

# Metadata e RDF

---

Fabio Vitali



# Introduzione

Oggi esaminiamo in breve:

- ◆ L'esigenza di meta-informazioni
- ◆ PICS
- ◆ Il modello dei dati di RDF
- ◆ Aspetti della sintassi di RDF
- ◆ Il Dublin Core



# Meta-informazioni

Tutta la famiglia di linguaggi XML ha uno scopo: generare informazioni che non siano soltanto destinati alla lettura, ma che possano essere riutilizzati per applicazioni automatiche.

Ad esempio, non c'è niente in un documento HTML che indichi l'argomento trattato o la fonte delle informazioni. L'unico tipo di ricerca che posso fare su un documento dunque è la ricerca sul contenuto.

Questo non è sufficiente nella maggior parte delle volte: io ottengo qualche migliaio di hit la maggior parte dei quali non serve assolutamente a niente.

Le meta-informazioni permettono agli autori di specificare informazioni sui loro documenti che siano non soltanto leggibili, ma anche interpretabili in maniera intelligente dalle applicazioni di rielaborazione, e soprattutto dai motori di ricerca.



# La responsabilità autoriale sul Web

Nel 1995 il W3C era preoccupato perché l'esistenza di siti con materiale "sconveniente" (pornografia, hacking, estremismo politico, ecc.) stava danneggiando lo sviluppo della rete.

Il Web, per sua natura, non permette meccanismi di controllo e valutazione preventiva dei contenuti, né meccanismi seri di identificazione degli autori di un documento. Questo significa che chiunque voglia inserire dati e documenti "opinabili" può farlo senza sforzo, rischio personale e ricusabilità.

In particolare si temeva in molte sedi l'intervento pesante dei governi, la nascita di un organismo centrale (governativo o meno, comunque controllabile politicamente) che decidesse cosa fosse lecito mettere e cosa no.

L'organismo centrale è radicalmente opposto alla filosofia d'uso del Web e di Internet in generale, in quanto collo di bottiglia per l'uso e la distribuzione di informazioni sulla rete.



# PICS (1)

PICS (Platform for Internet Content Selection) fu la risposta del W3C alle lamentele sulla qualità delle informazioni sul web poste da associazioni di genitori, associazioni di educatori, associazioni religiose e politiche, ecc.

PICS è un brillante meccanismo per coinvolgere le stesse associazioni. Immediatamente adottato ed implementato dai principali produttori di browser, PICS richiede il contributo attivo e indispensabile della stessa associazione di categoria.

PICS non invoca nessun meccanismo di censura sui contenuti (peraltro impossibile in una realtà veramente internazionale), ma propone l'uso di meccanismi di valutazione (rating) dei contenuti del sito, e individua nei singoli browser il compito di attivare o meno un controllo.



# PICS (2)

Inoltre, non fornisce un sistema fisso di categorie e valori, ma permette a ciascuna associazione di definire le categorie rilevanti e le voci di ciascuna categoria.

Infine, non fornisce un meccanismo centralizzato per memorizzare i rating di ciascun sito, ma un'architettura distribuita secondo la quale il seguace, il genitore, la scuola si abbona ad un particolare sito di rating, stabilisce le politiche di selezione del contenuto visibile, e può visitare soltanto quel sottoinsieme di Web che viene reputato innocuo dallo specifico fornitore di rating.

PICS non stabilisce categorie pre-definite, ma un meccanismo generale per identificare categorie. E' quindi possibile usare PICS non solo per fornire censure locali a categorie, ma in generale per esprimere concetti di qualunque tipo sui siti.



# PICS (3)

Ad esempio, l'associazione delle biblioteche universitarie americane fornisce attraverso PICS un rating dell'autorevolezza delle informazioni tecniche contenute sui siti: non per fare censura, ma per guida e valutazione delle informazioni.

Questo é in definitiva un meccanismo funzionante e sofisticato per esprimere meta-informazioni (cioè valutazioni) su siti Web. Tuttavia PICS richiede di identificare in anticipo tutte le categorie rilevanti e tutti i valori da assegnare a qualunque categoria.

Questa limitazione, più il successo di XML, più la nascente idea di Web semantico, hanno portato alla creazione di RDF.



# Il Web semantico

Il W3C considera l'ideale evoluzione del Web dal machine-representable al machine-understandable. L'idea è di generare documenti e siti che possano al tempo stesso essere letti ed apprezzati da individui, ma anche acceduti ed interpretati da agenti automatici alla ricerca di contenuti.

