

ウェブインテリジェンスを支える XML (再) 入門

「RDF (再) 入門」

的野晃整

産業技術総合研究所

情報技術研究部門

目次

- **RDFとは**
 - RDFモデル
 - RDF構文
 - RDFスキーマ
 - RDF応用
- **RDF問合せ**
 - RDF問合せ
 - RDFデータベース

RDFとは

クイズ: XML応用

- **XHTML** (The Extensible HyperText Markup Language)
- **SVG** (Scalable Vector Graphics)
- **SOAP** (Simple Object Access Protocol)
- **XSLT** (XML Stylesheet Language Transformations)
- **GML** (Geography Markup Language)
- **MathML** (Mathematical Markup Language)
- **SMIL** (Synchronized Multimedia Integration Language)
- **RDF** (Resource Description Framework)
 - 実はRDFだけが仲間はずれ。
 - 他のXML応用と何が違うのでしょうか？
 - ヒント：RDFの特徴が答えになってます。

RDFで記述できるモノ

- メタデータとは？

- データについてのデータ

- データとメタデータの明確な区別はない

- 例えば

- 書籍 

- 著者、タイトル、出版社、ページ数、ISBN など

- 人物 

- 名前、生年月日、メールアドレス、写真など

- 画像 

- 撮影日時、撮影場所、解像度など

RDF誕生の経緯

- 2001年頃

- Semantic Webの流行

- Tim Berners-Lee (Webを作った人) が提唱

- 機械が理解できるWeb

- Web上のあらゆる資源にメタデータを付与

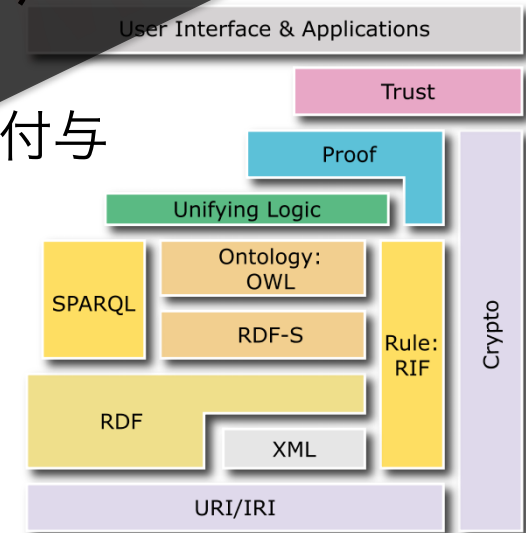
- 記述や交換のための統一ルールが必要

RDFが誕生

- 現在では…

- Semantic Webの流行には陰り

- しかし、RDFはメタデータ記述の地位を確立



メタデータの重要性

- **Web 2.0**
 - 集合知やタグ
- **ユビキタスコンピューティング**
 - トレーサビリティ
 - 自律移動支援
- **SOA (サービス指向アーキテクチャ)**
 - Webサービスの検索
- **Grid コンピューティング**
 - 動的なリソース分配

