

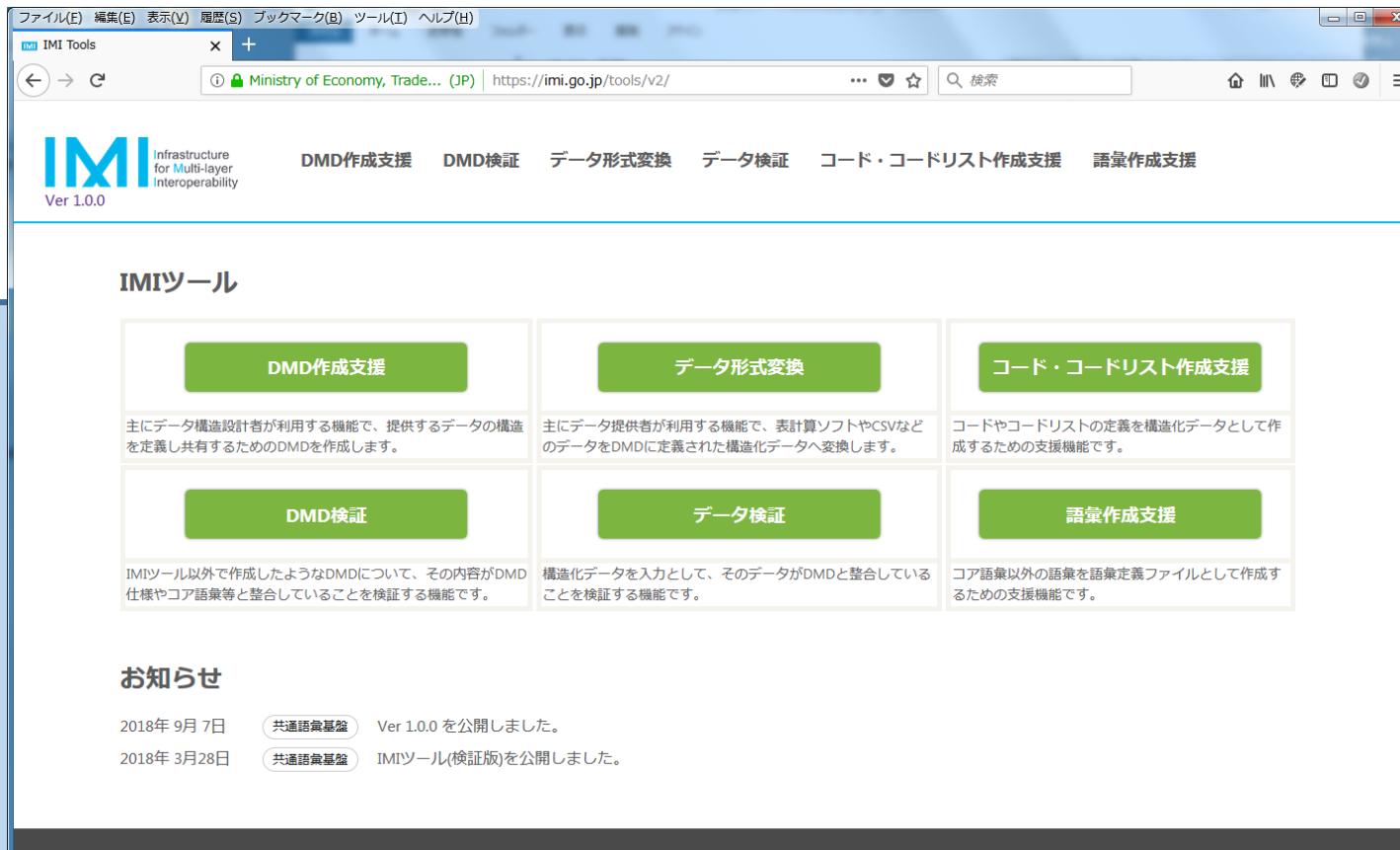
IMI共通語彙基盤ライブラリのご紹介

IPA 斎藤 浩 / IPA 豊田 耕司

2018年11月13日(火)

独立行政法人 情報処理推進機構
社会基盤センター産業プラットフォーム部
データ活用推進グループ

IMI共通語彙基盤ライブラリとIMIツールの関係



IMI共通語彙基盤ライブラリ

IMI共通語彙基盤ライブラリ バージョン1.0.0

IMI共通語彙基盤ライブラリは、データ入力ツールやデータを利用するアプリケーション、CMS等で共通語彙基盤対応の機能を組み込む際に利用して、構造化データを容易に扱えるようにするためのライブラリです。



