

オープンデータ
なんかでよく使う
Turtle形式なんだけど
XMLと違ってどう便利なの？

@SMAGCH

```
:catalog
  a dcat:Catalog ;
  dct:title "Imaginary Catalog" ;
  rdfs:label "Imaginary Catalog" ;
  foaf:homepage <http://example.org/catalog> ;
  dct:publisher :transparency-office ;
  dct:language <http://id.loc.gov/vocabulary/iso639-1/en> ;
  dcat:dataset :dataset-001 , :dataset-002 , :dataset-003 ;
  .
```

<http://www.w3.org/TR/vocab-dcat/#example>

dcatのSpecificationに出てくる

RDFの例ではXMLではなく、

Turtle形式で記述されている。

英語のWikiに出てくる

RDFの例から、

XMLとTurtleの

形式の違いを見比べてみる

```
@prefix eric:    <http://www.w3.org/People/EM/contact#> .
@prefix contact: <http://www.w3.org/2000/10/swap/pim/contact#> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
eric:me contact:fullName "Eric Miller" .
eric:me contact:mailbox <mailto:e.miller123(at)example> .
eric:me contact:personalTitle "Dr." .
eric:me rdf:type contact:Person .
```

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#" xmlns:eric="http://www.w3.org/
People/EM/contact#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:mailbox rdf:resource="mailto:e.miller123(at)example"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:personalTitle>Dr.</contact:personalTitle>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
    <rdf:type rdf:resource="http://www.w3.org/2000/10/swap/pim/contact#Person"/>
  </rdf:Description>
</rdf:RDF>
```

http://en.wikipedia.org/wiki/Resource_Description_Framework

```
<http://www.w3.org/People/EM/contact#me>  
  <http://www.w3.org/2000/10/swap/pim/contact#fullName>  
    "Eric Miller" .
```

```
@prefix eric:    <http://www.w3.org/People/EM/contact#> .  
@prefix contact: <http://www.w3.org/2000/10/swap/pim/contact#> .  
  
eric:me contact:fullName "Eric Miller" .
```

**XMLのxmlnsみたいに、@prefixで
エイリアスを作れる。
上の2つは同じ。**



EXAMPLE 1



```
@base <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel: <http://www.perceive.net/schemas/relationship/> .

<#green-goblin>
  rel:enemyOf <#spiderman> ;
  a foaf:Person ;      # in the context of the Marvel universe
  foaf:name "Green Goblin" .

<#spiderman>
  rel:enemyOf <#green-goblin> ;
  a foaf:Person ;
  foaf:name "Spiderman", "Человек-паук"@ru .
```

http://www.w3.org/TR/turtle/#h2_sec-intro

Predicate Lists

EXAMPLE 3

```
<http://example.org/#spiderman>  
  <http://www.perceive.net/schemas/relationship/enemyOf>  
    <http://example.org/#green-goblin> ;  
<http://xmlns.com/foaf/0.1/name>  
  "Spiderman" .
```

EXAMPLE 4

```
<http://example.org/#spiderman>  
  <http://www.perceive.net/schemas/relationship/enemyOf>  
    <http://example.org/#green-goblin> .  
  
<http://example.org/#spiderman>  
  <http://xmlns.com/foaf/0.1/name>  
  "Spiderman" .
```


Object Lists

EXAMPLE 5

```
<http://example.org/#spiderman>  
  <http://xmlns.com/foaf/0.1/name>  
    "Spiderman", "Человек-паук"@ru .
```

EXAMPLE 6

```
<http://example.org/#spiderman>  
  <http://xmlns.com/foaf/0.1/name> "Spiderman" .  
<http://example.org/#spiderman>  
  <http://xmlns.com/foaf/0.1/name> "Человек-паук"@ru .
```

IRIs

EXAMPLE 7

```
@prefix somePrefix: <http://www.perceive.net/schemas/relationship/> .  
  
