



DaXtra Competency

アウトプットプロフィール内 Competency 要素に関する資料

アジア太平洋地域

お問い合わせ: +852 3695 5133

asia@daxtra.com

サポート: +852 3695 5133

support@daxtra.com

Unit 1205 OfficePlus,
93-103 Wing Lok Street,
Sheung Wan, Hong Kong

ヨーロッパ・中東・アフリカ地域

お問い合わせ: +44 20 7801 6323

busdev@daxtra.com

サポート: +44 131 653 1260

support@daxtra.com

Top Floor South, Harbour Point,
Newhailes Road, Musselburgh,
EH21 6QD, United Kingdom

米国

お問い合わせ: +1 804 767-1351

usa@daxtra.com

サポート: +1 804 767-1351

helpdesk@daxtra.com

3310 W. Clay Street,
Richmond,
VA 23230, United States

目次

CVパーサ: アウトプットプロフィール	3
CVパーサ: カスタマイズプロフィール	5
デフォルトスキルと顧客スキル	5
デフォルトアウトプット	5
正規化された Competency 名	5
JSON 形式	6
非正規化 Competency 名	6
カスタムアウトプット	6
マスタースキルフাইルのサンプル	7
Daxtra の抽出したスキルからのマッピング(MSF 内 auth)	7
DaXtra が抽出しなかったスキルの識別(MSF 内 alias)	8
スキルグルーピング	9
スキル拡張	10
Competency 要素	12
Competency 要素 : CompetencyId サブ要素	13
Competency 要素 : TaxonomyId サブ要素	13
Competency 要素 : CompetencyEvidence サブ要素	14
Competency 要素 : CompetencyWeight サブ要素	16
特殊な Competency の例	18
“A-Level” Competency 機能	18
自動車免許	19
セキュリティークリアランス	20
資格からの語学能力抽出	21
TOEIC 880 => English Excellent	21
IELTS 6.5 => English Intermediate	21
付録 A - XML/JSON 要約	22
付録 B: 用語解説	23

CV パーサ: アウトプットプロフィール

CV/履歴書が DaXtra のパーサによって処理される際、その結果は XML または JSON のアウトプットプロフィールとして出力されます。このプロフィールは CV から抽出された全ての情報を含んでいます。プロフィールには個人情報、職歴、学歴など様々な項目があり、抽出された情報は正規化(ノーマライズ)され、整理された構造要素へ配置されます。

DaXtra は強力なパーサエンジンと連動する大規模なスキルタクソノミー(DaXtra スキルタクソノミーまたは DST)を保有しており、多くのスキル、職業名、資格が CV から抽出され、プロフィール内の **Competency**(能力・特性)要素に配置されます。このドキュメントではその **Competency** 要素について、そしてその抽出された情報を活用する方法を特に解説します。

以下は XML と JSON 両者それぞれのデフォルトアウトプット内の **Competency** 要素の例です。CV 内に“マーケティングマネジメント”というスキルがこのように書かれていたとします：

マーケティングマネジメントを 10 年にわたり経験しました。

Competency 要素 XML の例：

```
<Competency oids='2 ' description='Skill' auth='yes' name='Marketing Management' >
  <TaxonomyId id='marketing~misc' idOwner='DAXTRA' description='marketing &#62; misc' />
  <CompetencyEvidence name='Extent of Experience' typeDescription='Extent of Experience'
type='YEAR' >
    <NumericValue description='YEAR' >10</NumericValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias' >
    <StringValue >Marketing Management</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias' >
    <StringValue >マーケティングマネジメントを10年</StringValue>
  </CompetencyEvidence>
  <CompetencyWeight type='skillLevel' >
    <NumericValue minValue='0' maxValue='100' >18</NumericValue>
  </CompetencyWeight>
</Competency>
```

JSON の例 :

```
"Competency" : [  
  {  
    "skillUsed" : {  
      "value" : 10,  
      "type" : "Years"  
    },  
    "auth" : true,  
    "skillLevel" : 18,  
    "skillName" : "Marketing Management",  
    "TaxonomyId" : {  
      "idOwner" : "DAXTRA",  
      "description" : "marketing > misc"  
    },  
    "description" : "Skill",  
    "skillAliasArray" : [  
      "Marketing Management",  
      "マーケティングマネジメントを 10 年"  
    ]  
  },  
]
```

この **Competency** 要素の詳細は“Competency 要素”の項で説明します。

CV パーサ: カスタマイズプロフィール

デフォルトスキルと顧客スキル

CV/履歴書の抽出が行われる際、DaXtra のパーサは DaXtra スキルタクソノミー(DST)と言語的文脈の両方を使用し、スキル、資格、職業名を見分けます。スキルが DST を通して抽出された場合、そのスキルは'認証済み'とマークされ、正式な表記にノーマライズされます。例えば"B2B"は"Business to Business"へ、"Exchange"は"Microsoft Exchange Server"へと統一されます。