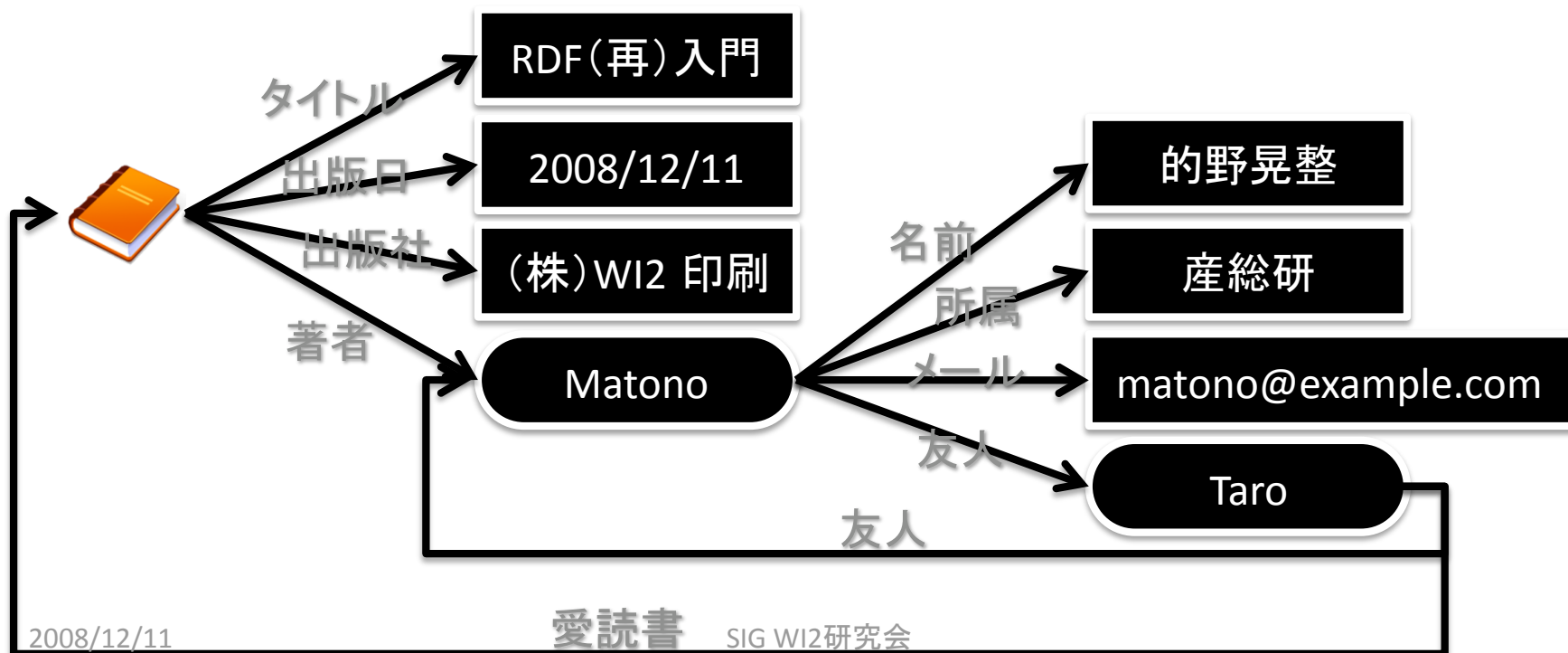
メタデータなしでは、実現不可能

クイズの答え 其の1

RDFは他のXML応用と何が違う？

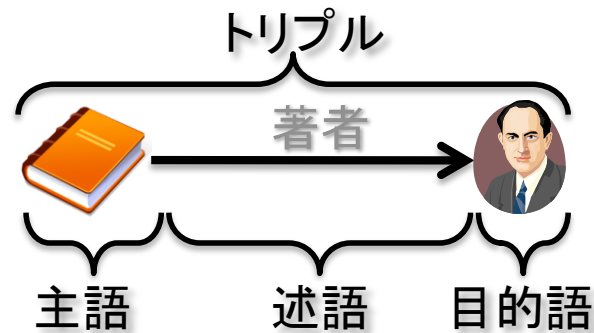
- モデルが違う

- 他のXML応用のモデルは木
- RDFのモデルは有向グラフ ⇔ 構文は木



RDFモデル

- トリプルの集合で有向グラフを表現
- トリプル (triple、ステートメント、文) とは
 - 構造：主語 (subject)、述語 (predicate)、目的語 (object) の三つ組



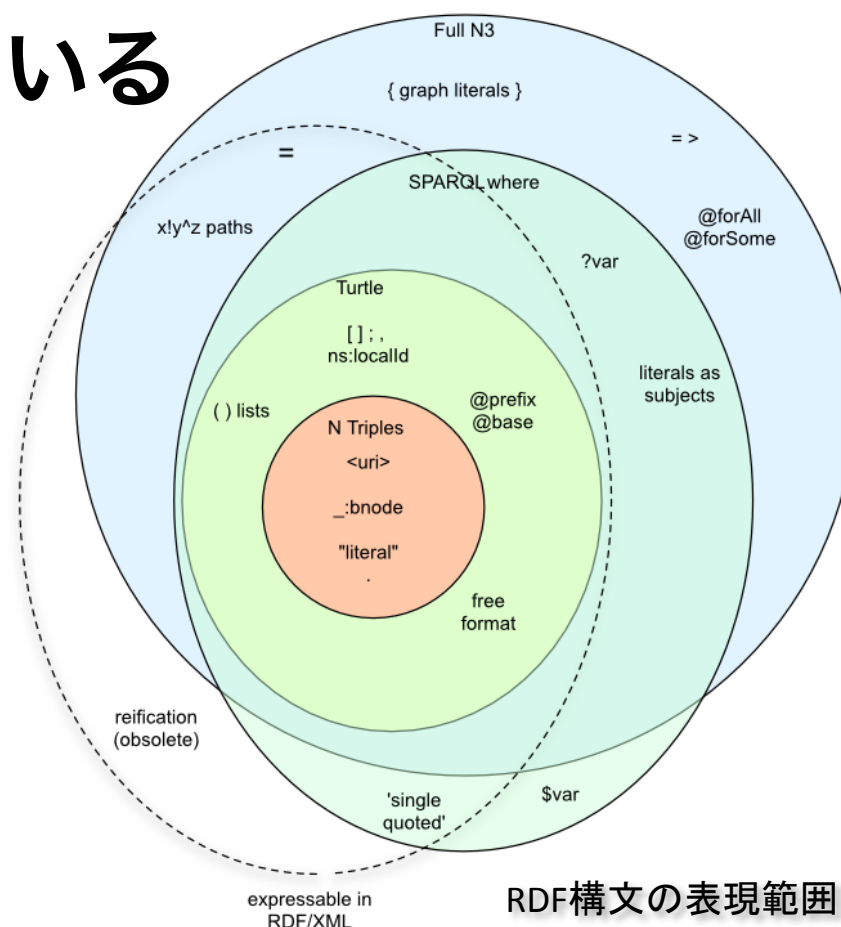
- 意味：この本の著者はこの人物である。
- 資源(リソース)とリテラル
 - 資源はURIで識別、リテラルは文字列や数字、日付など
 - 主語と述語：資源、目的語：資源 or リテラル