本ライブラリは、ワーキングドラフト版のIMI技術仕様に基づいて作成しています。

[ワーキングドラフト版について]
コメント及び空白文字の扱いが明示していなかったため、現仕様に準拠したもつとも寛容なパーサを実装しています。
他に明示されていない解釈においても同様な観点で実装しているため、正式版のIMI技術仕様と解釈に相違がある場合があります。

語彙の利用モデル

語彙(コア語彙、ドメイン語彙)の定義は、XMLのスキーマおよびRDFのスキーマとして提供され、実際のデータ交換に用います。語彙に基づいたデータ項目とその構造は、それらの語彙スキーマを参照してXMLとRDFそれぞれに定義することができます。

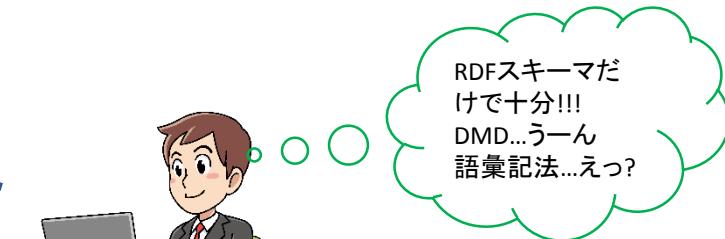
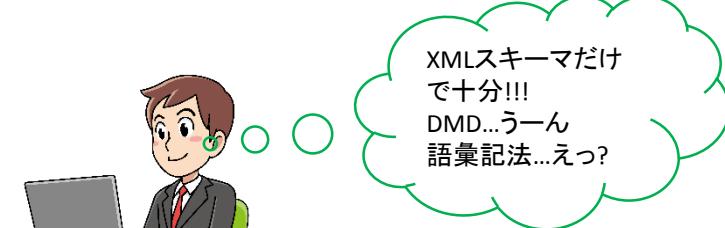
データ構造定義、各項目の値の範囲、書式、使用可能な文字セットの指定などの情報(データモデル)は、それら各定義ファイルへの参照を含め、Data Model Description (DMD)でまとめて記述します。それらを、語彙を用いたデータの利用者間でDMDを共有することによって、語彙に基づくデータ交換が実現されます。



XML Application 開発者



RDF Application 開発者



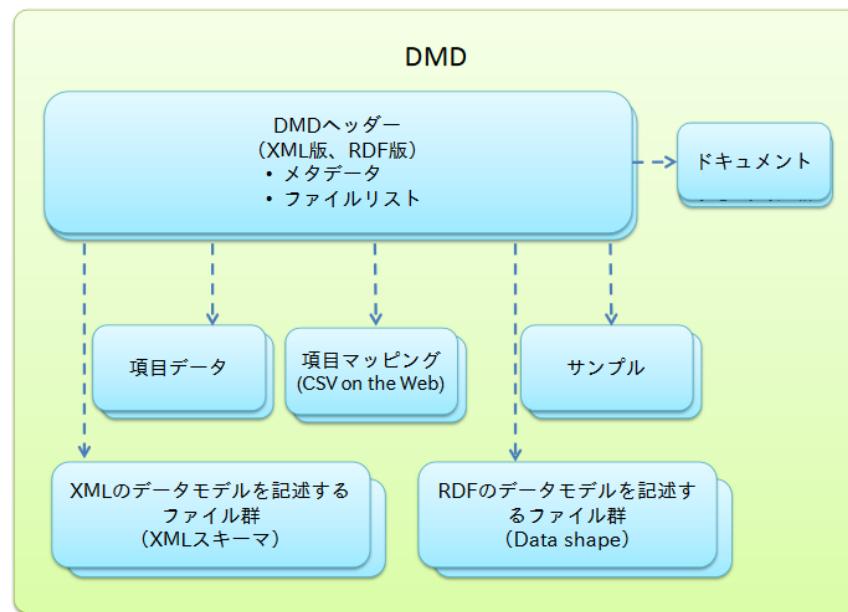
語彙を特定の利用形式に依存しない**中立的な形式**で記述することで、アプリケーションソフトウェア、データ作成者、アプリケーション開発者(データ利用者)の全てがそれぞれ扱いやすい形式でデータを扱うことができます。