Il Web si deve dunque dotare di una sovrastruttura di significato processabile da applicazioni, che lo renderebbe il Web semantico.

Tre sono le tecnologie chiave per questo sviluppo:

- ◆ XML, che permette di assegnare etichette e strutture descrittive ai documenti
- ◆ XPointer, che permette di individuare, negli URI, frammenti arbitrari di risorse pronte per il riuso.
- ◆ RDF, che permette di associare *affermazioni* a risorse, affermazioni che sono machine-processable e che permettono





# Applicazioni del Web semantico

- Resource discovery: per una efficienza migliore dei motori di ricerca
- Sistemi di catalogazione: per descrivere con esattezza il contenuto e le relazioni tra i contenuti di pagine, siti o librerie digitali
- Agenti software intelligenti: per facilitare l'acquisizione e lo scambio di nuova conoscenza
- Content rating: per descrivere collezioni di pagine secondo criteri di interesse o stima soggettivi.
- Collezioni: per rappresentare come connesse ed appartenenti ad un'unica opera pagine diverse magari su siti diversi
- Gestione del copyright, della proprietà intellettuale, e delle esigenze di privacy di singole pagine ed interi siti.



# RDF

E' una delle più importanti raccomandazioni del mondo W3C. TBL ci tiene moltissimo, il resto del W3C è variamente distribuito dagli entusiasti agli scettici.

Molti motori di ricerca stanno già usando RDF per descrivere il contenuto dei loro motori di ricerca. Tuttavia RDF ha senso se c'è attiva partecipazione da parte degli autori di siti, e ad oggi non esistono software ragionevoli per esprimere gradevolmente queste informazioni.

RDF è composto da due documenti:

- ◆ Model and Syntax Specification (W3C Recommendation del 22 febbraio 99): espone la struttura fondamentale del modello RDF, e descrive una possibile sintassi basata su XML.
- ◆ RDF Schema (W3C Candidate Recommendation 27 March 2000): espone la sintassi per definire schemi e vocabolari di metainformazioni.

Noi ci concentriamo solo sulla prima parte



# Il modello di RDF (1)

Il scopo sostanziale di RDF è quello di definire proprietà e valori a risorse descrivibili con URI.

Il concetto di URI è fondamentale per RDF (ogni cosa può essere descritta da un URI, sia documenti di rete che oggetti reali, come libri)

RDF definisce un modello concettuale fondamentale, e fornisce una possibile sintassi basata su XML.

La sintassi XML tuttavia non è necessaria: ogni rappresentazione che mantenga il modello è accettabile.



# Il modello di RDF (2)

Il modello di RDF è basato su tre concetti:

- ◆ La risorsa: le cose che vengono descritte, individuate o da un URI proprio, o da un URI a cui possono essere fatte risalire.
- ◆ Proprietà: uno specifico aspetto o attributo etichettato che voglio associare alla risorsa. Ogni proprietà ha un significato, una serie di valori leciti, è associabile ad uno o più tipi di risorsa, e si pone in relazione con altre proprietà.
- ◆ Affermazioni (*statement*): l'associazione di una proprietà ad una risorsa. Ogni affermazione ha una struttura obbligata del tipo “La proprietà X della risorsa Y vale Z”, o anche “Z è X per Y”