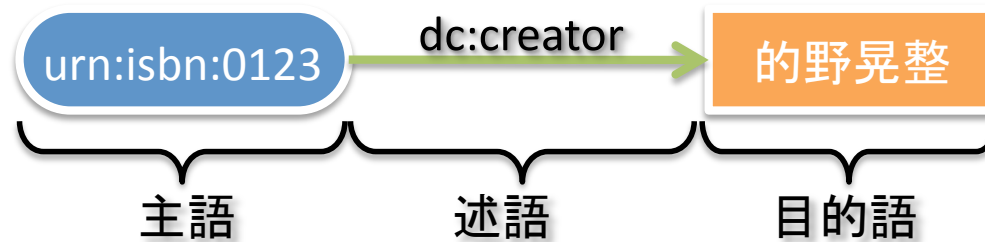
クイズの答え 其の2

RDFは他のXML応用と何が違う？

- RDFは枠組みであって、言語ではない
- 構文は別で定義されている
 - RDF/XML
 - N3 (Notation3)
 - N-Triples
 - TriG
 - TriX (RDF Triples in XML)
 - RXR (Regular XML RDF)
 - Turtle (Terse RDF Triple Language)



RDF/XMLの例1



```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  ...
  <rdf:Description rdf:about="urn:isbn:0123">
    <dc:creator>的野晃整</dc:creator>
  </rdf:Description>
</rdf:RDF>
```

XML名前空間

トリプルの始まりを表わす
主語となる資源のURI

```
<rdf:RDF ...>
  <rdf:Description rdf:about="urn:isbn:0123" dc:creator="的野晃整" />
</rdf:RDF>
```

RDF/XMLの例2

```
<rdf:RDF
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
```

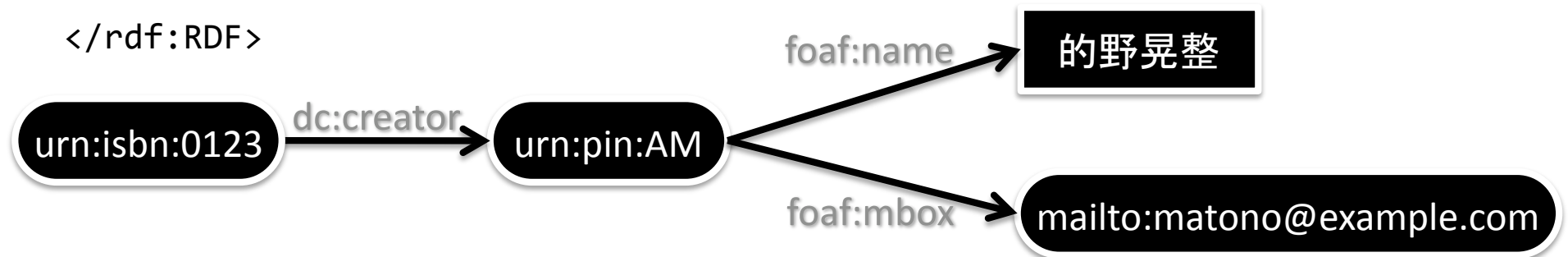
} XML名前空間

```
  <rdf:Description rdf:about="urn:isbn:0123">
    <dc:creator rdf:resource="urn:pin:AM"/>
  </rdf:Description>
```