デフォルトアウトプット

正規化された Competency 名

DaXtra のエンジンが生成するプロフィール内のスキルや資格は **Competency**(能力)要素として表記されます。もしこのような **Competency** が DaXtra スキルタクソノミー(DST)とマッチした場合、そのスキルは'認証済み'となり、**auth** 属性が XML では'yes' (Json では'true')にセットされ、**name** 属性(Json では **SkillName**)は正式な表記にノーマライズされます。

例えば履歴書内で"B2B"という文字列が発見されたとします。適切な文脈上ではこの文字列は'Business to Business'という名前で、**auth** 属性が'yes'にセットされた **Competency** ノードを構築します。このような処理が発生したのは、他のエイリアスと並行し'B2B'が DST 内の'Business to Business'というスキルの中に含まれているからです。

この CV 内で DaXtra のパーサがマッチした実際の文字列"B2B"はその後 **Competency** 要素内の **Alias** サブ要素に格納されます。正式表記の'Business to Business'は **Alias** サブノードの一番初め、および **Competency** ノードの **Name** 属性として置かれます。

この正式表記が **Alias** サブ要素の最初に保持されるのは **Competency** 要素の **Name** 属性は後で顧客の選ぶスキル名の形に再マッピングされる可能性があるからです。これについては**カスタムアウトプット**の項で説明します。

また **Name** 属性に入るノーマライズされた正式表記というのはデフォルトでは英語で、これは DaXtra のエンジンが多言語に対応しているためです。DST に登録された各スキルは英語・日本語・フランス語・ドイツ語など様々な言語の同意義の言葉をエイリアスとして持つため、代表して英語がそのスキル名になっています。この **Name** 属性に入るスキル名を日本語に変更したい場合にも上記の**カスタムアウトプット**の活用が有効です。

下記は"B2B"という文字列の例です。この文字列を DaXtra がスキルとして認証し、'Business to Business'にノーマライズしました。(関連のある項目のみの掲載)

XML 形式

```
<Competency oids='6 ' description='Skill' auth='yes' name='Business to Business'>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
```

```
<StringValue>Business to Business</StringValue>
</CompetencyEvidence>
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>B2B</StringValue>
</CompetencyEvidence>
```

JSON 形式

```
{
  "auth" : true,
  "skillName" : "Business to Business",
  "description" : "Skill",
  "skillAliasArray" : [
    "Business to Business",
    "B2B",
  ]
},
```

非正規化 Competency 名

前述のとおり、DaXtra のパーサは DST に頼らず、代わりに言語的文脈だけを利用しスキルを抽出することもできます。この場合には **Competency** ノードの **name** 属性は、テキスト内で実際にマッチした文字列になり、**auth** 属性は XLS では 'no' (Json では 'false') または省略されます。

カスタムアウトプット

顧客が独自のスキルや ID のリストにマッチを行いたい場合には、初めに行われるデフォルトのスキル抽出の後に、ユーザー独自スキルのマッピングを適用が可能です。

顧客は DaXtra の提供するツールを使い、独自のスキルタクソノミー(マスタースキルフファイルまたは MSF と呼びます)を開発、メンテナンスすることができます。また DaXtra が顧客のスキルリストから MSF を作成することもできます。

この MSF は顧客独自スキルリストを反映した Competency を含むカスタムプロフィールを生成するのに使用されます。このスキルリストにおいてはどのようなスキル名(多言語も可)や、マップしたい文字列(こちらも多言語可)を含めても構いません。

DaXtra のパーサがノーマライズされたスキルやされていないスキルを MSF 内に見つけた場合、下記の変更が **Competency** 要素に加えられます。

- **Competency** 要素内の **name** 属性は顧客の定めたスキル名に変更されます。(これが **Alias** サブノードの一行目にノーマライズされたスキル名を保存している理由です)
- **CompetencyId** サブ要素が顧客データベースのコードにマッチした **id** 属性とともに追加されます。(この ID を利用し顧客データベースへのスキルの自動格納が可能になります)

マスタースキルファイルのサンプル

以下はマスタースキルのサンプルの一部です:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<AllSkills>
  <entry>
    <category>skill</category>
    <type>Skill</type>
    <code>240</code>
    <descr>Biz2Biz</descr>
    <parent>MySkills</parent>
    <auth type="match">Business to Business</auth>
  </entry>
  <entry>
    <category>skill</category>
    <type>Skill</type>
    <code>123</code>
    <descr>デジタルマーケティング</descr>
    <parent>MySkills</parent>
    <alias locale="ja" cs="n" ambig="n">デジタル</alias>
  </entry>
</AllSkills>
```

このファイルにはコードとマッピングを含めた二つのスキルがあります。(Biz2Biz とデジタルマーケティング)

DaXtra の抽出したスキルからのマッピング(MSF 内 auth)

DaXtra の抽出したスキルを顧客の **code** や **descr** にマッピングするにはそのマッピング先を MSF 内で **auth** ノードを使い指定する必要があります。**auth** 要素とは“auth 名の追加されたスキルが認証され、MSF 内の **code** と **descr** に再マップされたかどうか”を表します。

例えば、顧客の MSF の中に‘Biz2Biz’という **descr(name)**、‘240’という **code** のスキルがあったとします(上記 MSF の例と同様)そして DaXtra のシステムが‘Business to Business’というスキルを発見した場合、MSF を使うことで以下のようなアウトプットが生成されます(関連のある項目のみの掲載)。

XML 形式:

```
<Competency oids='6 ' description='Skill' auth='yes' name='Biz2Biz'>
  <CompetencyId id='240' idOwner='me' />
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>Business to Business</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>B2B</StringValue>
  </CompetencyEvidence>
</Competency>
```

JSON 形式:

```
"Competency" : [
  {
    "auth" : true,
    "CompetencyId" : {
      "idOwner" : "me",
      "id" : "240"
    },
    "skillName" : "Biz2Biz",
    "description" : "Skill",
    "skillAliasArray" : [
      "Business to Business",
      "B2B"
    ]
  }
],
```

注目すべきは **Competency** 要素内で以前と異なる **name** 属性/**skillName** サブ要素('Business to Business' から 'Biz2Biz'へ)に変更され、**id** 属性/サブ要素(ここでは'240')と **idOwner** 属性/サブ要素(顧客のアカウント)が追加されたということです。元々の認証スキル('Business to Business')は **Alias** の初めに掲載されているので必要であればそこから復元することができます。

DaXtra が抽出しなかったスキルの識別(MSF 内 alias)

DST は広範囲にわたる業界を網羅しているものの、それでもまだ登録されていないスキルは存在します。MSF を利用することで、新しいスキルを追加する DaXtra システムの拡張が可能です。その場合、MSF に **entry** ノードを作成する必要があり、CV 内でどのようにこのスキルが記載されるかを **alias** 要素(複数可)によって設定します。MSF 内の **entry** は例えば以下ようになります:

```
<entry>
  <category>skill</category>
  <type>Skill</type>
  <code>123</code>
  <descr>デジタルマーケティング</descr>
  <parent>MySkills</parent>
  <alias locale="ja" cs="n" ambig="n">デジタル</alias>
</entry>
```

これは全ての“デジタル”という単語を‘デジタルマーケティング’にノーマライズし‘123’というユーザー **code** を割り当てるよう DaXtra のシステムに指示するという意味です。ただしこれは良い例ではないかもしれません。なぜなら“デジタル”という言葉は違う文脈で使われるかもしれないからです。 **alias** マッピングを登録する際には慎重になる必要があります。このようなケースをより特定の文脈に限定する補助として locale, cs, ambig という属性があります。

- locale – CV の言語を特定します。例えば“en”では英語の文脈でのみマッチし他の言語ではしません。
- cs – Case Sensitive の略で、“y”に設定されていれば大/小文字が一致した **alias** のみマッチします。
- ambig – Ambiguous (曖昧さ)の略でその単語が他の意味にもとれるかどうかを表します。これが“y”に設定されていると、その **alias** は他の既知のスキルが近接している場合のみに抽出します。

スキルグルーピング

MSF は個々にマッチした複数のスキルを一つの共同名(**descr**)へグルーピング(まとめあげ)するために使用することもできます。例えばもし顧客のアプリケーション内で複数の Microsoft の資格を‘Microsoft 資格’として記録したいとします。その場合 MSF 内でそのような資格を全て **auth** 要素として‘Microsoft 資格’という **entry** の中に並べます。また、例えば MCAF(Microsoft Certified Azure Fundamentals)など DaXtra が抽出しない特定の資格がある場合、それを **alias** 要素として同じ **entry** に並べることができます。MSF の例:

```
<entry>
  <category>qualification</category>
  <type>Degree/Qualification > Professional</type>
  <code>127</code>
  <descr>Microsoft Certified</descr>
  <parent>MySkills</parent>
  <auth type="match">Microsoft Official Trainer</auth>
  <auth type="match">Microsoft Certified Professional</auth>
  <auth type="match">Microsoft Certified Network Engineer</auth>
  <alias locale="en" cs="y" ambig="n">MCAF</alias>
</entry>
```

それではここで CV の中に“MCNE の資格を取得”というフレーズがあったとしましょう。

まず、“MCNE”は上記の通り DaXtra の DST により‘Microsoft Certified Network Engineer’として抽出されます。そして顧客の MSF 適用により‘Microsoft Certified Network Engineer’は‘Microsoft 資格’という **descr** に (XML プロフィールでは **name** 属性として記載)、**id** は‘127’として(プロフィール内では **CompetencyId** サブ要素の id 属性として)マップされます。この処理が生成するプロフィール内の **Competency** 要素は以下の通りです:

XML 形式:

