Summer School LDA

Libraries in the digital age: linked data technologies for a global knowledge sharing

Pula (Cagliari), 29 agosto - 1° settembre

Introduzione al Semantic Web

Oreste Signore

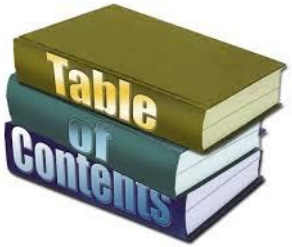
(W3C Italy)

Slide a: http://www.orestesignore.eu/education/lda/slides/lda1.pdf

Semantic Web

W3C

WORLD WIDE WEB consortium Ufficio Italiano

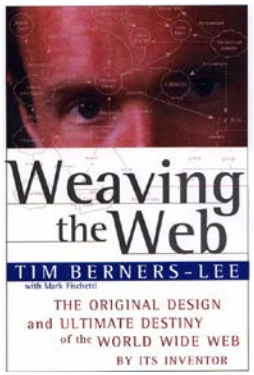


Ringraziamenti

- Questa presentazione è basata in gran parte sul materiale di presentazioni tenute da Ivan Herman, già W3C Semantic Web Activity Lead
- Il materiale di questa presentazione può essere riutilizzato nel rispetto delle leggi sul copyright e delle regole del W3C
- Un particolare ringraziamento agli organizzatori di Summer School LDA per avermi invitato a tenere questo seminario

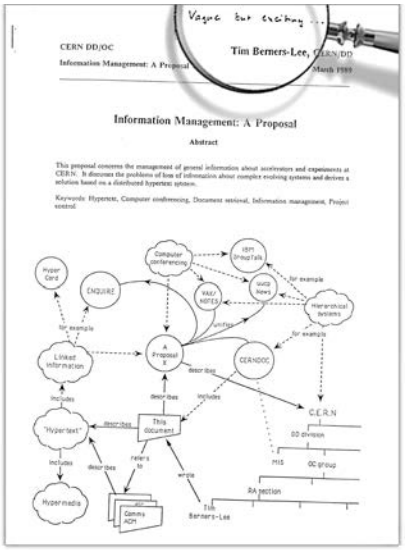
Contenuto

- Integrazione dei dati
  - ✓ Un esempio semplice
  - ✓ Merging e URI
- Semantic Web
  - ✓ Architettura
  - ✓ Metadati
- RDF
- Conclusioni



C'era una volta…

- 1970(?) Un ragazzo che parlava con il padre:
  - ✓ How to make a computer intuitive, able to complete connections as the brain did
- 1980, al CERN:
  - ✓ *Suppose all the information stored on computers everywhere were linked. Suppose I could program my computer to create a space in which anything could be linked to anything… There would be a single, global information space.*
- 1989 Vague but exiciting
- …e il Web fu …
- 1994
  - ✓ "The very first *International World Wide Web Conference*, at CERN, Geneva, Switzerland, in September 1994" http://www.w3.org/Talks/WWW94Tim/
- 1999 Semantic Web Activity nel W3C (ora: Data Activity)
- 2007 LOD (W3C Linking Open Data project)



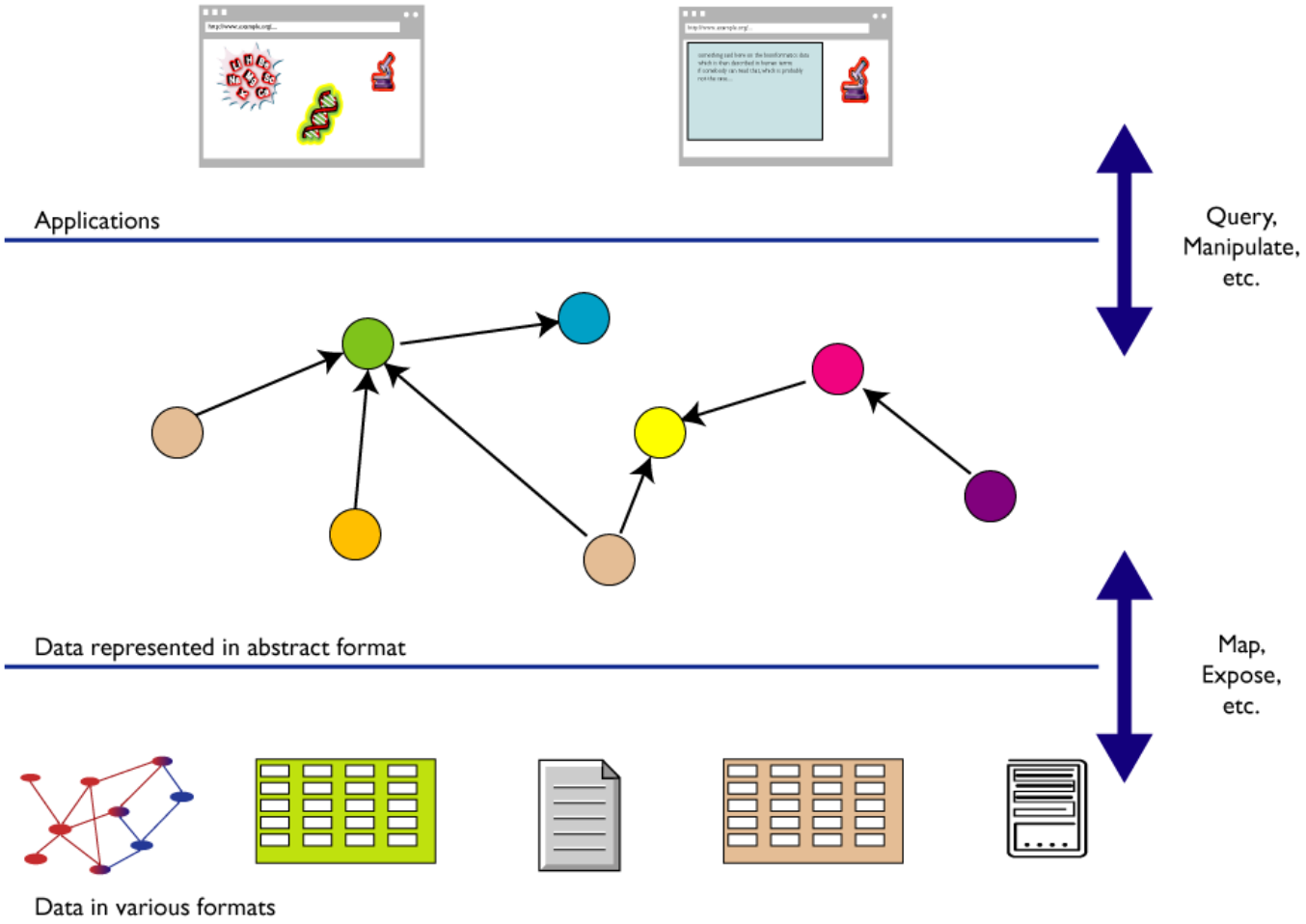
I limiti del Web attuale

- **Nel *web tradizionale* si rappresenta l' informazione utilizzando:**
  - ✓ **linguaggio naturale**
  - ✓ **grafica, elementi multimediali, struttura della pagina**
- **Spesso è necessario *combinare le informazioni* (provenienti da fonti diverse)**
- ***Per gli esseri umani è facile ...***
  - ✓ **dedurre fatti da informazioni incomplete**
  - ✓ **creare e seguire associazioni mentali**
  - ✓ **provare varie esperienze sensoriali**
  - ✓ **aggregare le informazioni indipendentemente dalle tecnologie utilizzate**
- **... ma *le macchine non sono intelligenti*!**
  - ✓ **non possono utilizzare informazioni parziali**
  - ✓ **hanno difficoltà ad aggregare informazioni strutturate in forma diversa**

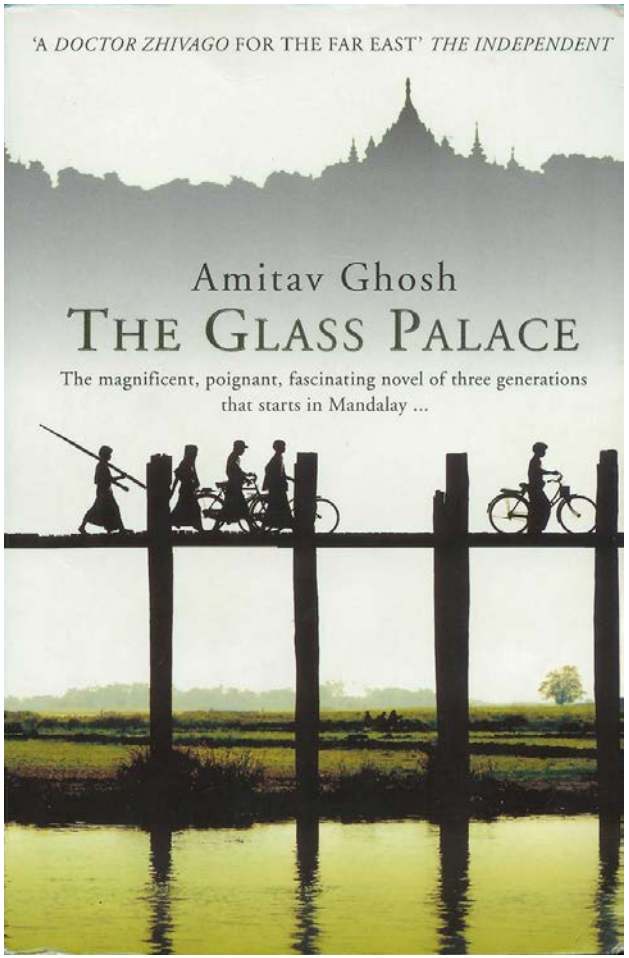
Data integration (visione semplificata)

- **Mappare i vari dati su una rappresentazione astratta dei dati**
  - ✓ **rendere i dati indipendenti dalla loro rappresentazione interna…**
- **Merge delle rappresentazioni risultanti**
- **Cominciamo a formulare query sul complesso risultante!**
  - ✓ **query che non sarebbe stato possibile eseguire sui singoli dataset**

L'integrazione dei dati



Un libro (nella biblioteca "A")



Una versione semplificata di una biblioteca (Dataset "A")

- **Tabella Book**

ID	Author	Title	Publisher	Year
ISBN 0-00-651409-X	id_xyz	The Glass Palace	Id_qpr	2000