述語となる資源のURI

```
  <rdf:Description rdf:about="urn:pin:AM">
    <foaf:name>的野晃整</foaf:name>
    <foaf:mbox rdf:resource="mailto:matono@example.com"/>
  </rdf:Description>
```

```
</rdf:RDF>
```



RDF/XMLの例2

```
<rdf:RDF
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/">
```

} XML名前空間

```
  <rdf:Description rdf:about="urn:isbn:0123">
```

```
    <dc:creator>
```

```
      <rdf:Description rdf:about="urn:pin:AM">
```

```
        <foaf:name>的野晃整</foaf:name>
```

```
        <foaf:mbox rdf:resource="mailto:matono@example.com"/>
```

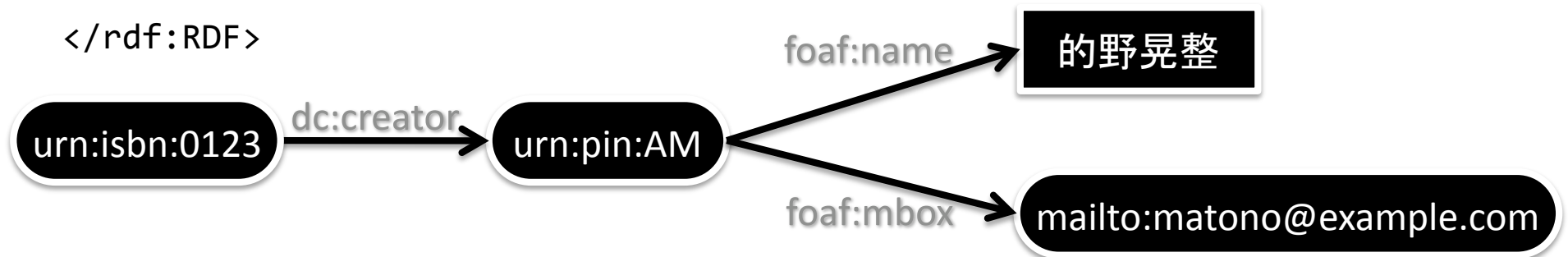
```
      </rdf:Description>
```

```
    </dc:creator>
```

```
  </rdf:Description>
```