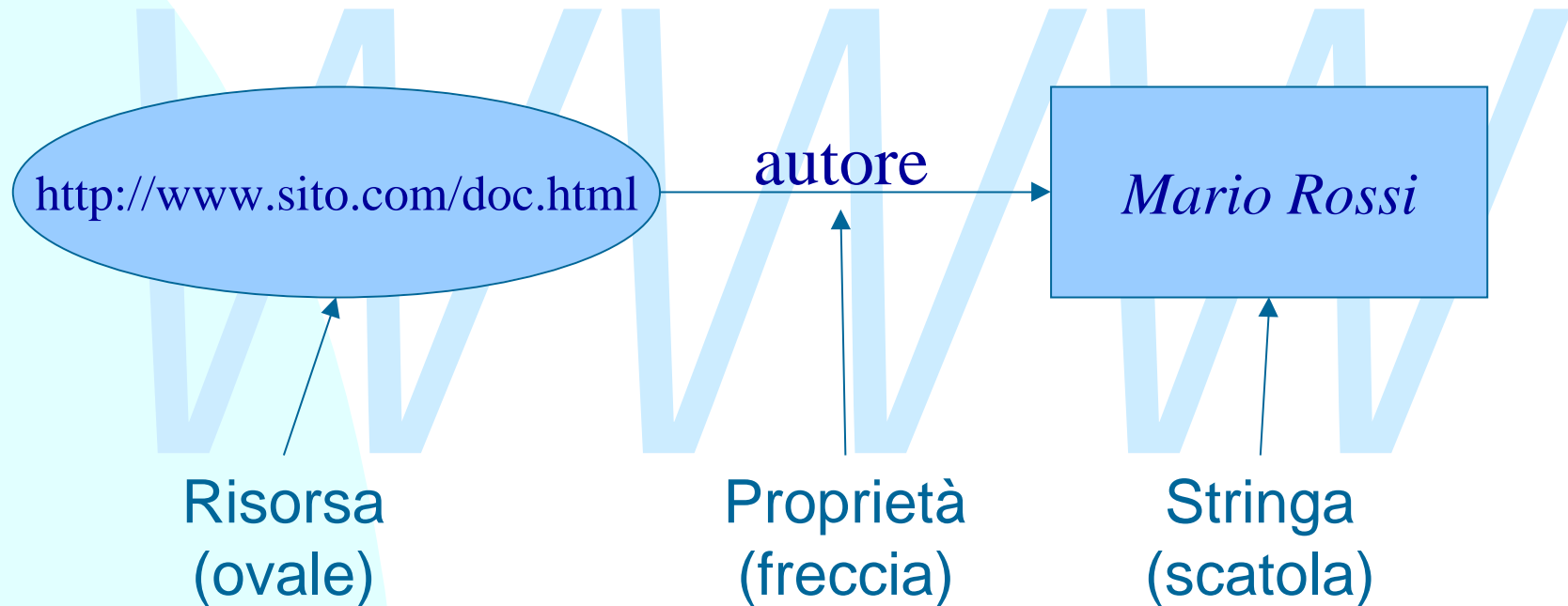
Ad esempio:

- ◆ La proprietà “autore” della risorsa “<http://www.sito.com/doc.html>” vale “Mario Rossi”
- ◆ “Mario Rossi è l'autore di <http://www.sito.com/doc.html>”



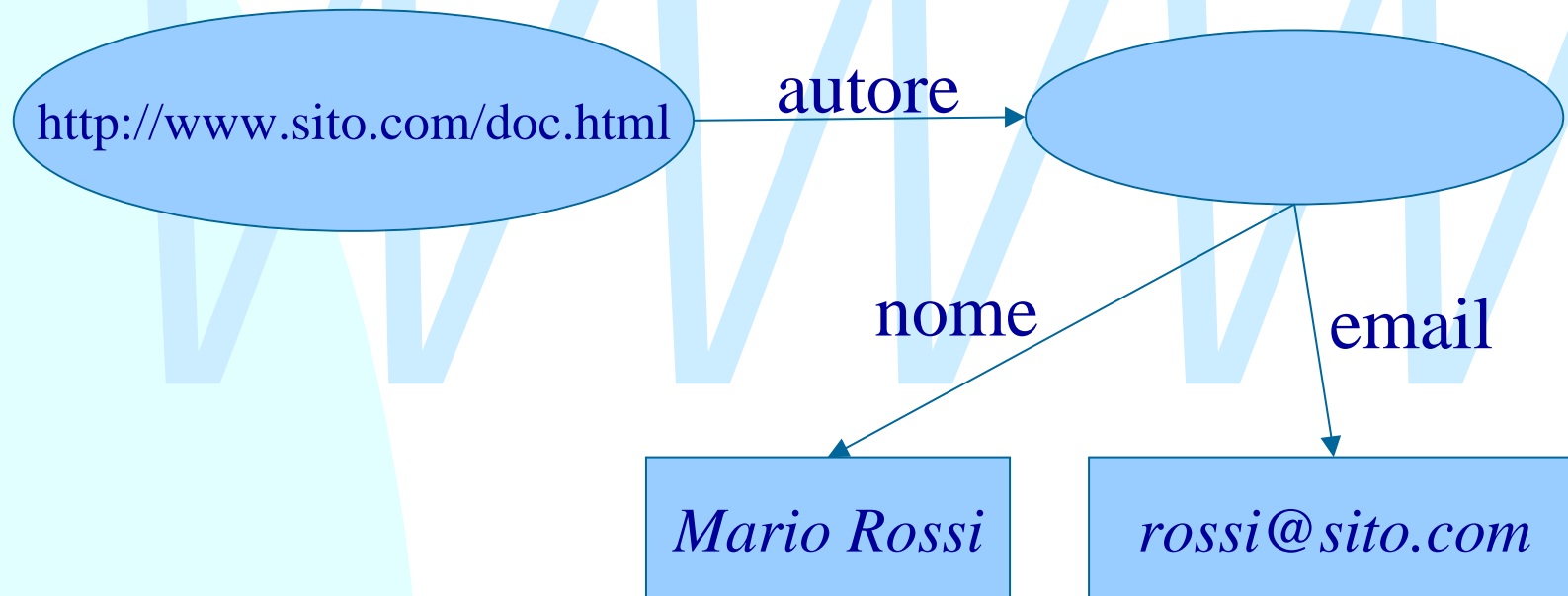
# Rappresentazione grafica (1)