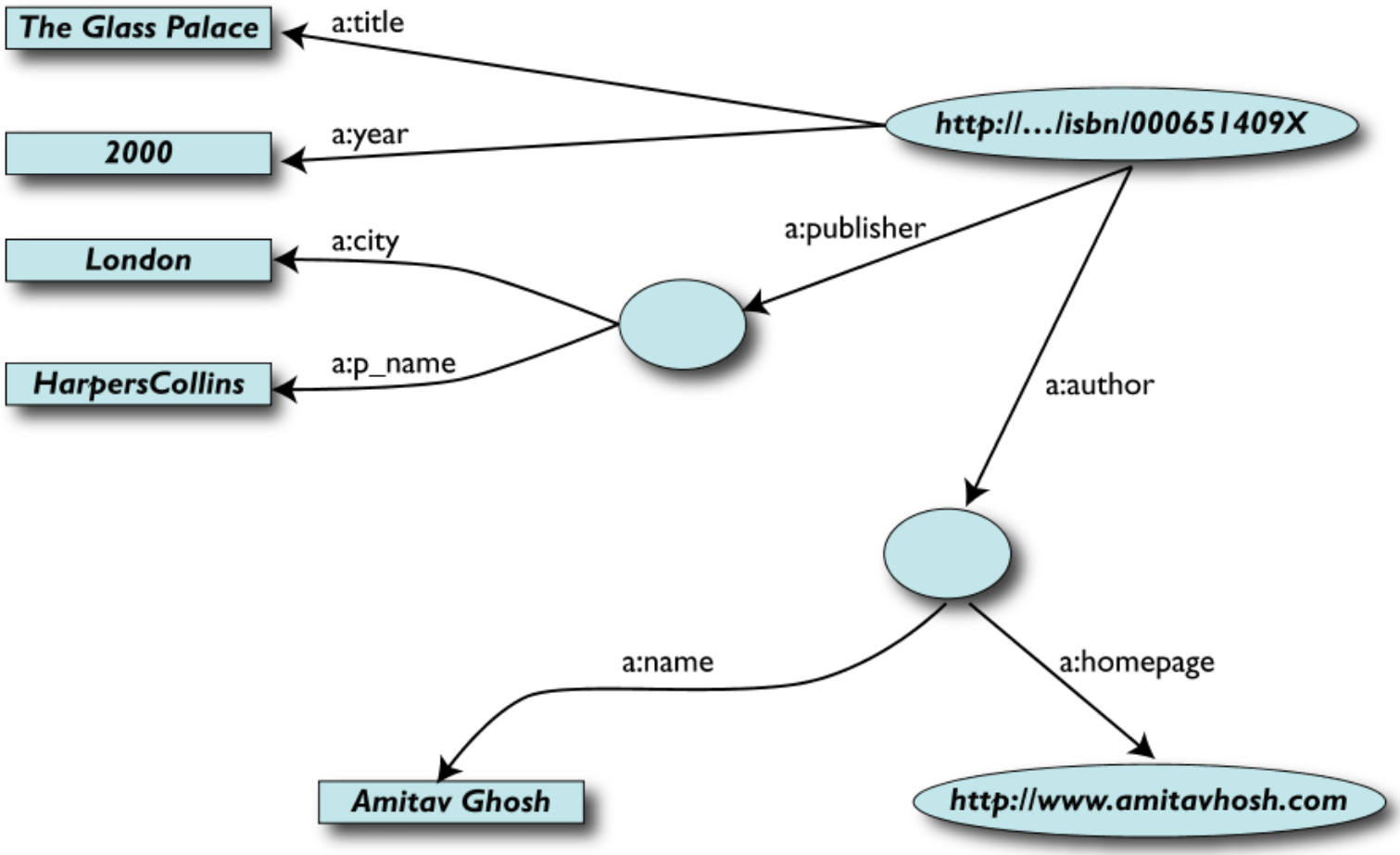
- **Tabella Author**

ID	Name	HomePage
id_xyz	Amitav Ghosh	http://www.amitavghosh.com

- **Tabella Publisher**

ID	PublisherName	City
id_qpr	Harper Collins	London

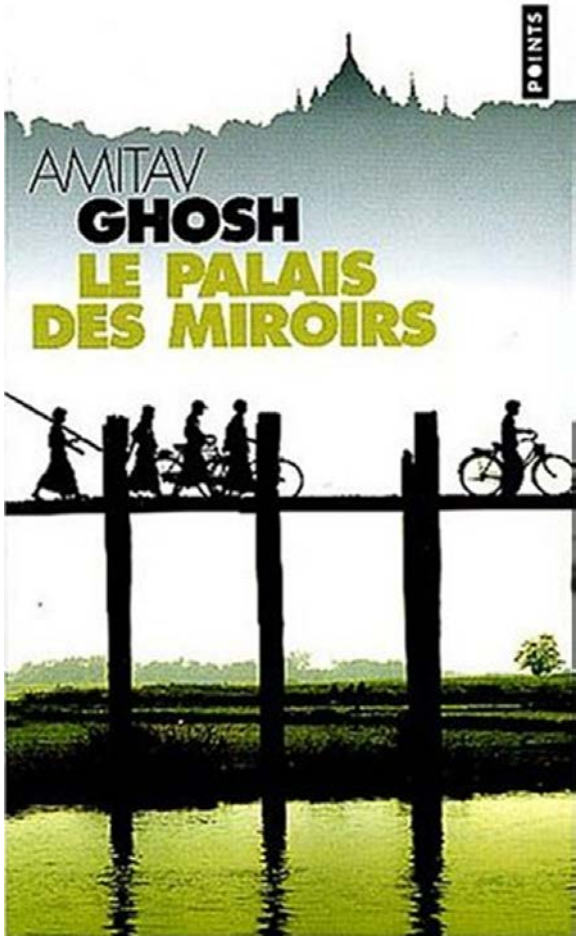
Rappresentata come grafo…



Alcune osservazioni sull'esportazione dei dati

- **Le relazioni formano un *grafo***
  - ✓ **i nodi individuano dati "reali" o contengono caratteri ("literal")**
  - ✓ **è inessenziale il modo in cui i grafi sono rappresentati nella macchina**
- **L' esportazione dei dati *non* comporta necessariamente una trasformazione fisica**
  - ✓ **le relazioni possono essere generate dinamicamente al momento della richiesta**
    - **con SQL "bridges"**
    - **scraping di pagine HTML**
    - **estrazione di dati da fogli Excel**
    - **etc.**
- **L' esportazione dei dati può essere *parziale***

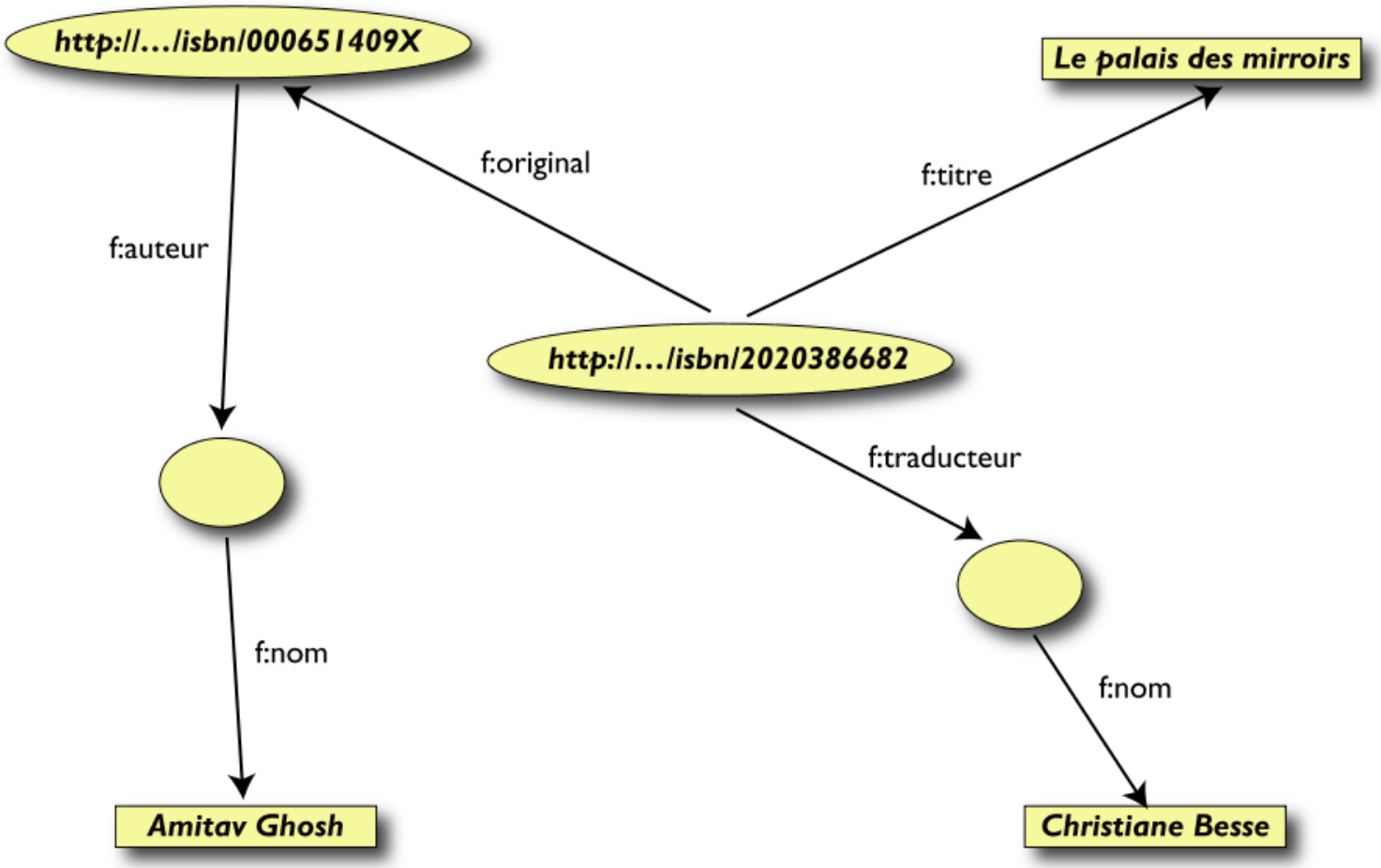
Un libro (nella biblioteca "F")