```
<Competency oids="77 " description="Degree/Qualification > Professional" auth="yes"
name="Microsoft Certified">
  <CompetencyId id="127" idOwner="me"/>
  <CompetencyEvidence typeDescription="Alias" name="Alias">
    <StringValue>Microsoft Certified Network Engineer</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription="Alias" name="Alias">
    <StringValue>MCNE</StringValue>
  </CompetencyEvidence>
</Competency>
```

初めに DST にノーマライズされた **name** は **Alias** の一行目にあり、実際の一致したテキスト(ここでは“MCNE”)はその下に記載されています。

他の例も見てみましょう。“MCAF の資格も取得済みです”というフレーズが CV に出てきたとします。‘MCAF’は DST の中にはないので通常は抽出されませんが、しかし MSF の **alias** を利用することで、この場合認識・抽出が行われ、そして‘Microsoft 資格’にマッピングされます。抽出されたプロフィールの例:

XML 形式:

```
<Competency oids="92 " description="Degree/Qualification > Professional"
name="Microsoft Certified">
  <CompetencyId id="127" idOwner="me"/>
  <CompetencyEvidence typeDescription="Alias" name="Alias">
    <StringValue>MCAF</StringValue>
  </CompetencyEvidence>
</Competency>
```

スキル拡張

反対に今度は、MSF を利用することで一つのマッチから複数の **Competency** を生成することも可能です。

例えば“TOEFL をパスしました”という文があれば DaXtra のシステムは‘TOEFL’という資格を生成しますが、もしかすると顧客はさらに‘TOEFL 資格’と‘英語の知識’という二つのスキルを生成したいと思うかもしれません。これを実現するためには MSF 内に以下のような二つのエントリーを作る必要があります：

```
<entry>
  <category>qualification</category>
  <type>Degree/Qualification > Professional</type>
  <code>1123</code>
  <descr>TOEFL資格</descr>
  <parent>MySkills</parent>
  <auth type="match">TOEFL</auth>
</entry>
```

```
<entry>
  <category>qualification</category>
  <type>Degree/Qualification > Professional</type>
  <code>1124</code>
  <descr>英語の知識</descr>
  <parent>MySkills</parent>
  <auth type="match">TOEFL</auth>
</entry>
```

これにより DaXtra が‘TOEFL’の資格を認識した時、顧客は **Competency** 要素をプロフィール内に二つ取得します。

XML 形式:

```
<Competency oids="96 " description="Degree/Qualification > Professional" auth="yes"
name="TOEFL 資格">
  <CompetencyId id="1123" idOwner="me"/>
  <CompetencyEvidence typeDescription="Alias" name="Alias">
    <StringValue>TOEFL</StringValue>
  </CompetencyEvidence>
</Competency>

<Competency oids="96 " description="Degree/Qualification > Professional" auth="yes"
name="英語の知識">
  <CompetencyId id="1124" idOwner="me"/>
  <CompetencyEvidence typeDescription="Alias" name="Alias">
    <StringValue>TOEFL</StringValue>
  </CompetencyEvidence>
</Competency>
```

Competency 要素

プロフィール内の **Competency**(能力・特性)要素は、候補者の持つスキル(Skill)、候補者が取得済み学位と資格(Degree/Qualification)、候補者が雇用されていた間についていたポジション(Held Position)、の3要素のうちどれかを表します。**Competency** 要素の構成は以下です：

```
<Competency description='Skill > IT' auth='yes' name='Python Programming'>
```

初めに、**Competency** 要素は **description**(説明)属性を持ち、その値は以下のうちのいずれかになります：

- Skill (スキル)
- Skill > IT (スキル>IT)
- Skill > Language (スキル>語学)
- Degree/Qualification (学位/資格)
- Degree/Qualification > Professional (学位/資格 > 会員)
- Held Position (職位)
- Industry (業界)
- Location (ロケーション…※スキルプロジェクトのみ)
- Organization (組織…※スキルプロジェクトのみ)

つまり DaXtra のパーサによって判断されたスキルの分類ということです。(※**カスタムアウトプット**の項よりスキルプロジェクト: DaXtra は顧客独自のスキルリストからマスタースキルファイルを作成します。)

name(名前)属性には履歴書上に見つかりマッチしたスキルの名前が入ります。この **name** は動的抽出、DaXtra 内蔵辞書、顧客のマスタースキルファイル(MSF)の三通りのパターンから選ばれます。動的抽出の場合では **name** は履歴書内で見つかった文字列通りとなりますが、DaXtra 内蔵辞書と顧客のスキルファイルでは **name** は正規化(ノーマライズ)されます。DaXtra 内蔵辞書の場合の正規化は DaXtra のシステムに格納されている正式な表記(英語)、顧客のスキルファイルの場合ではこの **name** 属性は MSF から取得されます。

auth(authorisation の略、認証)属性は boolean 型の値でそのスキルが DaXtra 内蔵辞書に見つかったかどうかを表します。DaXtra は一般的なスキルを網羅した辞書を保持しており、このような各“認証された”スキルは、正規化された名前と、DaXtra が類義語であると判断する様々なエイリアス(別名)を持ちます。このエイリアスは実際に履歴書へのマッチに使われる文字列です。また、認証されたスキルの多くは多言語のエイリアスを含んでいます。

Competency 要素 : CompetencyId サブ要素

Competency 要素は、例えばデータベースのコードなど、ある特定の分類スキマの内部 ID に結び付が可能で
す。この ID は **CompetencyId** サブ要素として表記されます。例えば 'XXCareer' という会社の分類スキマの
中で 'Python' に 20789 という ID が振られているとすると、XML ファイル内では以下のように表記されます :