*La proprietà “autore” della risorsa  
“<http://www.sito.com/doc.html>” vale “Mario Rossi”*



# Rappresentazione grafica (2)

*La proprietà “autore” della risorsa  
“<http://www.sito.com/doc.html>” è colui il cui nome è  
“Mario Rossi” e il cui e-mail è “rossi@sito.com”*



# Sintassi estesa

Il primo caso diventa in sintassi estesa:

```
<rdf:RDF>
  <rdf:Description about="http://www.sito.com/doc.html">
    <fv:autore>Mario Rossi</fv:autore>
  </rdf:Description>
</rdf:RDF>
```

E il secondo:

```
<rdf:RDF>
  <rdf:Description about="http://www.sito.com/doc.html">
    <fv:autore rdf:resource="#MR" />
  </rdf:Description>
  <rdf:Description id="MR">
    <q:nome>Mario Rossi</q:nome>
    <q:email>rossi@sito.com</q:email>
  </rdf:Description>
</rdf:RDF>
```



# Prima sintassi compatta

Esistono alcune forme equivalenti ma più compatte.  
Il primo esempio è equivalente alla seguente forma:

```
<rdf:Description  
  about="http://www.sito.com/doc.html"  
  fv:autore="Mario Rossi" />
```

Gli elementi di tipo stringa e non ripetuti vengono direttamente inseriti nel tag di "Description".





# Seconda sintassi compatta

Il secondo esempio è equivalente alla seguente forma:

```
<rdf:Description about="http://www.sito.com/doc.html">
  <fv:autore>
    <rdf:Description id="MR">
      <q:nome>Mario Rossi</q:nome>
      <q:email>rossi@sito.com</q:email>
    </rdf:Description>
  </fv:autore>
</rdf:Description>
```

Le descrizioni prive di identità autonoma possono essere annidate



# Tipi

E' possibile assegnare ad ogni Description un tipo appartenente ad uno schema di meta-informazioni (progettabile attraverso la definizione di schemi):

```
<rdf:Description about="http://www.sito.com/doc.html">
  <fv:autore>
    <rdf:Description id="MR">
      <rdf:type resource="/schema/persona"/>
      <q:nome>Mario Rossi</q:nome>
      <q:email>rossi@sito.com</q:email>
    </rdf:Description>
  </fv:autore>
</rdf:Description>
```

L'attributo type specifica l'URI della definizione del tipo nello schema usato.



# Terza sintassi compatta

E' possibile usare il valore del tipo come nome di elemento della descrizione relativa:

```
<rdf:Description about="http://www.sito.com/doc.html">
  <fv:autore>
    <fv:persona id="MR">
      <q:nome>Mario Rossi</q:nome>
      <q:email>rossi@sito.com</q:email>
    </fv:persona>
  </fv:autore>
</rdf:Description>
```

Lo schema dà cioè nome alle risorse anonime viste in precedenza.



# Contenitori

A volte è importante fare riferimento ad un insieme di risorse (ad esempio, se un documento è stato creato da più autori, o se lo stesso autore ha fatto più di un documento, ecc.)

In questo caso si deve inserire la risorsa all'interno di un contenitore a cui verrà applicata la proprietà.