} 入れ子

```
</rdf:RDF>
```

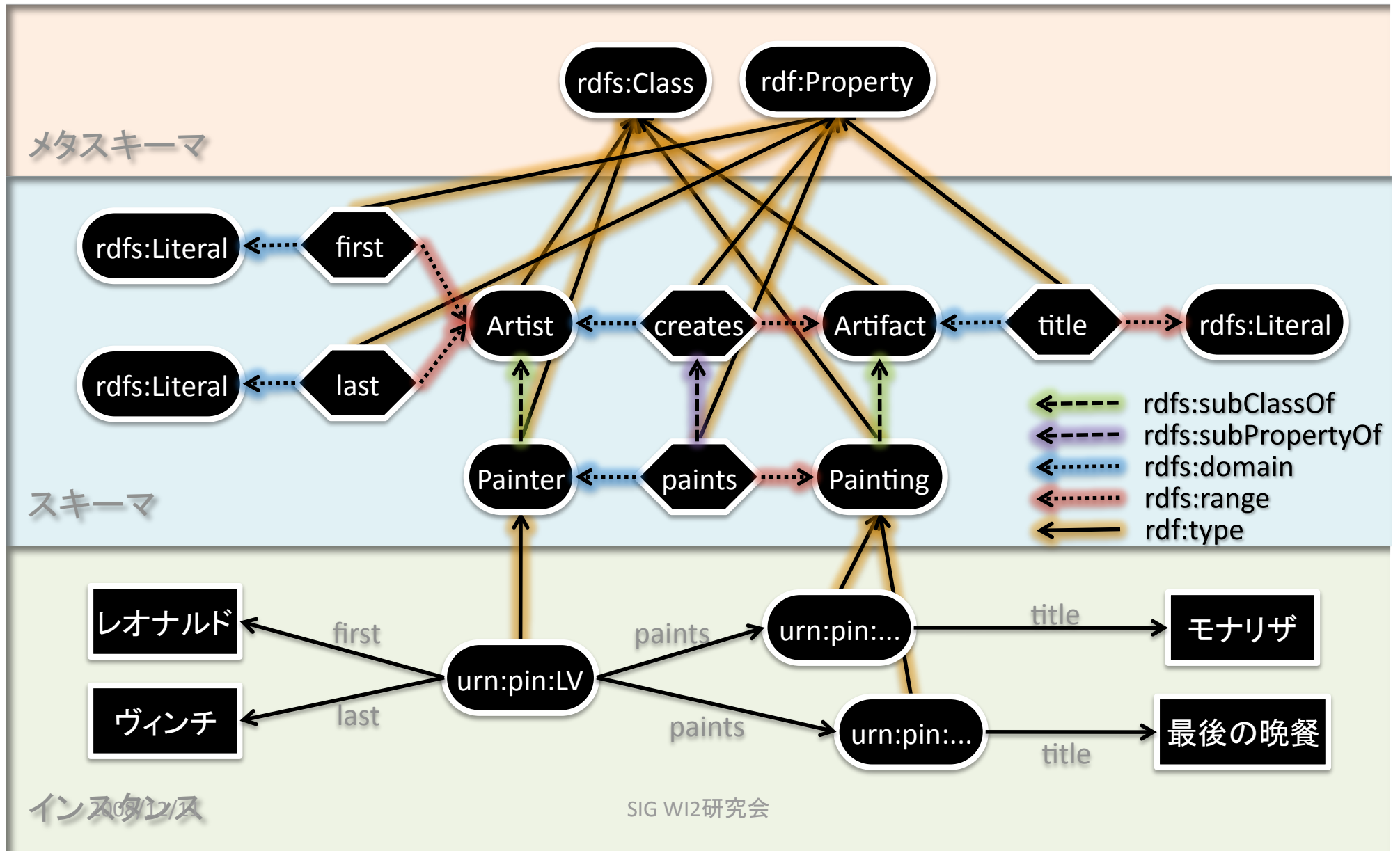


クイズの答え 其の3

RDFは他のXML応用と何が違う？

- 他のXML応用は拡張不可
- RDFはスキーマ定義によって拡張可能
 - RSS (RDF Site Summary)
 - FOAF (Friend of a Friend)
 - Dublin Core
 - Firefoxアドオンのメタデータ
 - RDF Schema
 - OWL (Web Ontology Language)

RDFスキーマの例



RDFスキーマで何ができる？

- 妥当性検証

- 交換や共有のために、語彙を統一

- 簡易な推論

- クラスやプロパティの派生関係のみ
- 例：「Person のインスタンスをすべて求める」
場合、Personの派生クラス、ArtistやPainterの
インスタンスも解に含まれる。
- より高度な推論のためには、オントロジーが必要。
 - オントロジー記述も実はRDF拡張のOWLで記述できる。

RDF応用

自分のFOAFを作ってみよう

<http://www.ldodds.com/foaf/foaf-a-matic>

FOAF-a-Matic

原作者[Leigh Dodds](#), 翻訳者[神崎正英](#).

はじめに

FOAF-a-maticは、FOAF ("Friend-of-A-Friend")によるあなたの紹介を作成するシンプルな Javascriptアプリケーションです。FOAFの詳細については、Edd Dumbillによる"[XML Watch: Finding friends with XML and RDF](#)"([日本語訳](#))、"[メタデータによる知人ネットワークの表現](#)"、[RDFWebのFOAFホームページ](#)、および [FOAFの語彙定義](#)を参照してください。

細かいことはさておき、FOAFとはひとことで言えば、XMLとRDFを使ってあなた自身を紹介する — 名前、メールアドレス、それに友人についてなど — ための手段です。こうすることで、ソフトウェアが(たとえばサーチエンジンの一部分として)あなたの紹介データを処理し、あなた自身やあなたが所属するコミュニティの情報を見つけたすといったことが可能になります。FOAFは、オンラインコミュニティにおけるさまざまな興味深い新展開をひきだす可能性を秘めているのです。Ben HammerselyがGuardian Unlimitedウェブサイトに書いた記事"[Click to the Clique](#)"には、これらのアイデアについて更に掘り下げた記述があります。

FOAF-a-Maticは、あなたのFOAFによる自己紹介を素早く簡単に作成するために用意されています。このページのフォームを使って、あなたが自己紹介に加えようと思う項目を記入してください。少なくとも、名前とメールアドレスだけは記入する必要があります。友人の場合も同様です。自己紹介に友人情報があれば、FOAF収集システムがあなたのFOAF自己紹介をインデックスに登録する際、みんなを人間のネットワークとしてつなぐことができるので、よかったら何名か加えてみてください(何人加えるかは、自由です)。