DMD(データモデル記述)仕様のバージョンについて

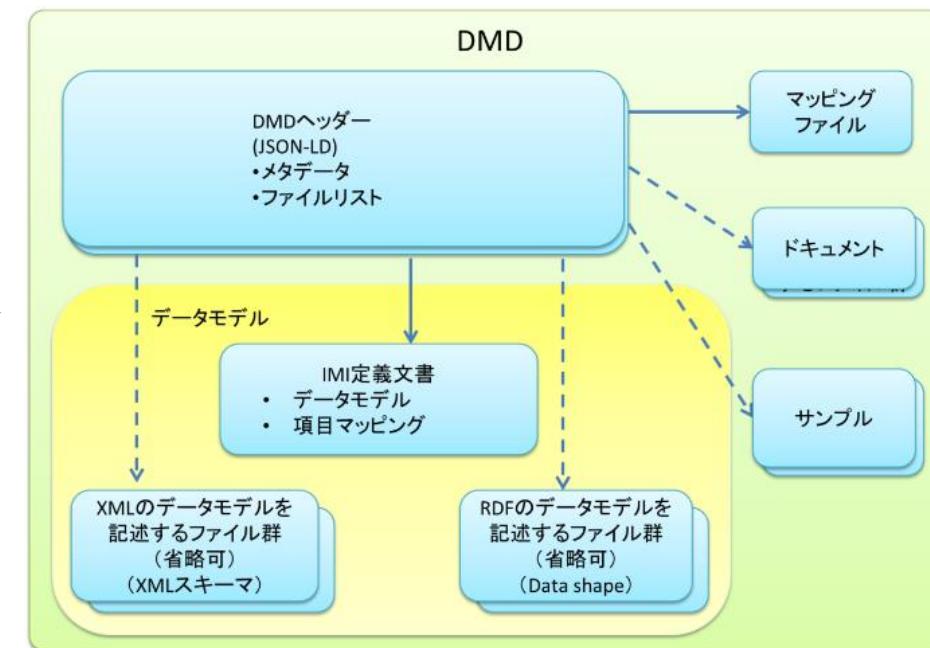
DMD2.0,2.1では、データモデル定義として XML, RDFなどで機械言語による判別できるように定義を持たせた仕様としています。

DMD3.0では、データモデル定義をIMI語彙記法に統一しました。また、マッピングを記述する技術仕様も併せて策定しました。

DMD2.0,2.1の構造



DMD3.0の構造



IMI共通語彙基盤ライブラリとIMI技術仕様の関係

IMI共通語彙基盤ライブラリは、適用業務として実装するアプリケーションに組み込み利用することで、IMI共通語彙基盤に対応した構造化データを容易に扱うことができます。

DMD(データモデル記述)仕様

IMI語彙記法

IMI共通語彙基盤ではRDF、XML等、**多様な形式・フォーマットで利用可能な語彙**を提供することを目的としている。このため、語彙自体は特定の利用形式に依存しない中立的な形式で記述する必要がある。IMI語彙記法は、これを実現するための記法である。IMI語彙記法では語彙の中の用語の定義を行う。IMI語彙記法で定義された語彙はRDFやXMLなど必要な形式に変換することができる。

マッピングファイル仕様

マッピングファイルは、DMD(「DMD仕様書」参照)が定めるデータモデルにおいて値が入る項目と、その代表的なラベルを1対1に結び付けるファイルであり、ユーザーへの表示や、データ入力用のCSVファイルを生成する際、及び、そのCSVファイルから**構造化データへの簡易的な変換**などに利用される。

構造化項目名記法

使用文字規則の記法

語彙を作り上げるためには、クラス用語とプロパティ用語およびそれらの関連を定義する必要があります。

クラス用語

事柄(事物や出来事)に関する概念です。
性質や属性を表すプロパティ用語と結びついて
使っています。

プロパティ用語

事柄の性質や事柄と事柄の関係を指示する概念
を表すための用語です。

※コア語彙は、この関連で定義し提供しています。

人型

...ic:氏名
...ic:生年月日

日付型

...ic:年
...ic:月
...ic:日

生年月日

型: ic:日付型

説明: 生年月日を記述するためのプロパティ用語
定義域: 人型

日

型: xsd:integer

説明: 日付や日時などの日の値を記述するためのプロパティ用語
定義域: 日時型 日付型

クラス定義文(class)