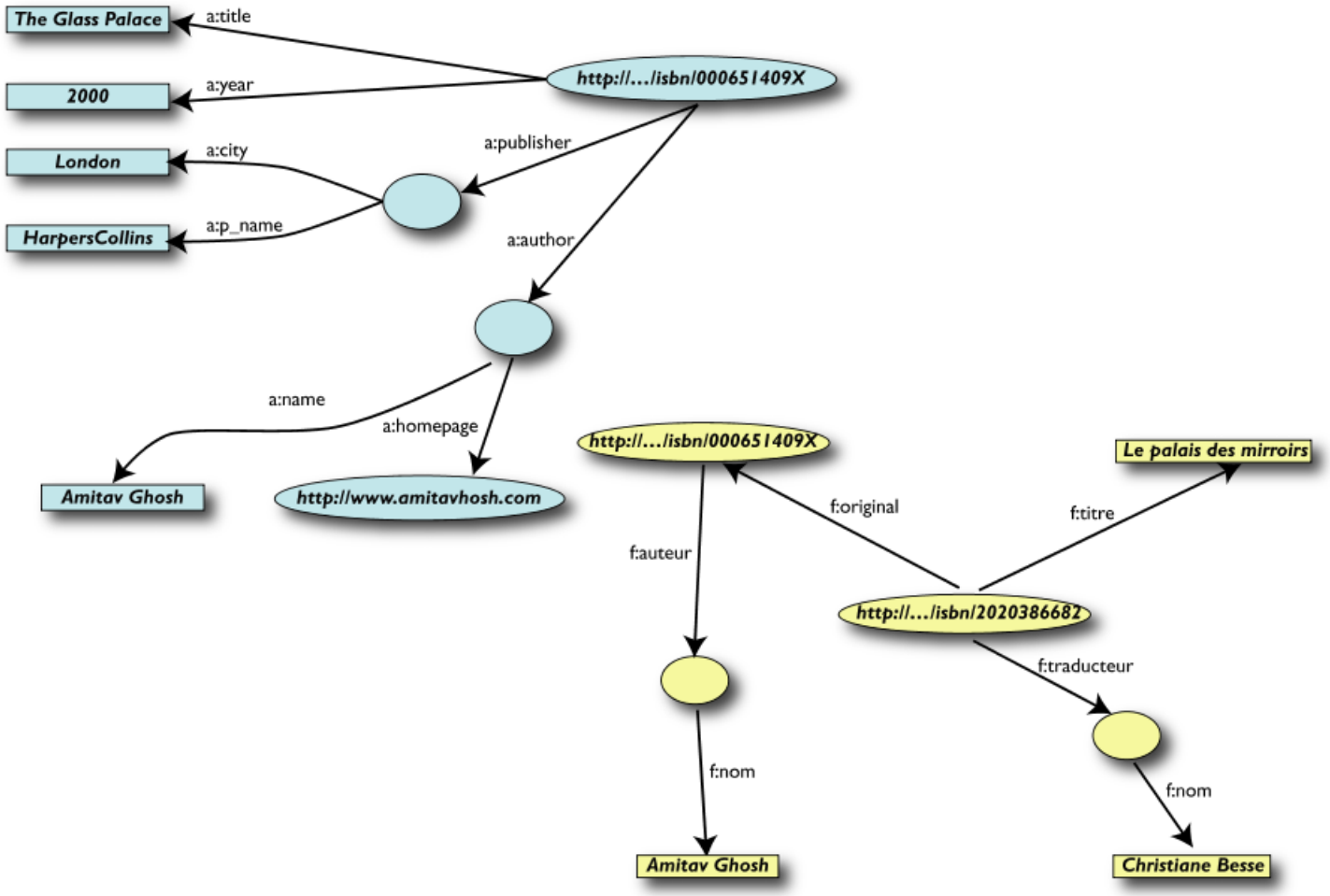
Un'altra biblioteca (dataset "F")

	A	B	C	D
1	**ID**	**Titre**	**Traducteur**	**Original**
2	ISBN 2020286682	Le Palais des Miroirs	$A12$	ISBN 0-00-6511409-X
3				
4				
5				
6	**ID**	**Auteur**		
7	ISBN 0-00-6511409-X	$A11$		
8				
9				
10	**Nom**			
11	Ghosh, Amitav			
12	Besse, Christianne			

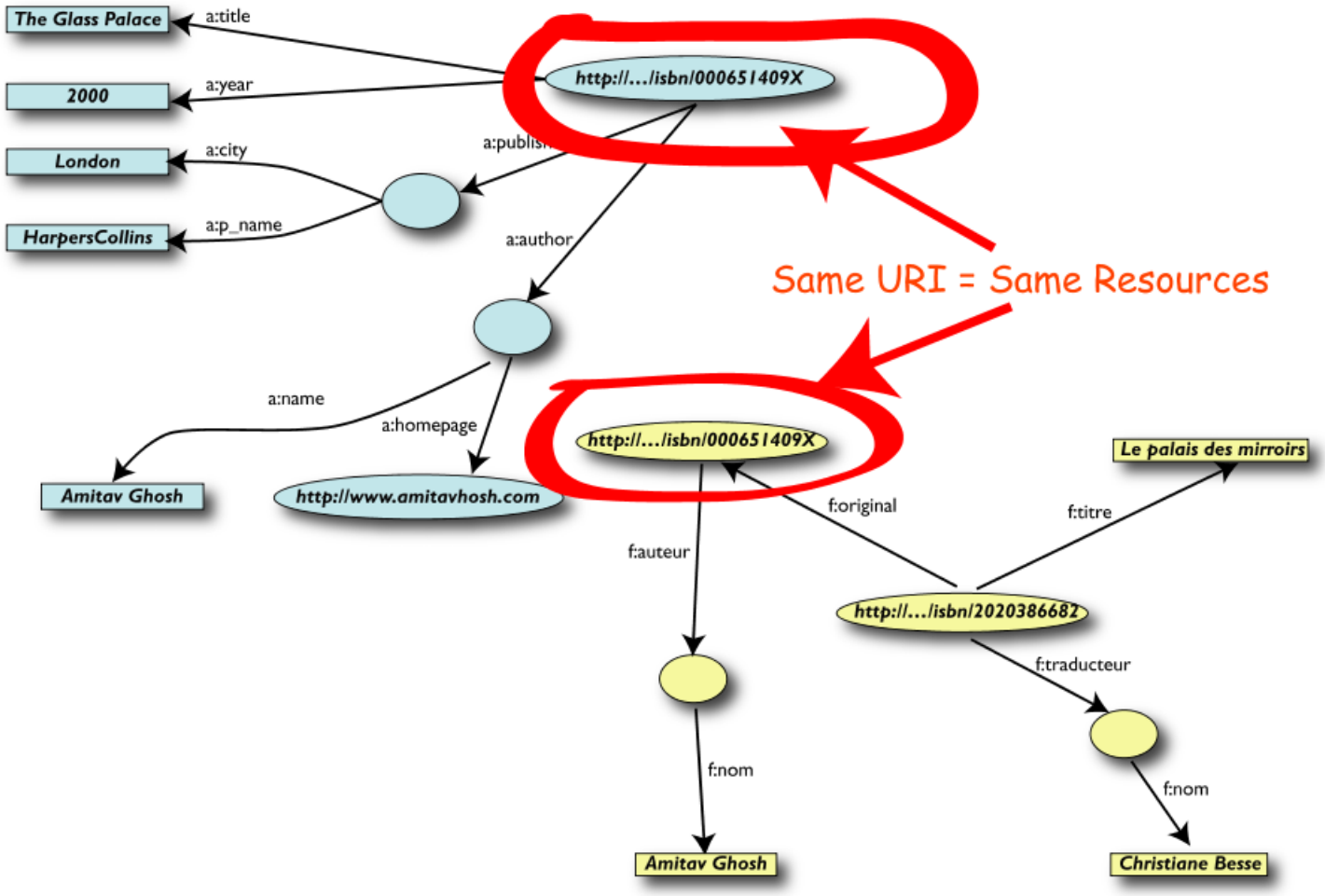
Secondo passo: esportare il secondo insieme di dati