<http://example.org/#green-goblin> somePrefix:enemyOf  
  <http://example.org/#spiderman> .
```

@prefixと@baseは case sensitive

EXAMPLE 8

```
PREFIX somePrefix: <http://www.perceive.net/schemas/relationship/>  
  
<http://example.org/#green-goblin> somePrefix:enemyOf  
  <http://example.org/#spiderman> .
```

後からサポートされたSPARQLシンタックス。PREFIX, BASEは case insensitive。終わりのドットなし

http://www.w3.org/TR/turtle/#h3_sec-iri

EXAMPLE 9

```
# A triple with all absolute IRIs
<http://one.example/subject1> <http://one.example/predicate1> <http://one.example/object1> .

@base <http://one.example/> .
<subject2> <predicate2> <object2> .      # relative IRIs, e.g. http://one.example/subject2

BASE <http://one.example/>
<subject2> <predicate2> <object2> .      # relative IRIs, e.g. http://one.example/subject2

@prefix p: <http://two.example/> .
p:subject3 p:predicate3 p:object3 .      # prefixed name, e.g. http://two.example/subject3

PREFIX p: <http://two.example/>
p:subject3 p:predicate3 p:object3 .      # prefixed name, e.g. http://two.example/subject3

@prefix p: <path/> .                      # prefix p: now stands for http://one.example/path/
p:subject4 p:predicate4 p:object4 .      # prefixed name, e.g. http://one.example/path/subject4

@prefix : <http://another.example/> .     # empty prefix
:subject5 :predicate5 :object5 .        # prefixed name, e.g. http://another.example/subject5

:subject6 a :subject7 .                  # same as :subject6 <http://www.w3.org/1999/02/22-rdf-syntax-
ns#type> :subject7 .

<http://伝言.example/?user=مَرْيَم&channel=R%26D> a :subject8 . # a multi-script subject IRI .
```

http://www.w3.org/TR/turtle/#h3_sec-iri

Literals

EXAMPLE 10

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
<http://example.org/#green-goblin> foaf:name "Green Goblin" .  
<http://example.org/#spiderman> foaf:name "Spiderman" .
```

Quoted Literals

EXAMPLE 11

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix show: <http://example.org/vocab/show/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

show:218 rdfs:label "That Seventies Show"^^xsd:string .
show:218 rdfs:label "That Seventies Show"^^<http://www.w3.org/2001/XMLSchema#string> .
show:218 rdfs:label "That Seventies Show" .
show:218 show:localName "That Seventies Show"@en .
show:218 show:localName 'Cette Série des Années Soixante-dix'@fr .
show:218 show:localName "Cette Série des Années Septante"@fr-be .
show:218 show:blurb '''This is a multi-line
literal with many quotes ("''''''')
and up to two sequential apostrophes ('').''' .
```

Numbers & Booleans

EXAMPLE 12

```
@prefix : <http://example.org/elements> .  
<http://en.wikipedia.org/wiki/Helium>  
  :atomicNumber 2 ;           # xsd:integer  
  :atomicMass 4.002602 ;      # xsd:decimal  
  :specificGravity 1.663E-4 . # xsd:double
```

EXAMPLE 13

```
@prefix : <http://example.org/stats> .  
<http://somecountry.example/census2007>  
  :isLandlocked false .      # xsd:boolean
```

Blank Nodes

EXAMPLE 14

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
_:alice foaf:knows _:bob .  
_:bob foaf:knows _:alice .
```

Nesting Unlabeled Blank Nodes

EXAMPLE 15

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
# Someone knows someone else, who has the name "Bob".  
[] foaf:knows [ foaf:name "Bob" ] .
```


EXAMPLE 16

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

```
[ foaf:name "Alice" ] foaf:knows [
  foaf:name "Bob" ;
  foaf:knows [
    foaf:name "Eve" ] ;
  foaf:mbox <bob@example.com> ] .
```

EXAMPLE 17