クラス定義文は、語彙の定義に用いられ、IMI定義文書内のいずれかの語彙に含まれるクラス用語を定義します。

プロパティ定義文(property)

プロパティ定義文では、制約として、型制約、パターン制約、値等価制約、値開下限制約、値下限制約、値開上限制約、値上限制約、使用可能文字制約が使用できます。

プロパティ設定文(set)

プロパティ設定文は、語彙の定義に用いられ、« クラス用語名 » で指定されたクラス用語のプロパティ用語集合に« プロパティ用語名 » で指定されたプロパティ用語を設定します。

用語使用宣言文(use)

用語使用宣言文では、制約として、パターン制約、回数制約、型制約、値等価制約、値開下限制約、値下限制約、値開上限制約、値上限制約、使用可能文字制約が使用できます。

マッピングファイル仕様

マッピングファイルは、DMD（「DMD仕様書」参照）が定めるデータモデルにおいて値が入る項目と、その代表的なラベルを1対1に結び付けるファイルであり、ユーザーへの表示や、データ入力用のCSVファイルを生成する際、及び、そのCSVファイルから構造化データへの簡易的な変換などに利用されます。

Mapping

項目ラベルと構造化項目名の対応の一覧であり、値が文字列又はその配列である1個以上のプロパティをもつ。各プロパティの名前はCSV等のヘッダーラベルであり、値はそのラベルが対応付けられる構造化項目名で表現します。



構造化項目名は、クラス用語>プロパティ用語>プロパティ用語>…のように「>」で区切って一つのクラス用語と任意の数のプロパティ用語を並べたものです。



構造化項目名記法

構造化項目名の記法は、IMI語彙などの階層構造をもつデータ表現の位置を文字列によって表現するための仕様です。

モデルとマッピングとデータの関係

Datamodel X

```
#prefix ic: "http://imi.go.jp/ns/core/2#"
#prefix ex: "http://example.org/"
vocab "http://example.org/" ;

#name "ニックネーム型"
class ex:ニックネーム型 {@ic:名称型};
#name "オレオレ名型"
class ex:オレオレ名型 {@xsd:string};
#name "オレオレ名"
property ex:オレオレ名 {@ex:オレオレ名型};
#name "オレオレ"
set ex:ニックネーム型 > ex:オレオレ名;

datamodel;
use ic:施設型>ic:名称>ic:表記;
use ic:施設型>ic:名称>ic:カナ表記;
use ic:施設型>ic:住所>ic:表記;
use ic:施設型>ic:地理座標>ic:緯度;
use ic:施設型>ic:地理座標>ic:経度;
use ic:施設型>ic:地理座標>ic:測地高度>ic:数値;
use ic:施設型>ic:連絡先>ic:電話番号;
use ic:施設型>ic:連絡先>ic:内線番号;
use ic:施設型>ic:関与>ic:役割[='設置主体'];
use ic:施設型>ic:関与>ic:関与者{@ic:組織型}>ic:ID>ic:識別値;
use ic:施設型>ic:記述>ic:種別[='想定収容人数'];
use ic:施設型>ic:記述>ic:説明;
use ic:施設型>ic:関与>ic:関与者{@ic:組織型}>ic:名称>ic:表記;
use ic:施設型>ic:参照>ic:参照先;
use ic:施設型>ic:備考;
```

DMD

Mapping A

```
"名称": "ic:施設型>ic:名称>ic:表記",
"名称_カナ": "ic:施設型>ic:名称>ic:カナ表記",
"住所": "ic:施設型>ic:住所>ic:表記",
"想定収容人数": "ic:施設型>ic:記述{種別='想定収容人数'}>ic:説明",
"連絡先": "ic:施設型>ic:連絡先>ic:電話番号",
"URL": "ic:施設型>ic:参照>ic:参照先",
"説明": "ic:施設型>ic:記述>ic:説明",
"備考": "ic:施設型>ic:備考"
```

Mapping B

```
"名称": "ic:施設型>ic:名称>ic:表記",
"名称_カナ": "ic:施設型>ic:名称>ic:カナ表記",
"住所": "ic:施設型>ic:住所>ic:表記",
"緯度": "ic:施設型>ic:地理座標>ic:緯度",
"経度": "ic:施設型>ic:地理座標>ic:経度",
"標高": "ic:施設型>ic:地理座標>ic:測地高度>ic:数値",
"電話番号": "ic:施設型>ic:連絡先>ic:電話番号",
"内線番号": "ic:施設型>ic:連絡先>ic:内線番号",
"災害種別_洪水": "ic:施設型>ic:種別コード[災害種別_洪水]>ic:識別値",
"備考": "ic:施設型>ic:備考"
```