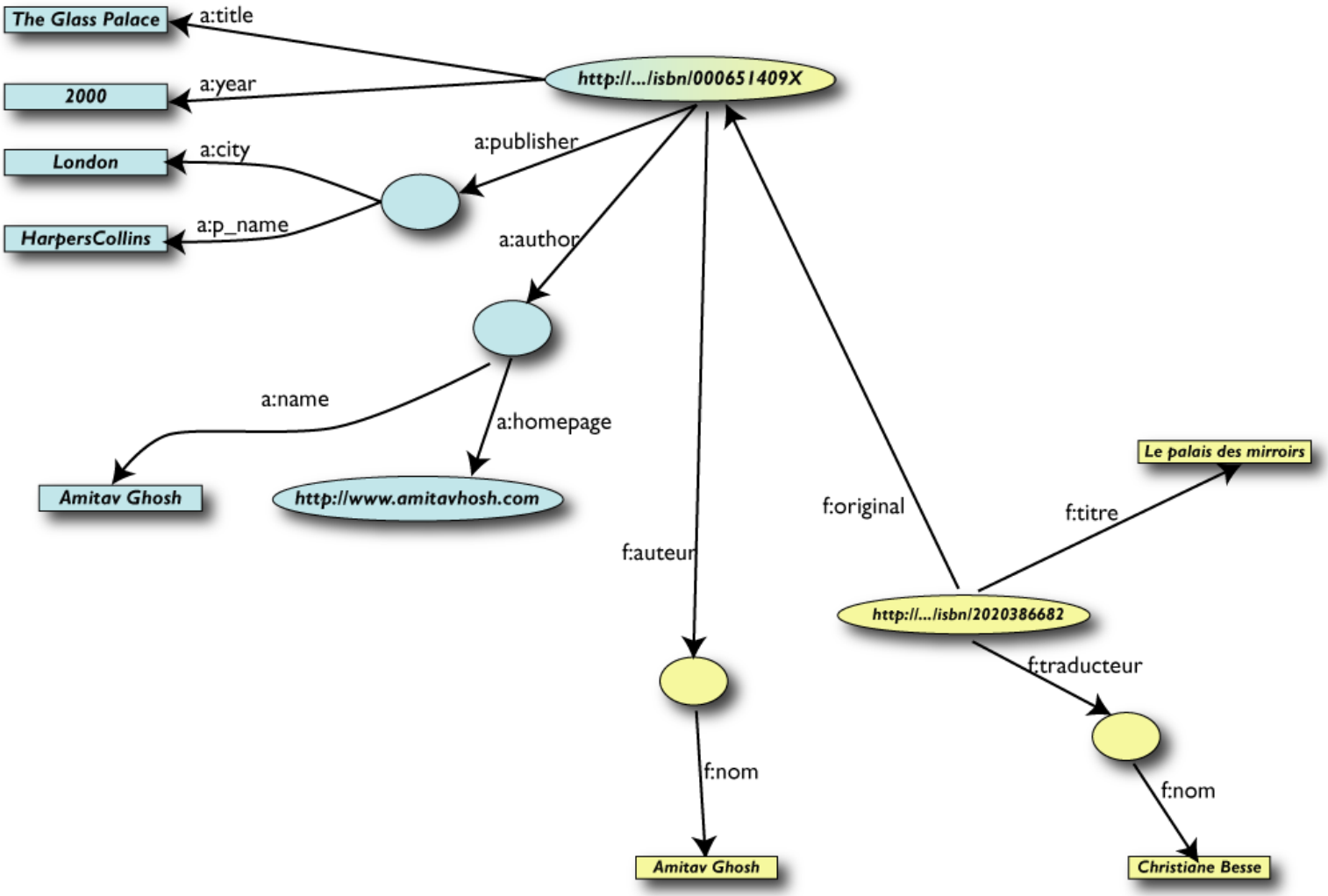
Terzo passo:merging dei dati



Terzo passo:merging dei dati

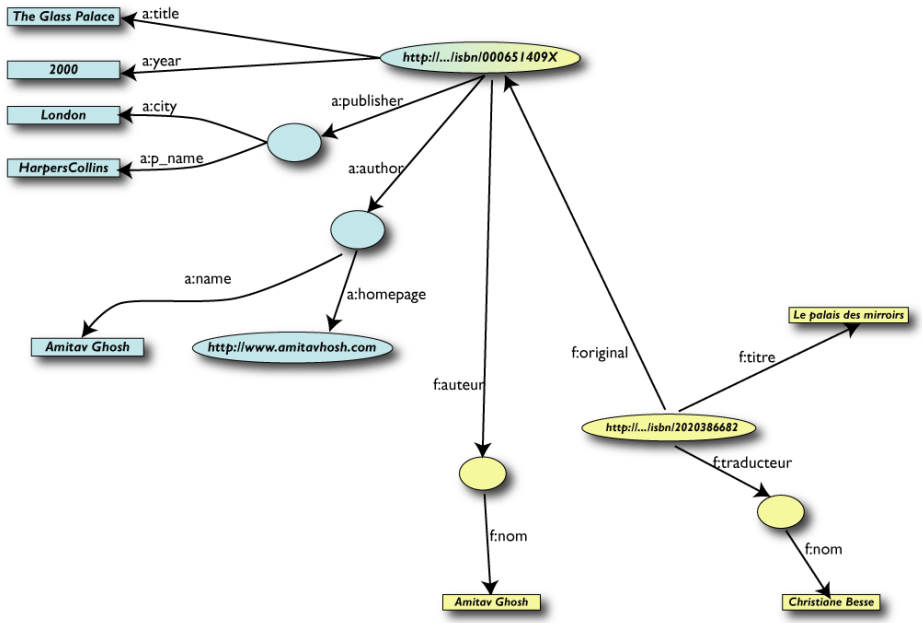


Merging delle risorse identiche



Le Query più semplici…

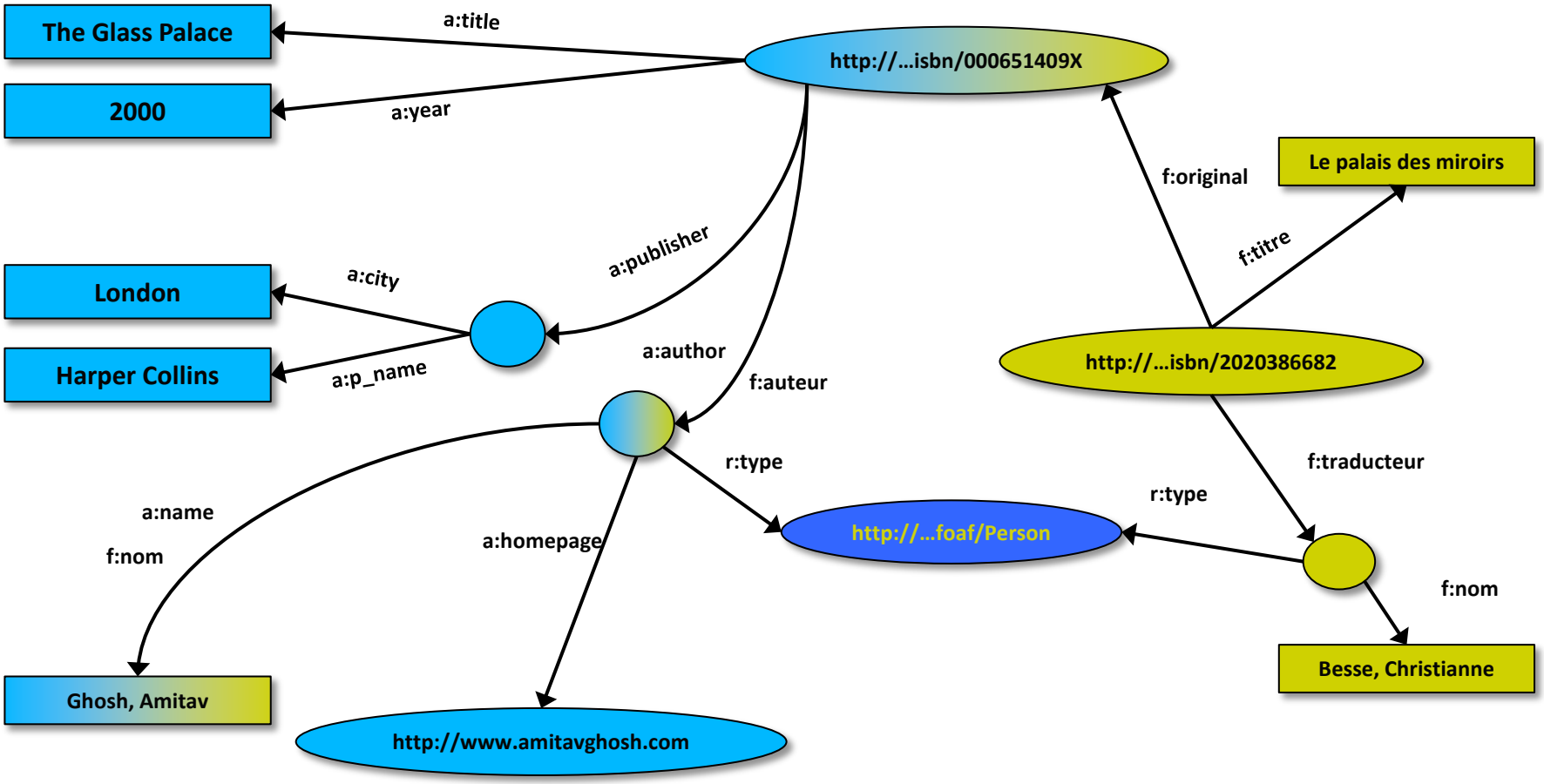
- L' utente dei dati "F" può ora formulare query del tipo: "*donnes-moi le titre de l'original*" o "*give me the title of the original*"
- Questa informazione non è nel dataset "F"…
- …ma può essere ritrovata grazie al merging con il dataset "A"!



…ma si può avere di più…

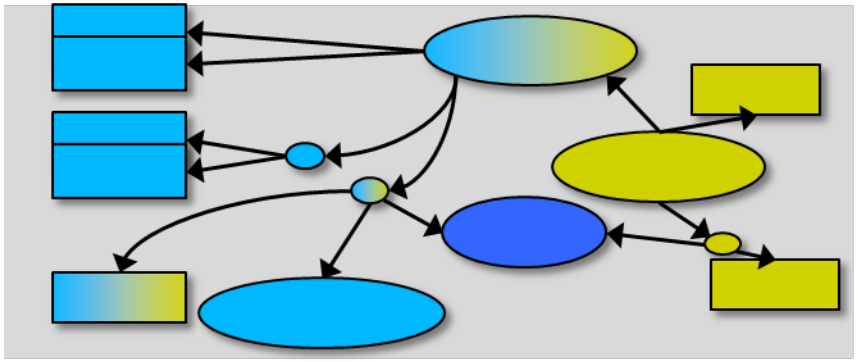
- Abbiamo la "*sensazione*" che a:author e f:auteur siano la stessa cosa
- Ma un *processo automatico* non se ne può accorgere!
- Aggiungiamo un po' di *informazione addizionale* ai dati combinati:
  - ✓ a:author same as f:auteur
  - ✓ entrambi identificano una "*Person*", che è un termine che una comunità può aver già definito:
    - una "*Person*" è definita univocamente dal suo nome e email, o codice fiscale
    - può essere usato come "*categoria*" per certi tipi di risorse
- e si può utilizzare la conoscenza extra *unendo altri grafi*…

Usiamo la conoscenza "extra"



Query più "ricche"