(注)あなたがこのページで入力した情報は、いかなる形でもほかで利用したり保存したりすることはありません。FOAFの生成はすべてクライアント側の処理で行われますから、あなたのプライバシーは安全です。

このアプリケーションについて意見がある場合、あるいはもっとFOAFについて知りたい場合は、[RDFWeb-devメーリングリスト](#)に参加しましょう。

更新情報: 現在、あなたのFOAFデータを作成・管理できるデスクトップ・アプリケーション [FOAF-a-Matic Mark 2](#)を作成中です。

2008/12/11

入力フォーム

RDF応用

自分のFOAFを作ってみよう

<http://www.ldodds.com/foaf/foaf-a-matic>

```
<rdf:RDF
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:admin="http://webns.net/mvcb/"
```

} XML名前空間

```
<foaf:PersonalProfileDocument rdf:about="">
```

```
  <foaf:maker rdf:resource="#me"/>
```

```
  <foaf:primaryTopic rdf:resource="#me"/>
```

```
  <admin:generatorAgent
```

```
    rdf:resource="http://www.ldodds.com/foaf/foaf-a-matic"/>
```

```
  <admin:errorReportsTo rdf:resource="mailto:leigh@ldodds.com"/>
```

```
</foaf:PersonalProfileDocument>
```

} 誰のFOAFなのか

```
<foaf:Person rdf:ID="me">
```

```
  <foaf:name>Akiyoshi MATONO</foaf:name>
```

```
  <foaf:title>Dr</foaf:title>
```

```
  <foaf:givenname>Akiyoshi</foaf:givenname>
```

```
  <foaf:family_name>MATONO</foaf:family_name>
```

```
  <foaf:nick>mato</foaf:nick>
```

} 名前、
肩書き、
ニックネーム

```
<foaf:mbox_sha1sum>d521b1bb0a1bf1c31532a4</foaf:mbox_sha1sum>
```

```
<foaf:homepage rdf:resource="http://example.com/~matono/">
```

```
<foaf:depiction rdf:resource="http://example.com/~matono/pic.jpg"/>
```

```
<foaf:phone rdf:resource="tel:012-345-6789"/>
```

```
<foaf:workplaceHomepage rdf:resource="http://www.aist.go.jp/">
```

```
<foaf:workInfoHomepage rdf:resource="http://www.dbgrid.org/">
```