```
_:a <http://xmlns.com/foaf/0.1/name> "Alice" .
_:a <http://xmlns.com/foaf/0.1/knows> _:b .
_:b <http://xmlns.com/foaf/0.1/name> "Bob" .
_:b <http://xmlns.com/foaf/0.1/knows> _:c .
_:c <http://xmlns.com/foaf/0.1/name> "Eve" .
_:b <http://xmlns.com/foaf/0.1/mbox> <bob@example.com> .
```

Collections

EXAMPLE 18

```
@prefix : <http://example.org/foo> .  
# the object of this triple is the RDF collection blank node  
:subject :predicate ( :a :b :c ) .  
  
# an empty collection value - rdf:nil  
:subject :predicate2 () .
```

Predicates

述語

Dublin Core

@prefix dc: <http://purl.org/dc/elements/1.1/> .

@prefix dct: <http://purl.org/dc/terms/> .

@prefix dctype: <http://purl.org/dc/dcmitype/> .

汎用的に使えるのが多くて便利

<http://dublincore.org/documents/2012/06/14/dcmi-terms/>

The Friend of a Friend (FOAF)

@prefix foaf: <http://xmlns.com/foaf/0.1/> .

人間を記述するのに便利

<http://www.foaf-project.org/>

RDF Schema

@prefix rdfs: <<http://www.w3.org/2000/01/rdf-schema#>> .

**RDFの拡張で、
基本的なデータモデルの記述に便利。
ローレベルな感じ。**

<http://www.w3.org/TR/rdf-schema/>

RDF Syntax

@prefix rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>> .

**XML記述でなく Turtleだと、
あんまり使わないかも。**

rdf:object, rdf:type, rdf:value rdf:first
rdf:rest, rdf:nil

<http://www.w3.org/TR/rdf-syntax-grammar/>

Data Catalog Vocabulary (DCAT)

@prefix dcat: <<http://www.w3.org/ns/dcat#>> .

データカタログのためのVocabulary

複数のデータを、

まとめて扱う時に便利

<http://www.w3.org/TR/vocab-dcat/>

YAN Place Vocabulary

YAN Event Vocabulary

ヨコハマ・LODのオリジナル語彙

http://fp.yafjp.org/yokohama_art_lod/place_rdf

http://fp.yafjp.org/yokohama_art_lod/event_rdf

リンクなど


http://fp.yafjp.org/yokohama_art_lod

ヨコハマ・アート・LOD

<http://opendata.shiga.jp/hanabi2014/>

**びわ湖大花火大会のオープンデータ。
アプリもたくさん、作られてる。**

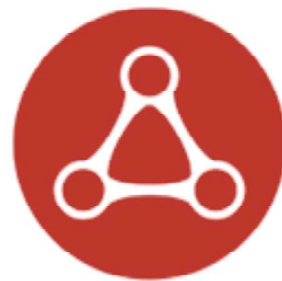
<http://lod.sfc.keio.ac.jp/challenge2014/>

 LOD Challenge 2014

・ 開催趣旨 ・ 応募情報 ・ スポンサー ・ イベント ・ LODチャレンジ受賞作品 ・ 公式ブログ



アイデア部門



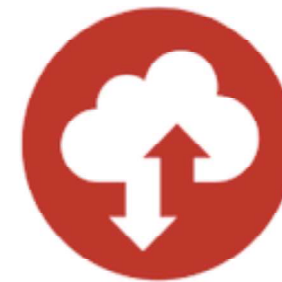
データセット部門



アプリケーション部門



ビジュアライゼーション部門



基盤技術部門



データセット



アイデア



アプリケーション



ビジュアライゼーション



基盤技術

応募期間 2014年10月1日(水)-2015年1月18日 (日)

授賞式2015年3月12日(木) 慶応三田キャンパス南校舎ホール

LODチャレンジJapan2013サイト

News

Event

おわり