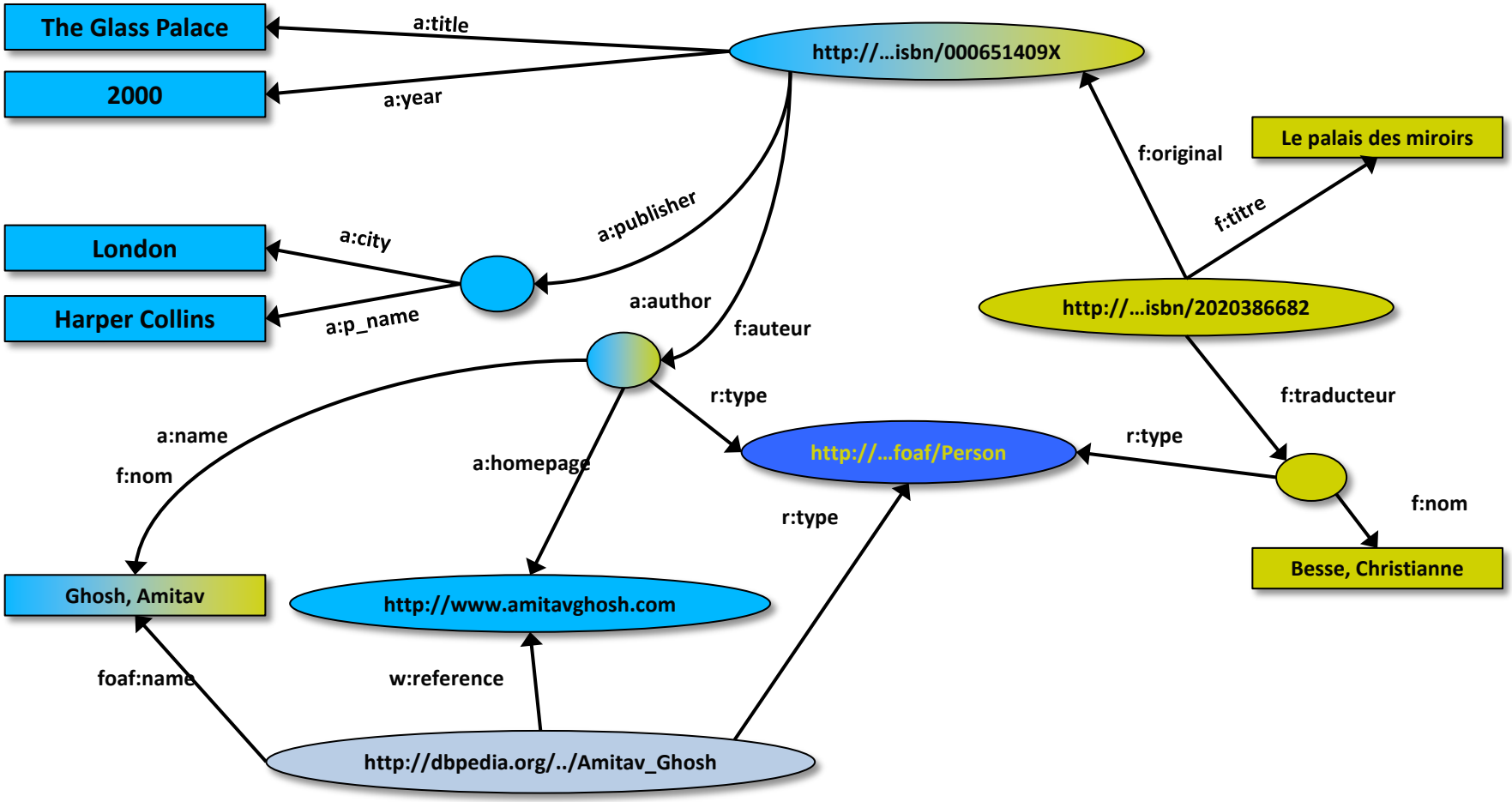
- **L'utente del dataset "F" può ora chiedere:**
  - ✓ **"donnes-moi la page d'accueil de l'auteur de l'original"**
    - **cioè… "dammi la homepage dell'autore dell'orginale**
- **L'informazione non è né nel dataset "A" né nel dataset "…**
- **…ma è stata resa accessibilegazie al processo di:**
  - ✓ **merge dei dataset "A" e "F"**
  - ✓ **aggiunta di tre semplici statement che fungono da "collante"**



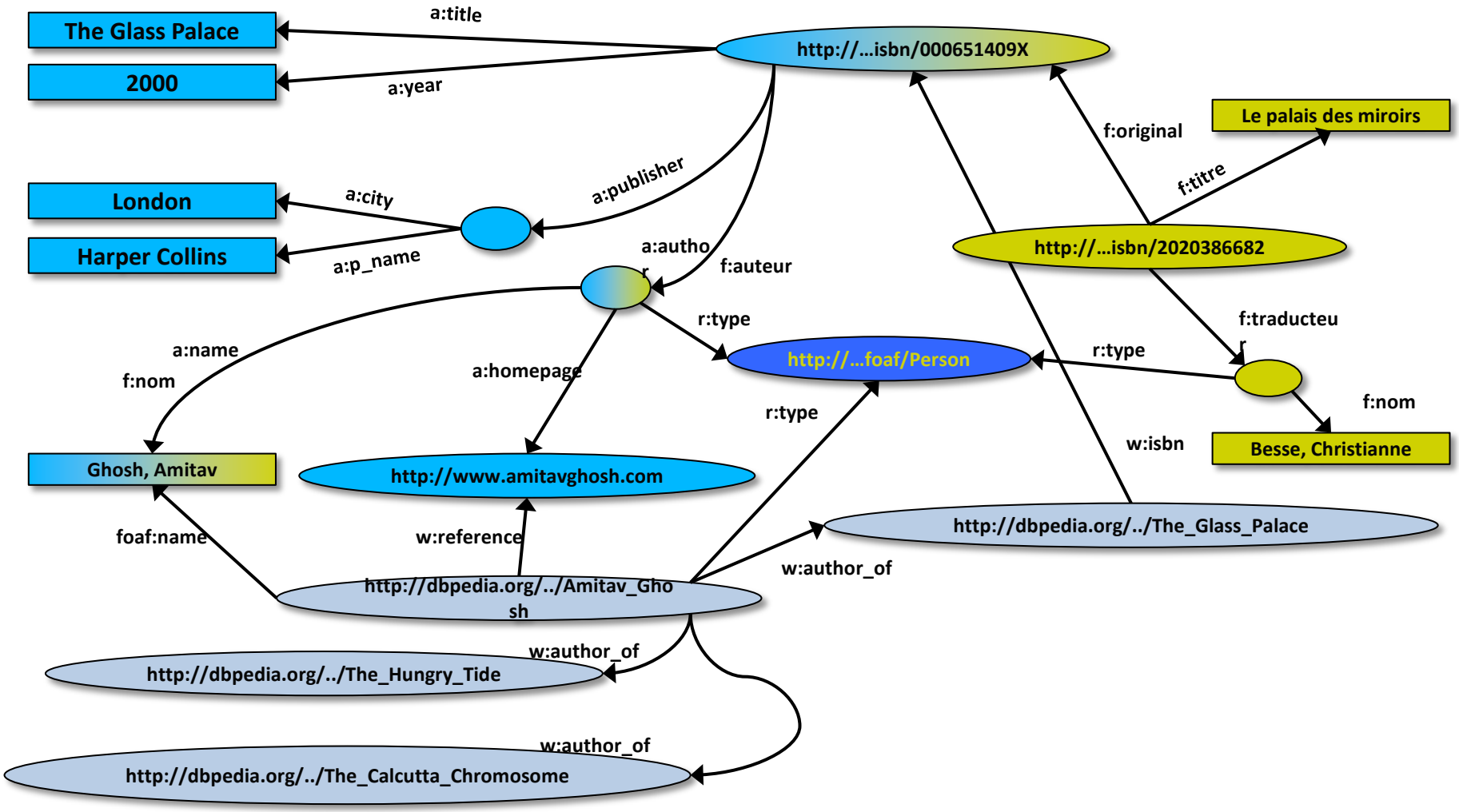
Combiniamo altri dataset

- **Per esempio, usando “Person”, il dataset può essere combinato con altre sorgenti di dati**
- **Per esempio, è possibile estrarre i dati contenuti in Wikipedia utilizzando alcuni tool, specifici**
  - ✓ **il progetto “dbpedia” può già estrarre l’informazione “infobox” da Wikipedia**

Combiniamo con i dati di Wikipedia



Combiniamo con i dati di Wikipedia



Sorpresi?

- **Forse, ma in realtà no**
- **È esattamente quanto avviene normalmente a tutti gli utenti del Web (ma in questo caso grazie a un processo *automatico*)**
- **La *differenza*: è necessario un po' più di rigore (per es. *dare un nome* alle associazioni) perché le macchine possano riuscirci**

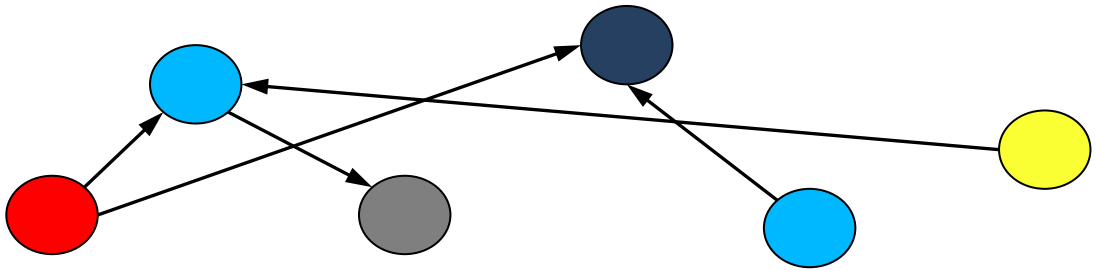
In realtà…

- **Abbiamo combinato dataset diversi**
  - ✓ **ognuno di essi può provenire da un *qualunque sito web***
  - ✓ **possono avere originariamente *formati differenti* (MySQL, fogli excel, XHTML, etc)**
  - ✓ **possono avere *nomi diversi per le relazioni* (multilinguismo)**
- **Li abbiamo potuti combinare perché avevano *lo stesso URI* (l' ISBN nell' esempio)**
- **Possiamo aggiungere *conoscenza addizionale*, utilizzando terminologie comuni definite dalle varie comunità**
- **Di conseguenza, è stato possibile identificare e utilizzare *nuove relazioni***

Cosa abbiamo fatto?