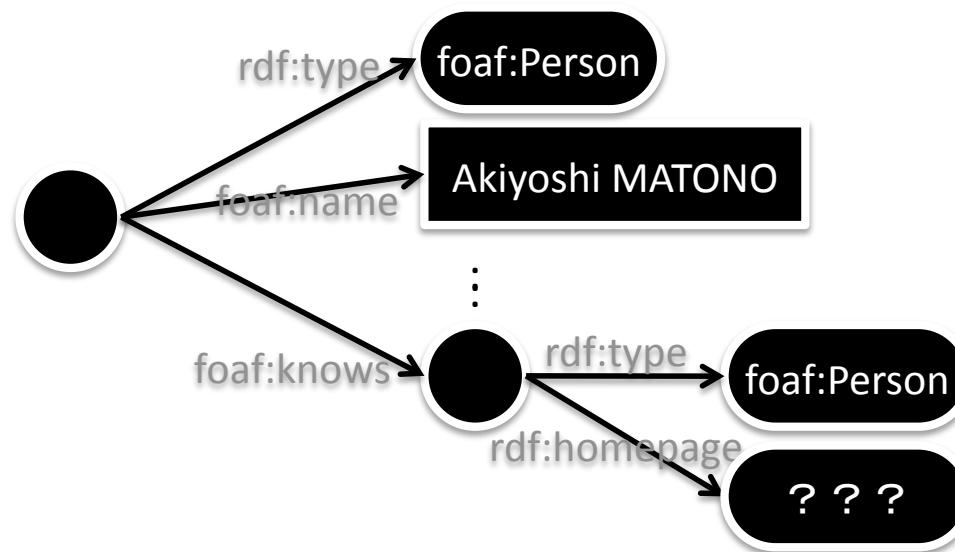
} メール、
ホームペー
ジ、
写真、
電話、

RDF問合せ

RDFデータに対する問合せ

- メタデータの検索

- そもそもメタデータは検索されるために付与
- 例えば、「知人全員のホームページは？」

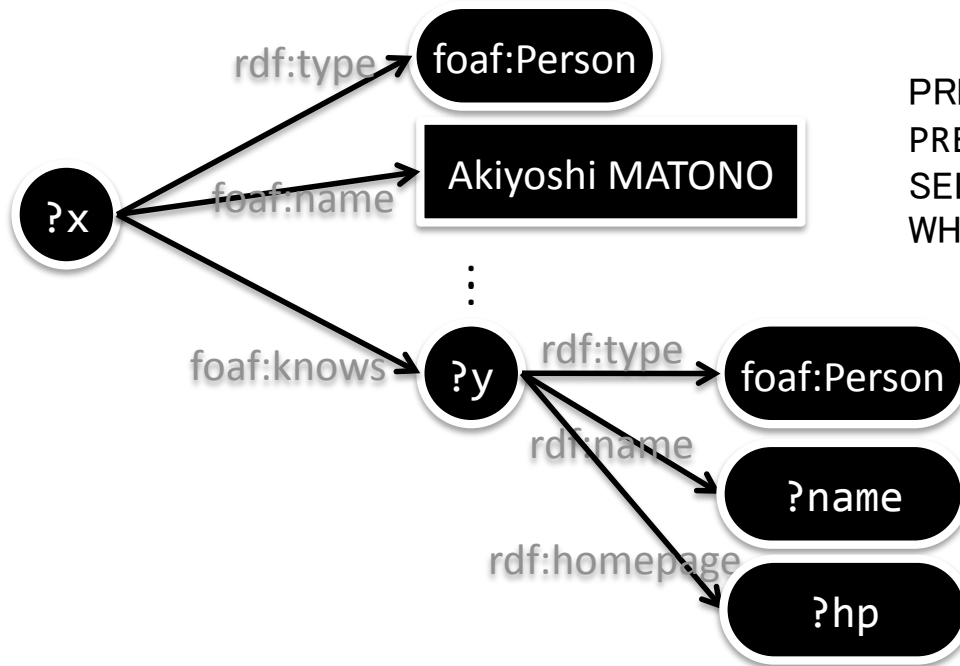


RDF問合せは、部分グラフ検索

問合せ言語：SPARQL

- SPARQL (SPARQL Protocol and RDF Query Language)
 - SQLに類似
 - SELECT, FROM, WHEREなど
 - 4種類の問合せ形式
 - **SELECT**：変数バインディング。関係DBと同じ形式
 - **CONSTRUCT**：与えたテンプレートに変数を代入して構築されたRDFグラフ。
 - **DESCRIBE**：資源に関するRDFグラフ。
 - **ASK**：問合せにマッチする部分グラフが存在するか否かのブール値。
 - FROM：問合せ対象のRDFデータを指定
 - 複数指定可。デフォルトグラフと名前付きグラフがある。
 - トリプルパターン
 - 複数の変数を含むトリプル
 - 例： { ?x foaf:name “Akiyoshi MATONO”. }
 - FILTER
 - 変数に条件を記述したフィルタリング
 - 例： { ?x foaf:age ?age . FILTER (?age > 30) . }
 - OPTIONAL
 - もし存在すれば、パターンに一致するグラフを返す
 - 関係演算で言うと、左外部結合)