```
<CompetencyId id='20789' idOwner='XXCareer'/>
```

id 属性は string 型の値(スペースはアンダーライン 2 つで表現)で、顧客のマスタースキルファイル内で定義さ
れている結び付けた内部 ID を示します。

idOwner 属性は string 型の値(スペース不可)で履歴書の処理に使用されたアカウントを表します。

Competency 要素 : TaxonomyId サブ要素

Competency 要素内の **TaxonomyId** サブ要素はタクソノミーによるクラス分けです。タクソノミー **id** 属性
は業種/業界を表します。これは会社の業種というよりも候補者の持つ職位、スキル、資格に関連しており、つ
まり IT 開発者が銀行で働いていた場合には 'banking-finance' (銀行・金融) ではなく 'it' (情報) になります。

```
<TaxonomyId id='it~progr-lang' idOwner='DAXTRA' description='it > progr-lang'/>
```

id 属性は string 型の値(スペースなし)でタクソノミー名を示します。時折、-mb (maybe の略)などのタグが
語尾についていることがありますが、これはそのタクソノミーへの関連が確実でないという意味です。

idOwner 属性は string 型の値(スペースなし)で、タクソノミーは内蔵されているためここでのアカウントは顧
客のものではなく 'DAXTRA' と表示されます。

description 属性は string 型の値でタクソノミーの名前を示します。

Competency[description=Skill*] における値は以下です :

accountancy	construction	engineering
aerospace	customer-service	engineering~manufacturing
agriculture	edu	engineering~mechanical
automotive	edu~higher	fashion-art
banking-finance	edu~preschool	food-drink
caretaking-cleaning	edu~school	health-safety
catering	electronics	health-safety~recycling

hr	medicine~nursing	security
hr~recruitment	medicine~pharmaceutical	social-services
insurance	mngm-consult	social-services~charity
it	oil-gas	social-services~forces
it~ccplusplus	procurement	social-services~public-sector
it~dotnet	property	social-services~youthwork
it~java	railway	sports
legal	retail	telecomms
marine	sales	transport
marketing	science	travel
media	science~library	
medicine	secretarial	



“>”を使用した階層に分かれる場合もあります。例：

```
it > progr-lang
medicine > infirmary > surgery
```

この“>”以後の部分はより詳細なエリアや業界内でのスキルのタイプを表す、業界特有のものです。

Competency[description=Degree/Qualification] または **Competency[description=Degree/Qualification > Professional]** においては他に以下の値が有効です：

high school or equivalent certification	intermediategraduate bachelors	HND/HNC or equivalent secondary
associates	masters	professional
vocational	doctorate	
some college	certification	

Competency 要素：CompetencyEvidence サブ要素

Competency 要素内の **CompetencyEvidence** サブ要素は以下の 3 種類です：

```
CompetencyEvidence[typeDescription=Extent of Experience]
CompetencyEvidence[typeDescription=Application]
CompetencyEvidence[typeDescription=Alias]
```

CompetencyEvidence[typeDescription=Extent of Experience] は職位やスキルの経験の期間の長さを表します。履歴書内に明記されている(例：“Java を三年経験”) か、そのスキルが特定の職歴に関連づけられている必要があります。

出力例：

```
<CompetencyEvidence name='Extent of Experience' typeDescription='Extent of Experience'
type='YEAR' lastUsed='2019' >
  <NumericValue description='YEAR'>3</NumericValue>
</CompetencyEvidence>
```