公開データ N1

名称	住所	想定収容人数	URL	連絡先	備考

公開データ N2

名称	住所	URL	電話番号	備考
				連絡先

公開データ N3

名称	名称_カナ	住所	緯度	経度	備考

公開データ N4

名称	住所	災害種別_洪水	連絡先	備考
				電話番号

データ形式変換(Converter)

Turtle, RDF/XML, XML

既存データの項目を施設Modelのデータ定義に割り当てることも可能。 Mapping作成者が検討

IMI共通語彙基盤ライブラリの構成

- ・IMILib2017Common
汎用スタティック関数群を収録しています。
- ・IMILib2017Common.Vocabulary
語彙を検証保持するためのクラスを定義しています。
- ・IMILib2017Common.Datamodel
データモデルを検証保持するためのクラスを定義しています。
- ・IMILib2017Common.DMD
DMDを検証保持するためのクラスを定義しています。
- ・IMILib2017Common.Mapping
マッピングを検証保持するためのクラスを定義しています。
- ・IMILib2017Common.Converter
マッピングを用いて変換を行うためのクラスを定義しています。
- ・IMILib2017Common.SC
各種バリデーションの際に発行されるエラーコードを収録しています。
- ・IMILib2017Common.Util
JSON-LD とその他(Turtle,RDF/XML,XML)の相互変換のための関数群を収録しています。

IMI共通語彙基盤ライブラリの構築

lib/charsets.js
lib/converter.js
lib/datamodel.js
lib/dmd.js
lib/mapping.js
lib/sc.js
lib/util.js
lib/vocabulary.js

lib/kparser.js
lib/vparser.js

parser generator

pegjs -o lib/kparser.js spec/kparser.pegjs 【構造化項目名】用
pegjs -o lib/vparser.js spec/vparser.pegjs 【IMI 定義文書】用

{

Webpack(with babel)



npm run build

convert ES6(2015+) code into a backwards compatible version of
JavaScript in current and older browsers or environments.

imi-lib-2017-common.js

実装で利用するライブラリ

IMI共通語彙基盤ライブラリの機能確認テスト (testツール)

imi-lib-2017-common.js

test/test.html
(Mocha/Chai)



個々の機能の確認テストを行っています。

※注:testツールは、ES6で書かれているため
ES6を解釈できないブラウザでは動作しません。

IMILib2017Common#parse_k

構造化項目名のパース機能

- ✓ 正常なクラス定義をエラーなくパースできること
- ✓ 異常なクラス定義(数字はじめの名称)が文法エラーを投げること
- ✓ 先頭の空白が無視されること
- ✓ 末尾の空白が無視されること
- ✓ 中間の空白が無視されること
- ✓ 散在する空白が無視されること
- ✓ 深さ1の構造化項目名をパースできること
- ✓ 深さ2の構造化項目名をパースできること
- ✓ 深さ3の構造化項目名をパースできること
- ✓ ルートクラスへのプライマリーキー指定をパースできること
- ✓ プロパティへのプライマリーキー指定をパースできること
- ✓ 末端のプロパティへのグループ指定をパースできること
- ✓ 中間のプロパティへのグループ指定をパースできること
- ✓ クラスへのグループ指定をパースできること
- ✓ 末端のプロパティへの言語タグをパースできること
- ✓ 中間のプロパティに付与された言語タグが文法エラーとなること
- ✓ クラスに付与された言語タグが文法エラーとなること
- ✓ 言語タグとグループを併用しても正しくパースできること
- ✓ プロパティに対するUR指定子が正しくパースできること
- ✓ クラスに対するUR指定子が正しくパースできること
- ✓ UR指定子とグループの併用が正しくパースできること
- ✓ 末端のプロパティへの固定値制約の単独指定が正しくパースできること
- ✓ 末端のプロパティへの固定値制約の複数指定が正しくパースできること
- ✓ 中間のプロパティへの固定値制約の複数指定が正しくパースできること
- ✓ クラス、中間、末端のプロパティへの固定値制約が正しくパースできること
- ✓ 末端のプロパティへの型制約が正しくパースできること
- ✓ 中間のプロパティへの型制約が正しくパースできること

passes: 255 failures: 0 duration: 1.60s 100%

テストスクリプトを表示できます。

```
expect(IMIToolsCore.parse_k("関与>関与者{0人}>氏名>姓名")).deep.equal({  
  "name": "関与",  
  "next": {  
    "name": "関与者",  
    "restriction": [{  
      "name": "人"  
    }],  
    "next": {  
      "name": "氏名",  
      "next": {  
        "name": "姓名"  
      }  
    }  
  }  
});
```