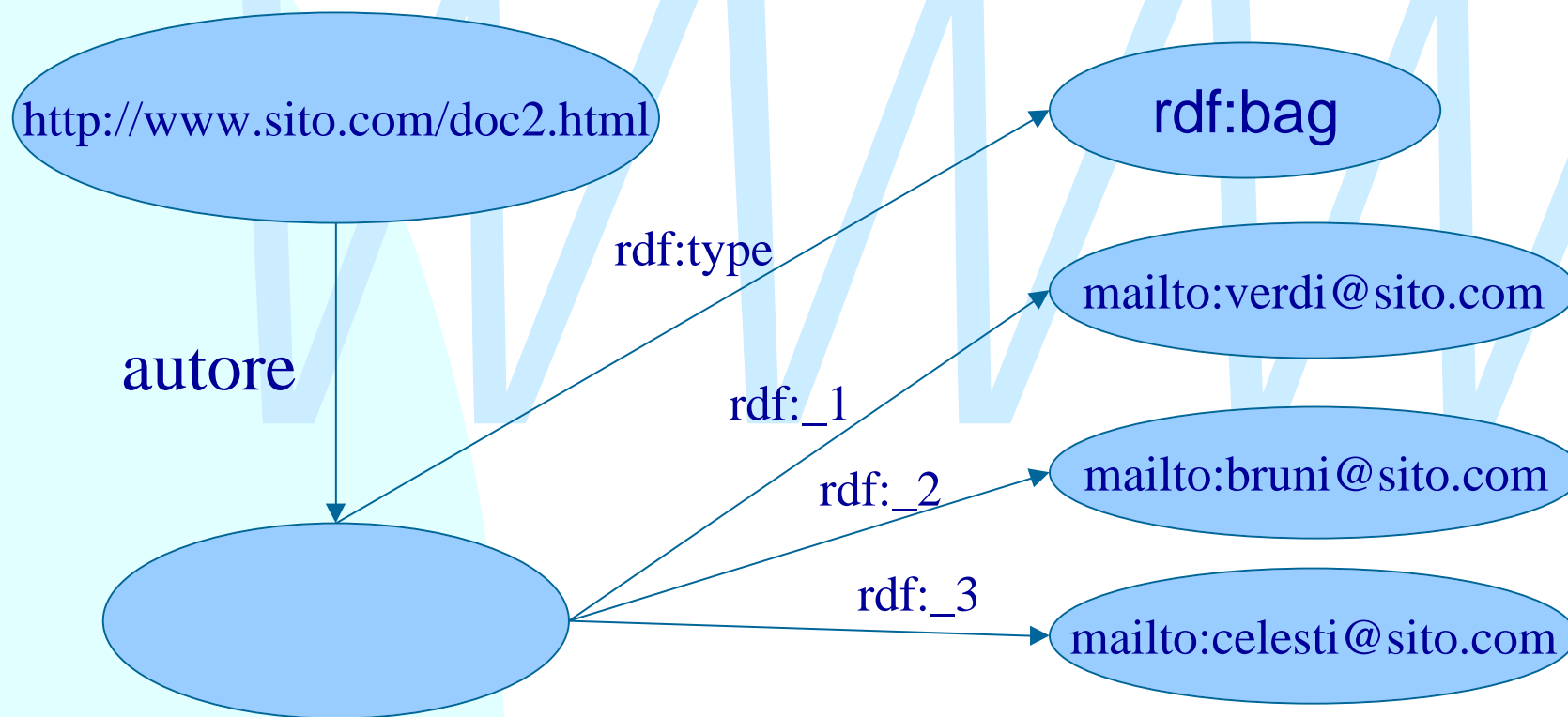
RDF definisce tre tipi di contenitori:

- ◆ **Bag.** Il bag è un insieme con ripetizioni. L'ordine non è rilevante, ma è possibile inserire più volte lo stesso elemento.
- ◆ **Sequenza.** La sequenza è un insieme con ripetizioni ed un ordine definito tra le risorse presenti.
- ◆ **Alternativa.** L'alternativa è un insieme senza ripetizioni in cui può essere scelto uno solo degli elementi. L'ordine degli elementi può essere usato per esprimere preferenza.



# Rappresentazione dei contenitori

*Gli autori della risorsa “<http://www.sito.com/doc2.html>” sono Andrea Verdi, Giovanni Bruni e Maria Celesti*



# Sintassi dei contenitori

*Gli autori della risorsa "http://www.sito.com/doc2.html" sono Andrea Verdi, Giovanni Bruni e Maria Celesti*

```
<rdf:Description about="http://www.sito.com/doc2.html">
  <fv:autore>
    <rdf:Bag>
      <rdf:li resource="mailto:verdi@sito.com"/>
      <rdf:li resource="mailto:bruni@sito.com"/>
      <rdf:li resource="mailto:celesti@sito.com"/>
    </rdf:Bag>
  </fv:autore>
</rdf:Description>
```

Analogamente si useranno <rdf:Seq> per le sequenze e <rdf:Alt> per le alternative.