```
<CompetencyEvidence name='Extent of Experience' typeDescription='Extent of Experience'
type='MONTH' lastUsed='2015'>
  <NumericValue description='MONTH'>58</NumericValue>
</CompetencyEvidence>
```

type 属性は時間の単位を表します。有効な値は YEAR、MONTH、DAY です。

lastUsed 属性は 4 桁の数字からなる西暦年で、スキルが最後に使われた日付を表し、職歴内の職位(‘Held Position’)とスキル(‘Skill, Skill’ > ‘IT’)においてのみ有効です。例えば、スキルが“2010 年から 2015 年まで”のひとつの職歴内から抽出された場合、この **lastUsed** の年は‘2015’となります。もし最終年が履歴書内で“現在”と書かれている場合には実際の現在の年が適用されます。

CompetencyEvidence[typeDescription=Application] はテキスト内に明記されている場合のそのスキルの応用先です。(例：“Oracle 8i – 設計と管理”)

出力例：

```
<CompetencyEvidence name='Application' typeDescription='Application' type='design'>
  <StringValue>設計</StringValue>
</CompetencyEvidence>
```

```
<CompetencyEvidence name='Application' typeDescription='Application' type='administration'>
  <StringValue>管理</StringValue>
</CompetencyEvidence>
```

CompetencyEvidence[typeDescription=Alias] は履歴書内で見つかり、スキルとしてマッチした文字列を表します。この欄の値はスキル名と同じ言葉になる場合もあれば違う場合もあります。(例：スキル名が‘ユーザーインターフェースデザイン’で、マッチの元となるエイリアスが‘UI デザイン’) このようになる理由は以下です(複数該当する場合があります)：

- a) **Competency** の **name** 属性が顧客の MSF から与えられた。
- b) **Competency** の **name** 属性が DaXtra の内臓辞書により正規化された。
- c) パーサの分析の結果 **Competency** の **name** 属性が変更された。

上記のうちのどれかが適用された場合、スキル名が MSF のものに変更される前に、**Alias** の一つ目がスキル名にセットされます。(ノーマライズの是非関わらず)そして残りのエイリアスはその下に併記されます。

エイリアス出力例 :

```
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>Python Programming</StringValue>
</CompetencyEvidence>
```

```
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>Python</StringValue>
</CompetencyEvidence>
```

出力例 2 (多言語エイリアスのノーマライズの例) :

```
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>French</StringValue>
</CompetencyEvidence>
```

```
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>Français</StringValue>
</CompetencyEvidence>
```

```
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>フランス語</StringValue>
</CompetencyEvidence>
```

```
<CompetencyEvidence typeDescription='Alias' name='Alias'>
  <StringValue>仏語</StringValue>
</CompetencyEvidence>
```

Competency 要素 : CompetencyWeight サブ要素

CompetencyWeight サブ要素は Competency 要素と他の Competency 要素と比べたスコア付けです。これはスキル、学位、資格、職位の重要性を表すので、それに応じたソートを可能とします。このウェイト(重要度)は同種類の **Competency** においてのみ適用されます。(例 : スキル同士(Competency[type=Skill]), 学位・

資格同士(Competency [type=Degree/Qualification])、職位同士(Competency[type=Held Position])での比較ができます)。Competency 要素は以下 3 種類の CompetencyWeight サブ要素を取りえます：

CompetencyWeight[type=skillLevel]

CompetencyWeight[type=skillProficiency]

CompetencyWeight[type=skillCount]

CompetencyWeight[type=skillLevel] はウェイトとしてスキルに与えられた数値です。この値は 0-100 の整数で、近似性、頻出性、スキルを利用した職歴の長さを含んだアルゴリズムによって計算されます。これは絶対値ではないので、他の履歴書へまたがる比較ではなく、同履歴書内での他のスキルとの比較としての利用に適しています。顧客によってはこの値を各候補者の上位スキルを選ぶのに役立てています(例：ウェイトの大きい順に上から 5 個)。

出力例：

```
<CompetencyWeight type='skillLevel'>
  <NumericValue minValue='0' maxValue='100'>99</NumericValue>
</CompetencyWeight>
```

NumericValue は int 型で有効な値は 0-100 です。

CompetencyWeight[type=skillProficiency]は簡素化されたスキルの洗練度のウェイトです。

この機能は以下のように記されます：

```
<CompetencyWeight type='skillProficiency'>
  <StringValue minValue='BASIC' maxValue='EXCELLENT'>EXCELLENT</StringValue>
</CompetencyWeight>
```

StringValue は string 型で有効な値は BASIC、WORKING、GOOD、EXCELLENT です。この値は **skillLevel** から算出されます。

Competency[description=Skill > Language](語学能力)の場合ではでは BASIC、INTERMEDIATE、ADVANCED、EXCELLENT、NATIVE が値となります。例：

```
<CompetencyWeight type='skillProficiency'>
  <StringValue minValue='BASIC' maxValue='EXCELLENT'>NATIVE</StringValue>
</CompetencyWeight>
```

Competency[description=Degree/Qualification] または **Competency[description=Degree/Qualification > Professional]** では以下の値を取りえます：

学士:

First_Class
Second_Class_Upper
Second_Class_Lower
Third_Class
Honours

修士:

Pass
Merit
Distinction

もしくは履歴書から抽出されたフリーテキスト 例:

```
<CompetencyWeight type='skillProficiency'>  
  <StringValue>GPA: 3.4</StringValue>  
</CompetencyWeight>
```

Competency[description=Held Position]では職歴への関連性が推定されます。

現在あるいは過去 1 年以内の職位 - EXCELLENT
過去 3 年以内 - GOOD
過去 5 年以内 - WORKING
その他 - BASIC

CompetencyWeight[type=skillCount] は履歴書内での出現回数です。

Competency[description=Industry]専用で、特定の業界と認識された職歴がいくつあるかを反映します。

特殊な Competency の例

“A-Level” Competency 機能

(イギリスの教育課程。日本語での実装は 2019 年 3 月現在されていません)

A Levels は科目と評価が一緒になっている場合の Competency です。

A,B,C,D は CompetencyWeight 要素で EXCELLENT, GOOD, INTEMEDIATE, BASIC として表されます。

つまり、“A Levels : Geography (A)”は以下の通りに示されます:

```
<Competency description='Degree/Qualification' auth='yes' name='A Levels/Grades > A Levels
> A Levels Geography'>
  <TaxonomyId id='high school or equivalent' idOwner='DAXTRA' description='high school or
  equivalent'/>
  <CompetencyEvidence name='Application' typeDescription='Application' type='Geography'>
    <StringValue>Geography</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>A Levels/Grades > A Levels > A Levels Geography</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>A Levels : Geography (A)</StringValue>
  </CompetencyEvidence>
  <CompetencyWeight type='skillLevel'>
    <NumericValue minValue='0' maxValue='100'>83</NumericValue>
  </CompetencyWeight>
  <CompetencyWeight type='skillProficiency'>
    <StringValue minValue='BASIC' maxValue='EXCELLENT'>EXCELLENT</StringValue>
  </CompetencyWeight>
</Competency>
```

自動車免許

Competency[description='Degree/Qualification > Professional'] として以下のように示されます:

```
<Competency description='Degree/Qualification > Professional' auth='yes' name='Unten Menkyo
Class 1 Futsu'>
  <TaxonomyId id='certification' idOwner='DAXTRA' description='certification'/>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>Unten Menkyo Class 1 Futsu</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>普通自動車免許</StringValue>
  </CompetencyEvidence>
  <CompetencyWeight type='skillLevel'>
    <NumericValue minValue='0' maxValue='100'>82</NumericValue>
  </CompetencyWeight>
</Competency>
```

セキュリティークリアランス

Competency[description='Degree/Qualification > Professional']として以下のように示されます:

```
<Competency description='Degree/Qualification > Professional' auth='yes' name='Security Clearance
> Security Clearance BC'>
  <TaxonomyId id='certification' idOwner='DAXTRA' description='certification'/>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>Security Clearance > Security Clearance BC</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>BC Security Clearance</StringValue>
  </CompetencyEvidence>
  <CompetencyWeight type='skillLevel'>
    <NumericValue minValue='0' maxValue='100'>82</NumericValue>
  </CompetencyWeight>
</Competency>
```

セキュリティークリアランスのレベルは以下です:

Security Clearance	Security Clearance POLY CI
Security Non Clearance	Security Clearance PUBLIC TRUST
Security Clearance CONFIDENTIAL	Security Clearance BC
Security Clearance TSSI	Security Clearance SC
Security Clearance SECRET	Security Clearance DV
Security Clearance TOP SECRET	Security Clearance CTC
Security Clearance SCI	Security Clearance CRB
Security Clearance SSBI	Security Clearance DBS
Security Clearance SBI	Security Clearance NATO
Security Clearance POLY FS	Security Clearance EU
Security Clearance POLY LS	

資格からの語学能力抽出

語学能力は言語に関する資格から抽出される場合もあります。例:

TOEIC 880 => English Excellent

```
<Competency description='Skill > Language' auth='yes' name='English'>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>English</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>TOEIC 880</StringValue>
  </CompetencyEvidence>
  <CompetencyWeight type='skillLevel'>
    <NumericValue minValue='0' maxValue='100'>84</NumericValue>
  </CompetencyWeight>
  <CompetencyWeight type='skillProficiency'>
    <StringValue minValue='BASIC' maxValue='EXCELLENT'>EXCELLENT</StringValue>
  </CompetencyWeight>
</Competency>
```

IELTS 6.5 => English Intermediate