基本動作で使用するvocabulary環境

1. 語彙記法で記述した、コア語彙

<https://imi.go.jp/ns/core/241/imicore241.imiv>

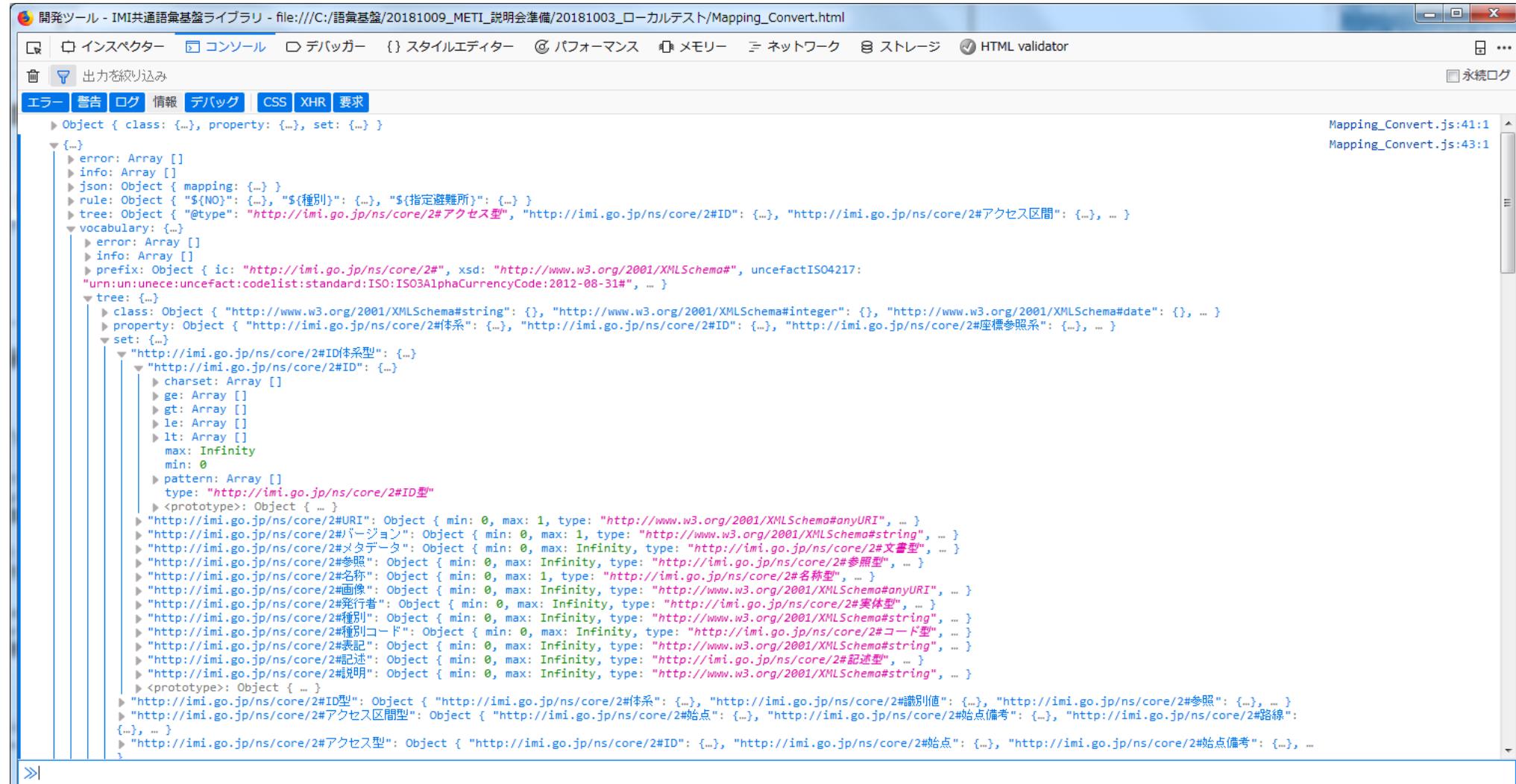
2. 語彙記法で記述した、利用するXSDスキーマ

```
#prefix xsd: "http://www.w3.org/2001/XMLSchema#"  
#prefix ns1: "urn:un:unece:uncefact:codelist:standard:UNECE:MeasurementUnitCommonCode:9#"  
#prefix ns2: "urn:un:unece:uncefact:codelist:standard:UNECE:MeasurementUnitCommonCodeDuration:4#"  
#prefix ns3: "urn:un:unece:uncefact:codelist:standard:ISO:ISO3AlphaCurrencyCode:2012-08-31#"  
#prefix ns4: "urn:un:unece:uncefact:codelist:standard:UNECE:MeasurementUnitCommonCodeLinear:4#"  
#prefix ns5: "urn:un:unece:uncefact:codelist:standard:UNECE:MeasurementUnitCommonCodeVolume:4#"  
#prefix ns6: "urn:un:unece:uncefact:codelist:standard:UNECE:MeasurementUnitCommonCodeWeight:4#"  
  
vocabulary "http://www.w3.org/2001/XMLSchema#" ;
```

```
class xsd:string ;  
class xsd:integer ;  
class xsd:date ;  
class xsd:dateTime ;  
class xsd:anyURI ;  
class xsd:decimal ;  
class xsd:time ;  
class xsd:double ;  
class xsd:nonNegativeInteger ;  
  
class ns1:MeasurementUnitCommonCodeContentType ;  
class ns2:MeasurementUnitCommonCodeDurationContentType ;  
class ns3:ISO3AlphaCurrencyCodeContentType ;  
class ns4:MeasurementUnitCommonCodeLinearContentType ;  
class ns5:MeasurementUnitCommonCodeVolumeContentType ;  
class ns6:MeasurementUnitCommonCodeWeightContentType ;
```