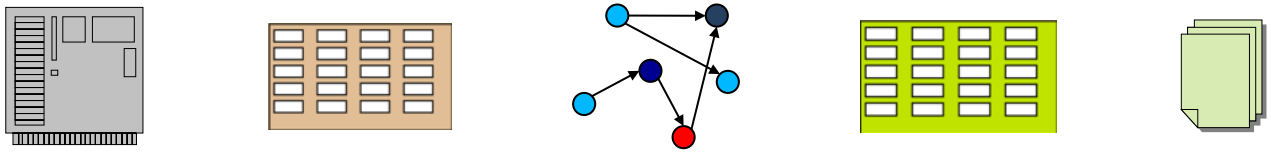
Applications

Manipulate
Query
…



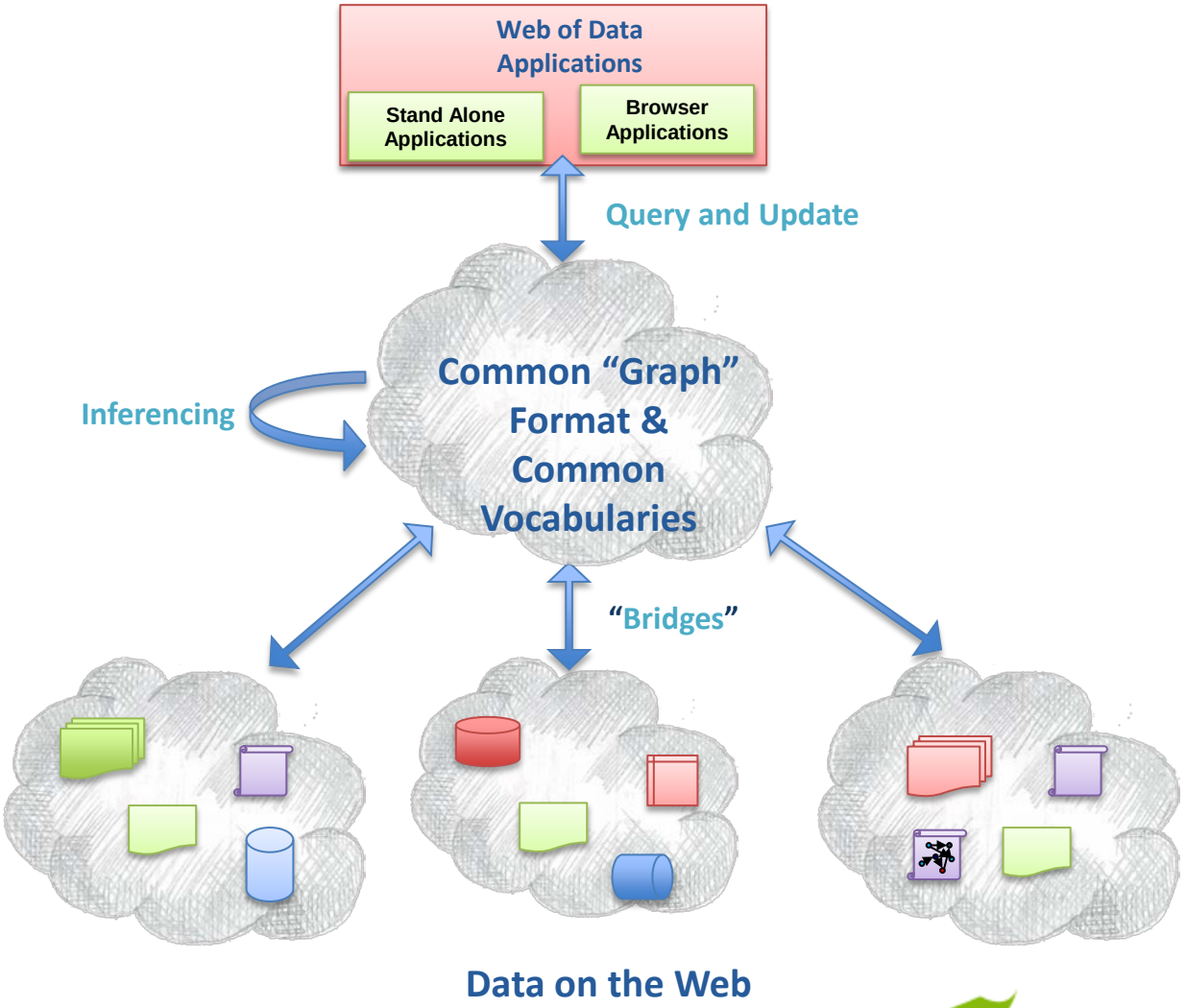
Data represented in abstract format

Map,
Expose,
…



Data in various formats

Cosa abbiamo fatto? (un'altra visione)



I vantaggi del processo di astrazione…

- … la rappresentazione come grafo è indipendente dalle struttura reali sottostanti
- … modifiche a database schema, strutture XHTML, etc. non hanno impatto sul complesso:
  - ✓ “schema independence”
- … possono essere aggiunti, senza soluzione di continuità, nuovi dati o nuove connessioni
  - ✓ flessibilità, espandibilità

"Effetto network"

- Esteso ai dati sul Web
- Mediante gli URI possiamo collegare **qualunque** dato a **qualunque** altro

E il processo può essere anche più ricco

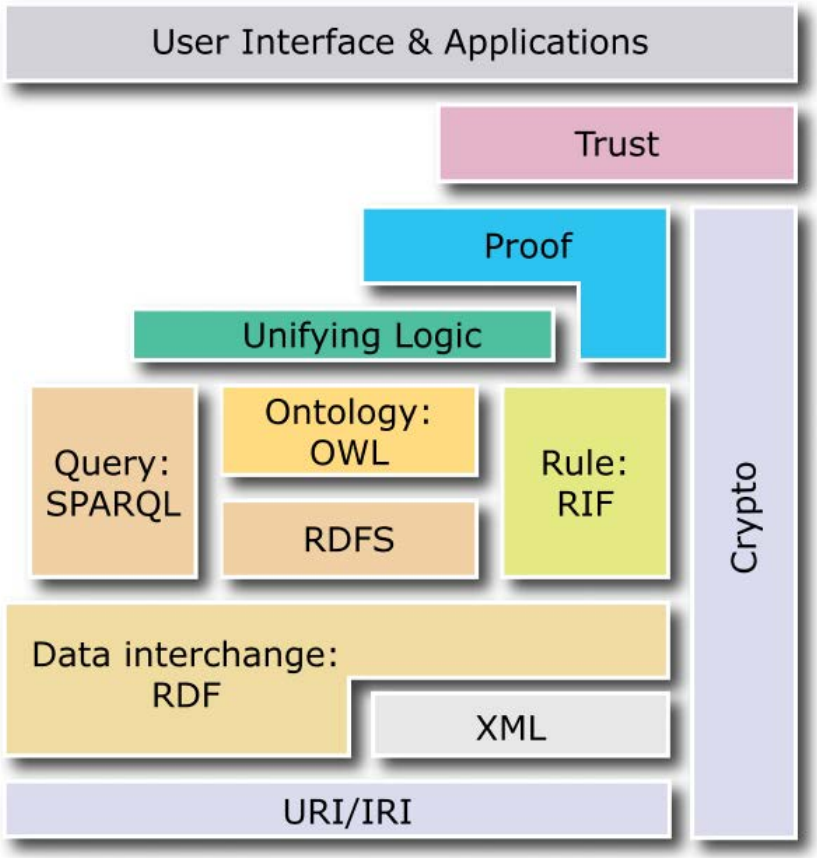
- La conoscenza addizionale può essere anche molto complessa
- È qui che entrano in gioco le *ontologie*, le *regole*, etc.
- Il processo di *astrazione* è vantaggioso perché la rappresentazione come *grafo* è indipendente dalle strutture dati sottostanti

E il Semantic Web?

- **Il Semantic Web fornisce le tecnologie che rendono possibile questa integrazione!**
  - ✓ **lo scopo di questo tutorial è appunto quello di fornire unquadro complessivo…**

L'architettura del Semantic Web

- **Il Semantic Web ...**
  - ✓ **è un' infrastruttura basata su metadati per poter svolgere *ragionamenti* sul Web**
  - ✓ ***estende*, non sostituisce il web attuale**
- **I metadati sono:**
  - ✓ **Informazioni, elaborabili automaticamente (*machine understandable*), relative a una risorsa web o a qualche altra cosa**
  - ✓ **... *data about data***
  - ✓ **... informazioni che possono essere utilizzate da *intelligent software agents* per fare un uso appropriato delle risorse**
  - ✓ **... *dati* ...**
  - ✓ **... che possono essere *descritti da altri metadati* ...**



Perché i metadati?

- **Nel Web di oggi tutte le informazioni sono "*machine readable*"**
- **Nel Semantic Web le informazioni devono essere "*machine understandable*". Quindi occorrono:**
  - ✓ ***nomi non ambigui* per le risorse (URI)**
  - ✓ **un *data model condiviso* per esprimere i metadati (RDF)**
  - ✓ **un modo per *accedere ai metadati* sul Web**
  - ✓ ***vocabolari* condivisi (*ontologie*)**

Architettura del Web

- Decentralizzazione
- Gli elementi fondamentali
  - ✓ URI
    - L'innovazione più fondamentale del Web
    - Possono identificare qualunque cosa (risorse, concetti)
  - ✓ HTTP
    - Format negotiation
    - Protocollo per recuperare le risorse (fetch resources)
  - ✓ HTML
    - Strutturazione dei documenti
- RDF (Resource Description Framework)
  - ✓ è per il Semantic Web ciò che HTML è stato per il Web

Cosa è RDF?

- **L' uso efficace dei metadati richiede la definizione di convenzioni per:**
  - ✓ ***semantica*** **(definita dalle singole comunità disciplinari)**
  - ✓ ***sintassi*** **(organizzazione dei data element per l' elaborazione automatica)**
  - ✓ ***struttura*** **(vincolo formale sulla sintassi)**
- **RDF:**
  - ✓ ***Resource Description Framework***
  - ✓ **strumento base per *codifica*, *scambio* e *riutilizzo* di metadati strutturati**
  - ✓ **consente l' *interoperabilità* tra applicazioni che si scambiano sul Web informazioni *machine-understandable***

RDF è per il Semantic Web ciò che HTML è stato per il web

Triple RDF

- **Formalizzando quanto abbiamo fatto:**
  - ✓ **Abbiamo "collegato" i dati...**
  - ✓ **Ma il semplice collegamento non è sufficiente:**
    - **occorre assegnare un nome ai dati**
  - ✓ **Ecco come nascono le triple RDF:**
    - **Una connessione etichettata tra due risorse**

Triple RDF (cont.)