# Referenti e referenti distributivi (1)

Quando si fa un'affermazione su una collezione, a cosa ci riferiamo: alla collezione o ai singoli elementi?

- ◆ Dire «*Mario Rossi ha scritto “<http://www.sito.com/doc.html>” e “<http://www.sito.com/doc2.html>”*» equivale a dire «*Mario Rossi ha scritto “<http://www.sito.com/doc.html>”*» e «*Mario Rossi ha scritto “<http://www.sito.com/doc2.html>”*».
- ◆ Tuttavia, dire «*La commissione composta da [Andrea Verdi](#), [Giovanni Bruni](#) e [Maria Celesti](#) ha approvato la pubblicazione di “<http://www.sito.com/doc2.html>”*» non è equivalente a dire che [Andrea Verdi](#), [Giovanni Bruni](#) e [Maria Celesti](#) hanno approvato la pubblicazione: è stata la commissione ad approvarlo, ed uno dei membri poteva senza dubbio aver votato contro.



# Referenti e referenti distributivi (2)

Il referente è l'indicazione della risorsa a cui ci riferiamo. RDF definisce tre referenti come attributi dell'elemento "Description":

- ◆ about è un referente globale, che si riferisce alla collezione intera, ma non ai singoli elementi.
- ◆ aboutEach è un referente distributivo, che si riferisce non alla collezione ma a ciascuno dei suoi membri.

```
<rdf:Bag ID="docs">  
  <rdf:li resource="http://www.sito.com/doc.html" />  
  <rdf:li resource="http:// www.sito.com/doc2.html" />  
</rdf:Bag>  
<rdf:Description aboutEach="#docs">  
  <fv:autore>Mario Rossi</fv:autore>  
</rdf:Description>
```

- ◆ aboutEachPrefix è un referente distributivo definito su un pattern (ad esempio su una directory di un sito Web).





# Reificazione ed indirezione (1)

Come è possibile fornire meta-informazioni su una meta-informazione? Ad esempio come posso esprimere la frase «*Ugo Bianchi dice che Mario Rossi è l'autore di "http://www.sito.com/doc.html"*»?

Questo in breve significa attribuire proprietà ad una affermazione, per la qual cosa occorre considerare la meta-informazione come una risorsa da descrivere.

Questa procedura si chiama *reificazione* (riduzione ad oggetto) della affermazione (o statement). Con una indirezione (un referente allo statement reificato) potrò allora esprimere statement su di esso.



# Reificazione ed indirezione (2)

Ogni statement, infatti, è equivalente ad un bag di tipo “Statement” in cui la risorsa è il *soggetto*, il valore è l’*oggetto*, e la proprietà è il *predicato*.

Quindi sono equivalenti:

```
<rdf:Description about="http://www.sito.com/doc.html">  
  <fv:autore>Mario Rossi</fv:autore>  
</rdf:Description>
```

e

```
<rdf:Description>  
  <rdf:subject resource="http://www.sito.com/doc.html"/>  
  <rdf:predicate resource="/schema/autore" />  
  <rdf:object>Mario Rossi</rdf:object>  
  <rdf:type resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement"/>  
</rdf:Description>
```



# Reificazione ed indirizzazione (3)

Uno Statement reificato può essere usato come risorsa destinatario di un'altra proprietà:

```
<rdf:Description>
  <rdf:subject resource="http://www.sito.com/doc.html" />
  <rdf:predicate resource="/schema/autore" />
  <rdf:object>Mario Rossi</rdf:object>
  <rdf:type resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement" />
  <fv:attribuitoA>Ugo Bianchi</fv:attribuitoA>
</rdf:Description>
```



# Sintassi compatta di reificazione

L'attributo "bagID" permette di considerare uno statement esplicito come se fosse reificato. La stessa description può dunque avere un "ID", utile per indicarla come fonte di ulteriori meta-informazioni, ed un "bagID" utile per esprimere meta-informazioni su di essa.