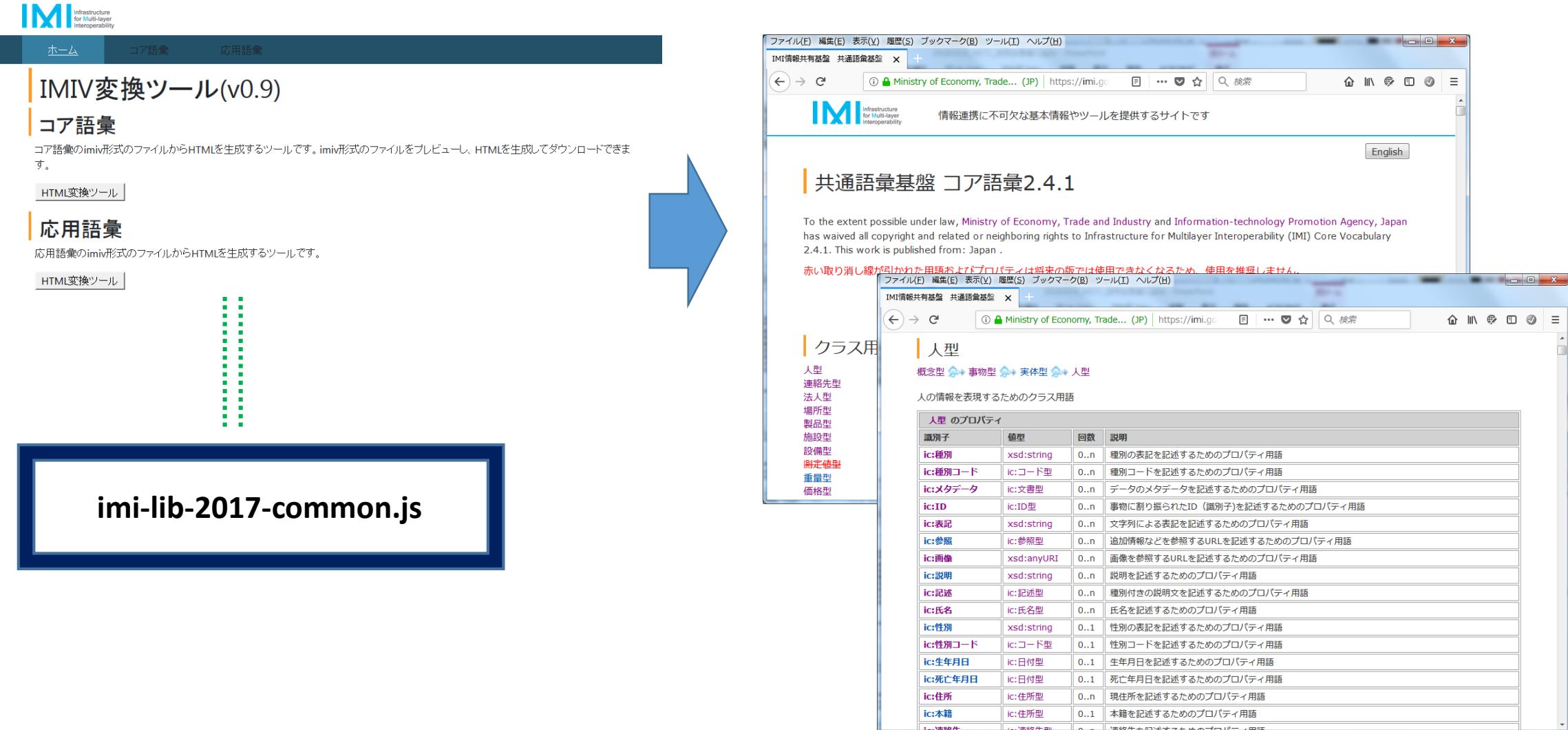
オブジェクトの参照



例えば、vocabulary.tree.set内では、記法から展開されたvocabulary setを見ることが出来ます。

IMI共通語彙基盤ライブラリの実装例

コア語彙のhtmlは、本ライブラリを使ってIMI事務局内で作成した「IMIV変換ツール」が実現しています!!



The diagram illustrates the process of generating IMI Core Vocabulary pages from IMIV files. On the left, the 'IMIV変換ツール(v0.9)' interface is shown, featuring sections for 'コア語彙' and '応用語彙', each with a 'HTML変換ツール' button. A large blue arrow points from this interface to the right, where the resulting pages are displayed. On the right, the 'IMI情報共有基盤 共通語彙基盤' page is shown, with the title '共通語彙基盤 コア語彙2.4.1'. Below the title, a note states: 'To the extent possible under law, [Ministry of Economy, Trade and Industry](#) and [Information-technology Promotion Agency, Japan](#) 2.4.1. This work is published from: Japan.' The page then lists '人型' (Human Type) under 'クラス用' (Class Use), with a table of properties:

識別子	値型	回数	説明
ic:種別	xsd:string	0..n	種別の表記を記述するためのプロパティ用語
ic:種別コード	ic:コード型	0..n	種別コードを記述するためのプロパティ用語
ic:メタデータ	ic:文書型	0..n	データのメタデータを記述するためのプロパティ用語
ic:ID	ic:ID型	0..n	事物に割り振られたID（識別子）を記述するためのプロパティ用語
ic:表記	xsd:string	0..n	文字列による表記を記述するためのプロパティ用語
ic:参照	ic:参照型	0..n	追加情報などを参照するURLを記述するためのプロパティ用語
ic:画像	xsd:anyURI	0..n	画像を参照するURLを記述するためのプロパティ用語
ic:説明	xsd:string	0..n	説明を記述するためのプロパティ用語
ic:記述	ic:記述型	0..n	種別付きの説明文を記述するためのプロパティ用語
ic:氏名	ic:氏名型	0..n	氏名を記述するためのプロパティ用語
ic:性別	xsd:string	0..1	性別の表記を記述するためのプロパティ用語
ic:性別コード	ic:コード型	0..1	性別コードを記述するためのプロパティ用語
ic:生年月日	ic:日付型	0..1	生年月日を記述するためのプロパティ用語
ic:死亡年月日	ic:日付型	0..1	死亡年月日を記述するためのプロパティ用語
ic:住所	ic:住所型	0..n	現住所を記述するためのプロパティ用語
ic:本籍	ic:住所型	0..1	本籍を記述するためのプロパティ用語

IMI共通語彙基盤ライブラリ DEMO

DEMO1

testツールを拡張して、CSVデータをRDF/XMLに変換してDEBUG情報とともに表示してみる。

DEMO2

DMDの選択、マッピングファイルの選択、対象データを各種の構造化データに変換まで行ってみる。
また、ボキャブラリ(語意記法)からxsd schemaを作成する!!