```
<Competency description='Skill > Language' auth='yes' name='English'>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>English</StringValue>
  </CompetencyEvidence>
  <CompetencyEvidence typeDescription='Alias' name='Alias'>
    <StringValue>IELTS 6.5</StringValue>
  </CompetencyEvidence>
  <CompetencyWeight type='skillLevel'>
    <NumericValue minValue='0' maxValue='100'>48</NumericValue>
  </CompetencyWeight>
  <CompetencyWeight type='skillProficiency'>
    <StringValue minValue='BASIC'
maxValue='EXCELLENT'>INTERMEDIATE</StringValue>
  </CompetencyWeight>
</Competency>
```

付録 A - XML/JSON 要約



XML					JSON									
XML 要素	XML 属性/サブ要素	XML 属性/サブ要素	XML 属性	値の例	内容の例	JSON オブジェクト	JSON オブジェクト	JSON オブジェクト	値の例					
Competency	oids			22_1 30_4_1		Competency								
	description			Held Position			description			Held Position				
	auth			no			auth			false				
	name			Registered Nurse			skillName			Registered Nurse				
	CompetencyId	idOwner					me	CompetencyId	idOwner			me		
			id				JOB:123			id			JOB:123	
	TaxonomyId	id					medicine	TaxonomyId	idOwner				DAXTRA	
			idOwner				DAXTRA			description			medicine	
							medicine							
	CompetencyEvidence	name					Extent of Experience	SkillUsed						
		typeDescription					Extent of Experience		value			92		
		type					MONTH			type			Months	
		lastUsed					2013							
		NumericValue	description				MONTH				2013			
	CompetencyEvidence	typeDescription					Application	skillAliasArray						
		name					Application							
		type					Health							
		StringValue					Health							
	CompetencyEvidence	typeDescription					Alias	Health						
		name					Alias							
StringValue				Registered Nurse										
CompetencyWeight	type				skillLevel	skillLevel								
	NumericValue	minValue			0					39				
		maxValue			100									
CompetencyWeight	type				skillProficiency	skillProficiency								
	StringValue	minValue			BASIC					BASIC				
		maxValue			EXCELLENT									

付録 B: 用語解説

XML	JSON	説明
Competency	Competency	スキル・資格・職位を含む要素/オブジェクト
oids		Competency ノードを CV 内の情報に結び付ける連続した番号
description (Competency)	description	Competency の種類
auth	auth	DaXtra の内臓スキル一覧に該当するかどうか
name (Competency)	skillName	ノーマライズされた抽出スキルの名前
CompetencyId	CompetencyId	カスタム ID の情報が入るサブ要素/オブジェクト
idOwner (CompetencyId)	idOwner (CompetencyId)	その CV のパーキングに使われたアカウント名
id (CompetencyId)	Id	スキルを顧客のデータベースに結び付けるカスタム ID
TaxonomyId	TaxonomyId	
id (TaxonomyId)		タクソノミーの名前
idOwner (TaxonomyId)	idOwner (TaxonomyId)	タクソノミーの所有者アカウントの名前(常に DAXTRA)
description (TaxonomyId)	description (TaxonomyId)	タクソノミーの名前
CompetencyEvidence (for name='Extent of Experience')	SkillUsed	経験の長さの情報が入るサブ要素/オブジェクト
name='Extent of Experience'		上記参照
typeDescription='Extent of Experience'		上記参照
type (CompetencyEvidence, for name='Extent of Experience')	type	時間の長さの種類、例:月や年
lastUsed		スキルや職位が最後に使われた日
NumericValue (CompetencyEvidence, for name='Extent of Experience')	value	スキルが使われた合計の時間の長さ
	lastUsed	スキルや職位が最後に使われた日
CompetencyEvidence (for name='Alias')		CV 内に見つかったこのスキルにマッチする文字列の詳細を含むサブ要素
name='Alias'		上記参照
typeDescription='Alias'		上記参照
StringValue (CompetencyEvidence, for name='Alias')	skillAliasArray	CV 内に見つかったこのスキルにマッチする文字列の詳細を含むサブ要素/オブジェクト
CompetencyEvidence (for name='Application')		スキルの応用先を説明するサブ要素

<code>name='Application'</code>		上記参照
<code>typeDescription='Application'</code>		上記参照
<code>type (CompetencyEvidence, for name='Application')</code>		このスキルの応用先の種類
<code>StringValue (CompetencyEvidence, for name='Application')</code>	<code>skillAliasArray</code>	このスキルの応用先の種類(JSON の場合値は skillAliasArray)
<code>CompetencyWeight (for type='skillLevel')</code>		スキルの熟練度アルゴリズムにより割り当てられたウェイト数値を表すサブ要素
<code>type='skillLevel'</code>		上記参照
<code>NumericValue (CompetencyWeight)</code>	<code>skillLevel</code>	熟練度アルゴリズムにより割り当てられたこのスキルのウェイト
<code>CompetencyWeight (for type='skillProficiency')</code>		ノーマライズされたウェイト数値や自由な文字列を表すサブ要素
<code>type='skillProficiency'</code>		上記参照
<code>StringValue (CompetencyWeight)</code>	<code>skillProficiency</code>	アルゴリズムによって割り当てられたこのスキルの熟練度のウェイト