La frase precedente è equivalente al blocco:

```
<rdf:Description about="http://www.sito.com/doc.html"
  bagID="ST1">
  <fv:autore>Mario Rossi</fv:autore>
</rdf:Description>
```

```
<rdf:Description aboutEach="#ST1">
  <fv:attribuitoA>Ugo Bianchi</fv:attribuitoA>
</rdf:Description>
```



# Dublin Core (1)

Il Dublin Core è uno schema di meta-informazioni ideato per assegnare etichette ragionevoli alle risorse della rete.

Si chiama Dublin Core perché è considerato il nucleo (core) delle meta-informazioni interessanti per qualunque risorsa, e perché è nato da un'iniziativa di bibliotecari, archivisti, fornitori di contenuto e esperti di markup svoltasi nel 1995 a Dublino.

Dublin Core è ovviamente indipendente da qualunque sintassi, ma ben si adatta a tutte, ed in particolare a RDF.

Dublin Core versione 1 ha introdotto esattamente quindici categorie di meta-informazioni utili per la catalogazione di risorse di rete. La versione 2 non ha modificato le dieci categorie, ma ha aggiunto un meccanismo di sottoclassi (detti *qualificatori*) delle categorie, ed ha introdotto un elenco iniziale di qualificatori.



# Dublin Core (2)

Le quindici categorie descrivono meta-informazioni di tre tipi:

## Contenuto

Title  
Subject  
Description  
Type  
Source  
Relation  
Coverage

## Proprietà intellettuale

Creator  
Publisher  
Contributor  
Rights

## Istanza

Date  
Format  
Identifier  
Language

I qualificatori permettono di specificare ulteriormente informazioni di queste categorie, secondo questi criteri:

- ◆ **Raffinamento dello schema:** fornisce alcuni significati più precisi sui termini. Ad esempio, “Date” ha come qualificatori: “created”, “valid”, “available”, “issued”, “modified”)
- ◆ **Supporto per codifiche specifiche:** permette di usare i valori di specifiche codifiche all’interno del Dublin Core. Ad esempio, “Subject” ha come qualificatori: “LCSH” (Library of Congress Subject Headings), “MeSH” (Medical Subject Headings), “DDC” (Dewey Decimal Classification), ecc.



# Un esempio di Dublin Core in RDF

```
<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:dc="http://purl.org/metadata/dublin_core#">
  <Description about="http://www.dlib.org">
    <dc:Title>D-Lib Program</dc:Title>
    <dc:Description>The D-Lib program supports the
community of people with research interests in digital
libraries and electronic publishing.</dc:Description>
    <dc:Publisher>Corporation For National Research
Initiatives</dc:Publisher>
    <dc>Date>1995-01-07</dc>Date>
    <dc:Subject><Bag>
      <li>Research; statistical methods</li>
      <li>Education, research, related topics</li>
      <li>Library use Studies</li>
    </Bag></dc:Subject>
    <dc:Type>World Wide Web Home Page</dc:Type>
    <dc:Format>text/html</dc:Format>
    <dc:Language>en</dc:Language>
  </Description>
</RDF>
```



# Conclusioni

Oggi abbiamo parlato di

- ◆ L'esigenza di meta-informazioni
- ◆ PICS
- ◆ Il modello dei dati di RDF
- ◆ Aspetti della sintassi di RDF
- ◆ Il Dublin Core





# Riferimenti

- Ora Lassila, Ralph R. Swick, *Resource Description Framework (RDF), Model and Syntax Specification*, W3C Recommendation 22 February 1999, <http://www.w3.org/TR/REC-rdf-syntax>
- Dan Brickley, R.V. Guha, *Resource Description Framework (RDF) Schema Specification 1.0*, W3C Candidate Recommendation 27 March 2000, <http://www.w3.org/TR/rdf-schema>
- Ora Lassila, *Introduction to RDF Metadata*, W3C NOTE 1997-11-13, <http://www.w3.org/TR/NOTE-rdf-simple-intro-971113.html>
- Tim Berners-Lee, Dan Connolly, Ralph R. Swick, *Web Architecture: Describing and Exchanging Data*, W3C Note 7 June 1999, <http://www.w3.org/1999/04/WebData>
- Eric Miller, *An Introduction to the Resource Description Framework*, D-Lib Magazine, May 1998, <http://www.dlib.org/dlib/may98/miller/05miller.html>
- S. Weibel, C. Lagoze, M. Wolf, *Dublin Core Metadata for Resource Discovery*, RFC 2413, September 1998