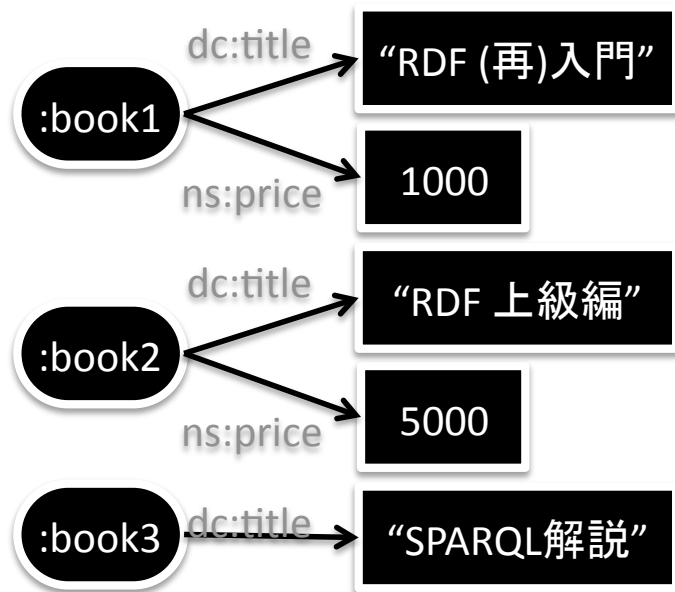
SPARQLの例1



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?hp
WHERE { ?x rdf:type foaf:Person .
        ?x foaf:name "Akiyoshi MATONO" .
        ?x foaf:knows ?y .
        ?y rdf:type foaf:Person .
        ?y foaf:name ?name .
        ?y foaf:homepage ?hp }
```

name	hp
"Takeo KUNISHIMA"	<http://example.com/~kunishima>
"Toshiyuki AMAGASA"	<http://example.com/~amagasa>
"Kenji HATANO"	<http://example.com/~hatano>
"Hiroshi SAKAMOTO"	<http://example.com/~sakamoto>

SPARQLの例2



```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX ns: <http://example.org/ns#>
SELECT ?title ?price
WHERE {
    ?x dc:title ?title .
    OPTIONAL { ?x ns:price ?price .
               FILTER (?price < 3000) }
}
```

title	price
“RDF (再)入門”	1000
“RDF 上級編”	
“SPARQL解説”	

RDFデータベース

名前	ストレージ	開発言語	問合せ言語	構文	推論	備考
Sesame	RDB, memory, native	Java	SPARQL, SeRQL,	RDF/XML, N3, N-Triples, turtle, trix, trig	Yes	それなりに速い
Jena	RDB, memory	Java	SPARQL	RDF/XML, N3, N-Triples, turtle,	Yes	有名
RDF Suite	RDB	Java/C++	RQL, RUL	RDF/XML	Yes	構造が特徴的
Redland	RDB, Berkeley DB, memory, native	C	SPARQL, RDQL	RDF/XML, N- Triples, turtle, trig	No	サポート言語が 豊富

RDFデータベースの構造

- Triple Store [Sesame, Jena, Redland, 他多数]
- Schema-based [RDFSuite]
- Multiple-indexing [Harth05, Wood05]
- Path-based [Kim05, Matono05]
- Property table [Wilkinson06]
- Vertical partitioning [Abadi07]
- Hexastore [Weiss08]

[Harth05] A. Harth and S. Decker. Optimized index structures for querying rdf from the web. In LA-WEB, 2005.

[Wood05] D. Wood et al. Kowari: A platform for Semantic Web storage and analysis. In XTech, 2005.

[Kim05] Y. Kim et al. The path index for query processing on RDF and RDF Schema. In ICACT, 2005.

[Matono05] A. Matono et al. A path-based relational RDF database. In ADC, 2005.

[Wilkinson06] K. Wilkinson. Jena property table implementation. In SSWS, 2006.

[Abadi07] D. J. Abadi et al. Scalable Semantic Web Data Management using vertical partitioning. In VLDB, 2007.

[Weiss08] C. Weiss et al. Hexastore: Sextuple Indexing for Semantic Web Data Management. In VLDB, 2008.

分散RDFデータベース

- Unstructured P2P [Nejdl02]
- Structured P2P [Cai04]
- Join index hierarchy [Stuckenschmidt04]
- Bloom filter-based [Heine06, 的野08]
- MapReduce-based [Tanimura09]

[Nejdl02] W. Nejdl et al. EDUTELLA: a P2P networking infrastructure based on RDF. in WWW, 2002.

[Stuckenschmidt04] H. Stuckenschmidt et al. Index Structures and Algorithms for Querying Distributed RDF Repositories. In WWW, 2004.

[Cai04] M. Cai et al. RDFPeers: A Scalable Distributed RDF Repository based on A Structured Peer-to-Peer Network. In WWW, 2004.

[Heine06] F. Heine, Scalable P2P based RDF Querying. in INFOSCALE, 2006.

[的野08] 的野ら、ブルームフィルタを用いた分散 RDF 問合せ処理のための索引手法. In DEWS, 2008

[Tanimura09] Tanimura et al. Storage Scheme for Parallel RDF Database Processing Using Distributed File System and MapReduce, In HPC-ASIA, 2009

今後の展望

- **RDFの課題**

- 問合せパフォーマンスの改善
 - スケーラビリティ
 - 結合演算
- オントロジー等の推論処理
 - 従来の関係演算の組み合わせでよい？
- アノテーション
 - 最大の課題

- **RDFの未来は明るい？**

- 正直わかりません。
- が、メタデータの未来が明るいのは確実