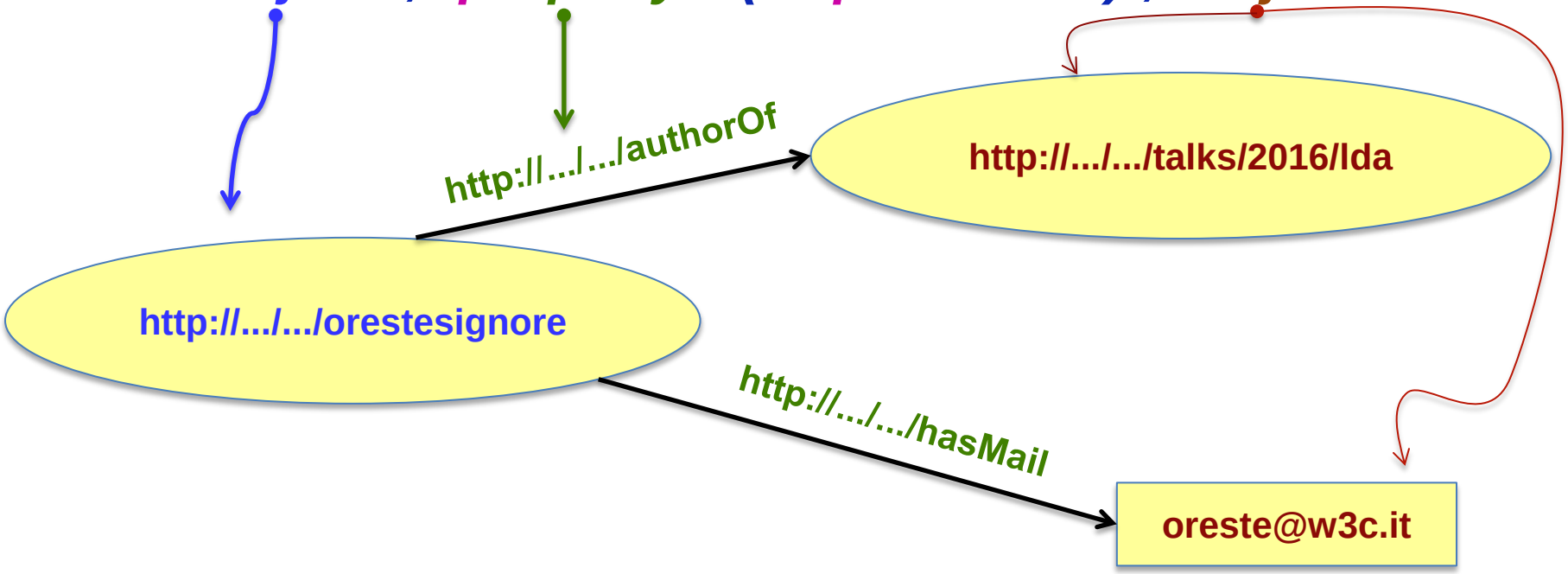
- **Le risorse possono usare un qualunque URI**
  - ✓ `http://www.example.org/file.html#home`
  - ✓ `http://www.example.org/f.xml#xpath(//q[@a=b])`
  - ✓ `http://www.example.org/form?a=b&c=d`
- **Le triple RDF costituiscono un grafo diretto (o grafo orientato) etichettato (directed, labeled graph)**
  - ✓ **(è questo il modo migliore di considerarle!)**

Triple RDf (cont.)

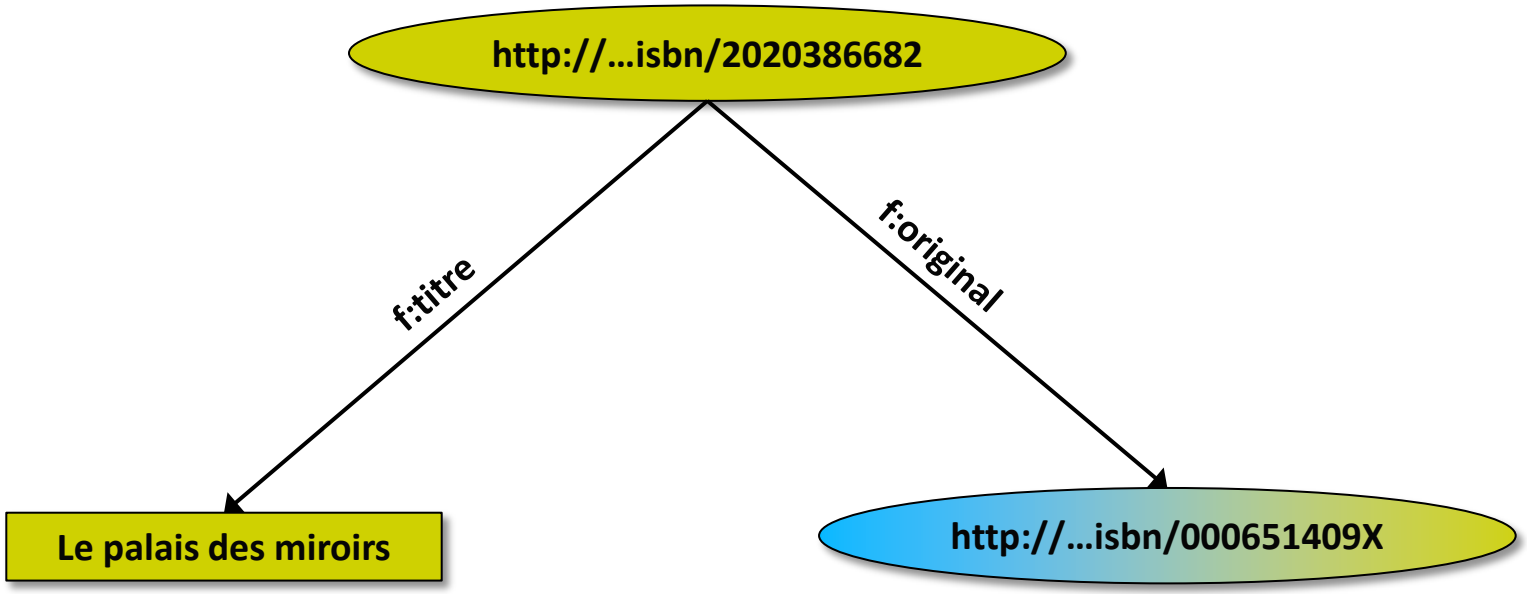
- **Una tripla RDF (s,p,o) è definita in modo che:**
  - ✓ **"s", "p" sono URI, cioè risorse sul Web; "o" è un URI o un "literal"**
  - ✓ **dal punto di vista concettuale: "p" *collega*, o *mette in relazione* "s" e "o"**
  - ✓ **si noti che vengono utilizzati URI per denotare i nomi: per esempio, possiamo utilizzare http://www.example.org/original**
  - ✓ **ecco la codifica completa della tripla:**
    `<http://…isbn 6682>, <http://…/original>, <http://…isbn 409X>`
- ***RDF* è un modello generale per queste triple (con un formato machine readable come RDF/XML, Turtle, n3, RXR)**
  - ✓ ***ed è tutto qui!* (semplice, dopo tutto)**

RDF in due parole

- **Una tripla RDF (s,p,o)**
  - ✓ *"subject", "property" (o "predicate") , "object"*



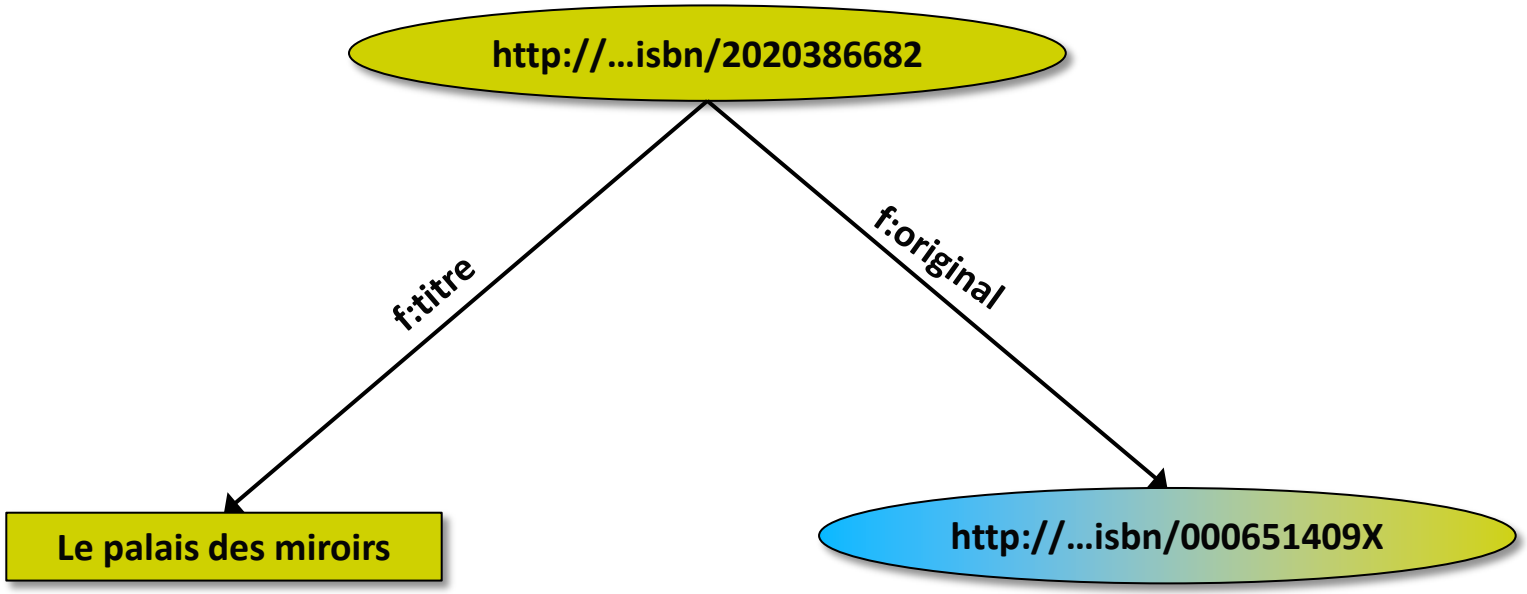
Un esempio semplice (in RDF/XML)




```
<rdf:Description rdf:about="http://…/isbn/2020386682">
    <f:titre xml:lang="fr">Le palais des mirroirs</f:titre>
    <f:original rdf:resource="http://…/isbn/000651409X"/>
</rdf:Description>
```

