referencedBy>
  <keywordDefinition>
    <code code="ich_keyword_type_3" codeSystem="2.16.840.1.113883.3.989.2.2.1.5.1"/>
    <statusCode code="active"/>
    <value>
      <item code="MANU001" codeSystem="2.16.840.1.113883.3.X.Y.Z">
        <displayName value="Big Manufacturer"/>
      </item>
    </value>
  </keywordDefinition>
</referencedBy>
```

注：スキーマでは複数のitem要素を記述することが認められているが、eCTD v4.0では1つのKeyword Definitionにつき1つだけ指定すること。



色の使用方法についてはXMLでの色の使用例を参照のこと。

<sup>20</sup> 最終的な実装用語集はESTRIのウェブサイトに掲載される。

### 8.2.18.2 Keyword Definition の表示名の更新

送信した Keyword Definition に誤りが含まれていることがある。送信者が Keyword Definition の表示名を修正する必要がある場合（同じ概念や語句の表記（綴りなど）を正す場合など）は、その表示名だけを更新できる。Keyword Definition コードは変更できない。

#### シーケンス 1

```
<referencedBy>
  <keywordDefinition>
    <code code="ich_keyword_type_3" codeSystem="2.16.840.1.113883.3.989.2.2.1.5.1"/>
    <statusCode code="active"/>
    <value>
      <item code="MANU001" codeSystem="2.16.840.1.113883.3.X.Y.Z">
        <displayName value="Ace Manufacturer"/>
      </item>
    </value>
  </keywordDefinition>
</referencedBy>
```

#### シーケンス 2

```
<referencedBy>
  <keywordDefinition>
    <code code="ich_keyword_type_3" codeSystem="2.16.840.1.113883.3.989.2.2.1.5.1"/>
    <statusCode code="active"/>
    <value>
      <item code="MANU001" codeSystem="2.16.840.1.113883.3.X.Y.Z" >
        <displayName value="Acme Manufacturer" updateMode="R"/>
      </item>
    </value>
  </keywordDefinition>
</referencedBy>
```

*displayName* を正しい値に変更

注：表示名を更新する場合、同じ Application 内のすべての Submission Unit（過去に提出したものも含む）に対して更新が適用される。この更新は他の Application の Submission Unit には適用されない。特定の Submission Unit にのみ表示名の更新を適用する場合、新しい Keyword Definition を定義すること。

### 8.2.18.3 複数の Submission Unit における Keyword Definition の使用

送信者によって 1 度送信された Keyword Definition は、その定義が更新されない限り再送する必要はない。Keyword コードは、Application 内のすべての Submission Unit を通して同じでなければならない。つまり、更新できるのは表示名のみである。個々の概念に対し、Keyword Definition コードと表示名の組を 1 つだけ定義すること。つまり、1 つの Application の中で同じ概念を複数回定義することはできない。

#### シーケンス 1

シーケンス 1 で定義された Keyword

```
<referencedBy>
  <keywordDefinition>
    <code code="ich_keyword_type_3"
    codeSystem="2.16.840.1.113883.3.989.2.2.1.5.1"/>
    <statusCode code="active"/>
    <value>
      <item code="MANU003" codeSystem="2.16.840.1.113883.3.X.Y.Z" >
```

```

        <displayName value="Simple Manufacturer"/>
    </item>
</value>
</keywordDefinition>
</referencedBy>

```

### シーケンス 1 の Context of Use で使用された Keyword Definition

```

<component>
  <priorityNumber value="1000"/>
  <contextOfUse>
    <id root="8c590801-c4ca-4940-bb4d-5a4cd32685d7"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <!--Document titled "Controls for Material YYY"-->
      <!--"Controls for Material YYY"というタイトルの文書-->
      <documentReference>
        <id root="d0c6463c-7538-4ac8-827d-65b083c3893d"/>
      </documentReference>
    </derivedFrom>
    <referencedBy typeCode="REFR">
      <keyword>
        <code code="MANU003" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>
      </keyword>
    </referencedBy>
    <referencedBy typeCode="REFR">
      <keyword>
        <code code="SUB001" codeSystem="2.16.840.1.113883.3.X.Y.Z"/>
      </keyword>
    </referencedBy>
  </contextOfUse>
</component>

```



Context of Use の keyword はすべて例示の目的でのみ示す。使用可能な組み合わせについては、コントロールド・ボキャブラリおよび ICHM4 「医薬品の承認申請のためのコモン・テクニカル・ドキュメント (CTD) の構成」 ファイルを参照のこと。



色の使用方法については [XML での色の使用例](#) を参照のこと。

### シーケンス 3 で定義された Keyword

この Keyword Definition について XML で送信する追加情報はない。シーケンス 1 で送信された値「MANU003」と表示名「Simple Manufacturer」が引き続き利用可能である。

### シーケンス 3 の Context of Use で使用された Keyword Definition

```

<component>
  <priorityNumber value="2000"/>
  <contextOfUse>
    <id root="64e51fb8-4608-4c3a-af52-68b5cc02345b"/>
    <code code="ich_3.2.s.2.3" codeSystem="2.16.840.1.113883.3.989.2.2.1.1.1"/>
    <statusCode code="active"/>
    <derivedFrom>
      <!--"Controls for Material BCD"というタイトルの文書-->
      <documentReference>
        <id root="23967c61-99bf-4090-863c-15b524ee242e"/>
      </documentReference>
    </derivedFrom>
  </contextOfUse>
</component>

```