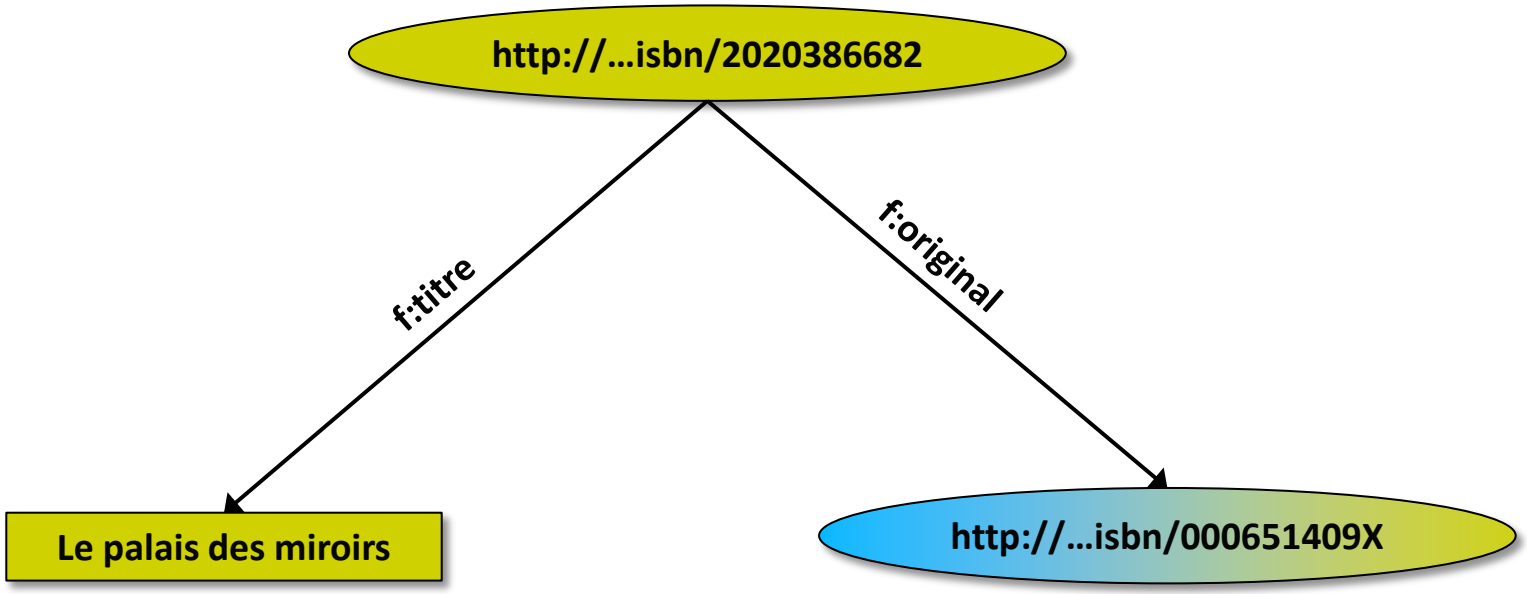
(Nota: per semplificare gli URI sono stati usati I namespace)

Un esempio semplice(in Turtle)




```
<http://…/isbn/2020386682>
    f:titre "Le palais des miroirs"@fr ;
    f:original <http://…/isbn/000651409X> .
```

Un esempio semplice (in RDFa)




```
<p about="http://…/isbn/2020386682">The book entitled
“<span property="f:title" lang="fr">Le palais des mirroirs</span>”
is the French translation of the
“<span rel="f:original" resource="http://…/isbn/000651409X">Glass
Palace</span>”</p> .
```

Quale sintassi?

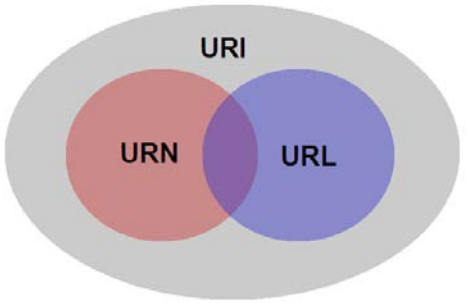
- La sintassi (RDF/XML, Turtle) è semplicemente *sintassi*
- La cosa importante sono il *modello* sottostante e i *concetti*
- Non tratteremo in dettaglio gli aspetti sintattici (abbiamo comunque già visto alcuni esempi in Turtle e in RDF/XML)
  - ✓ si tratta di trasformazioni meccaniche, ben documentate e supportate da molti tool

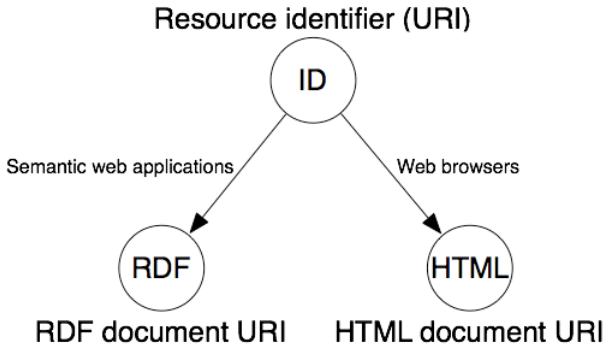
Il ruolo fondamentale degli URI

- **Gli URI hanno reso possibile il *merge***
- ***Chiunque* può creare (meta)dati su *qualunque* risorsa sul Web, per esempio:**
  - **lo *stesso* file XHTML può essere annotato con altri termini**
  - **è possibile *aggiungere semantica* alle risorse Web esistenti utilizzando URI**
  - **gli URI rendono possibile *collegare* (con *proprietà*) i dati tra di loro**
- ***Gli URI sono la base del ruolo di RDF nel Web***
  - **si può reperire l' informazione utilizzando tool già esistenti**
  - **per questo motivo il "Semantic Web", è il … "Semantic *Web*"**

URI, URL,URN …

- **URI**
  - ✓ Uniform Resource Identifier
  - ✓ stringa che identifica univocamente una risorsa generica
    - può essere un indirizzo Web, un documento, un'immagine, un file, un servizio, un indirizzo di posta elettronica, etc.
- **URL**
  - ✓ Uniform Resource Locator
  - ✓ URI che, oltre a identificare una risorsa, fornisce mezzi per agire su di essa o per ottenere una rappresentazione della risorsa descrivendo il suo meccanismo di accesso primario o la sua "ubicazione" ("location") in una rete
- **URN**
  - ✓ Uniform Resource Name
  - ✓ URI che identifica una risorsa mediante un "nome" in un particolare dominio di nomi ("namespace").
    - può essere usato per parlare di una risorsa senza lasciar intendere la sua ubicazione o come ottenerne una rappresentazione.
    - per esempio l'URN urn:isbn:0-395-36341-1 è un URI che consente di individuare univocamente un libro mediante il suo nome 0-395-36341-1 nel namespace dei codici ISBN, ma non suggerisce dove e come possiamo ottenere una copia di tale libro

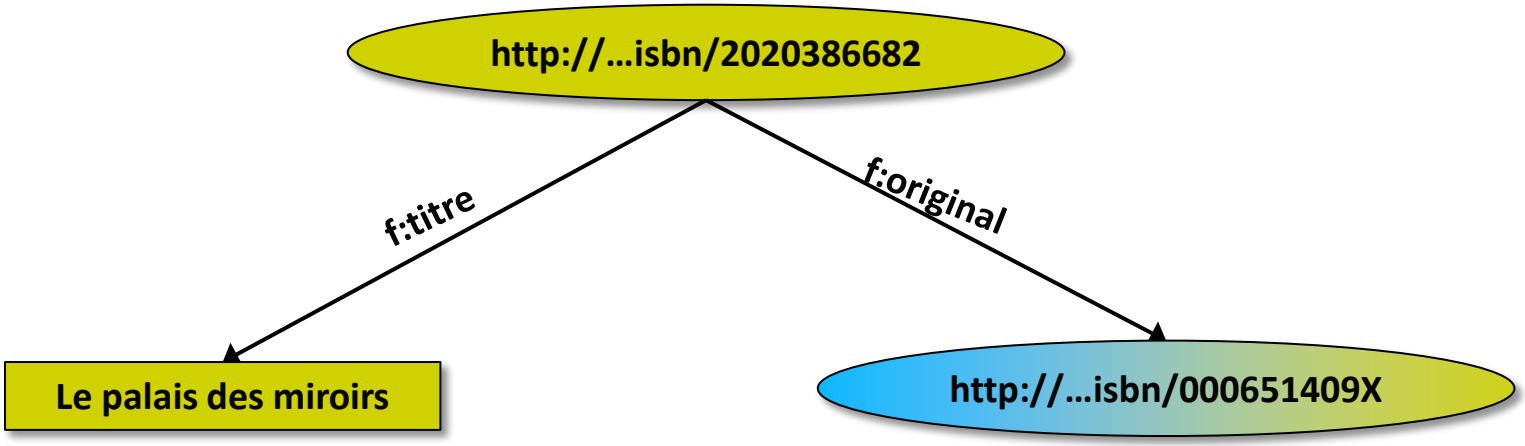




Cool URI don't change!

https://www.w3.org/TR/cooluris/

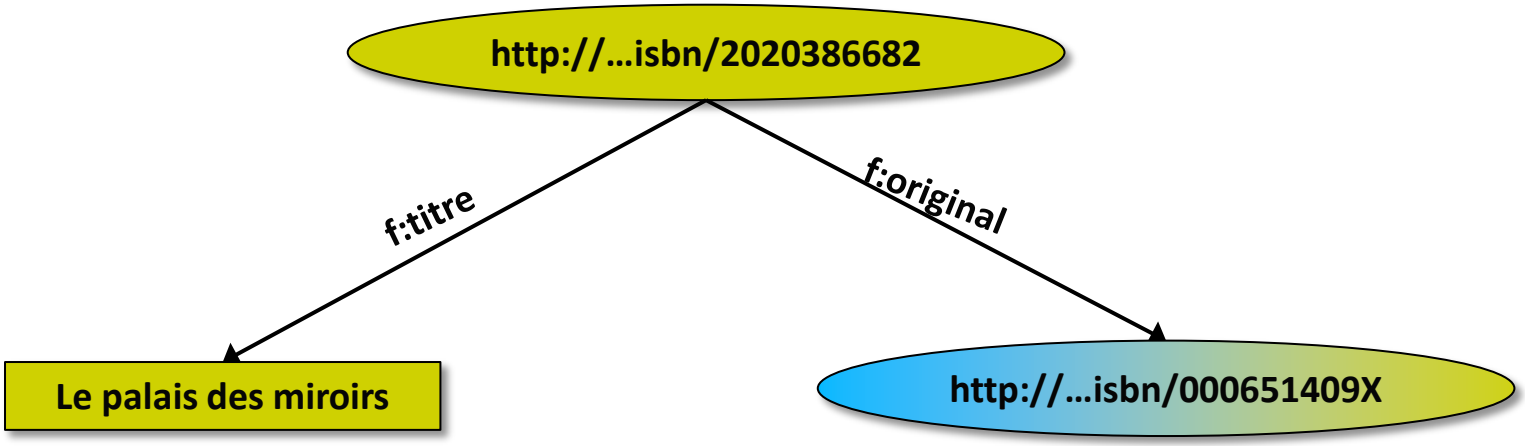
RDF/XML: princìpi



- Codificare nodi e archi come elementi o “literal”

```
«Element for http://…/isbn/2020386682»
  «Element for original»
      «Element for http://…/isbn/000651409X»
  «/Element for original»
«/Element for http://…/isbn/2020386682»
«Element for http://…/isbn/2020386682»
  «Element for titre»
      Le palais des miroirs
  «/Element for titre»
«/Element for http://…/isbn/2020386682»
```

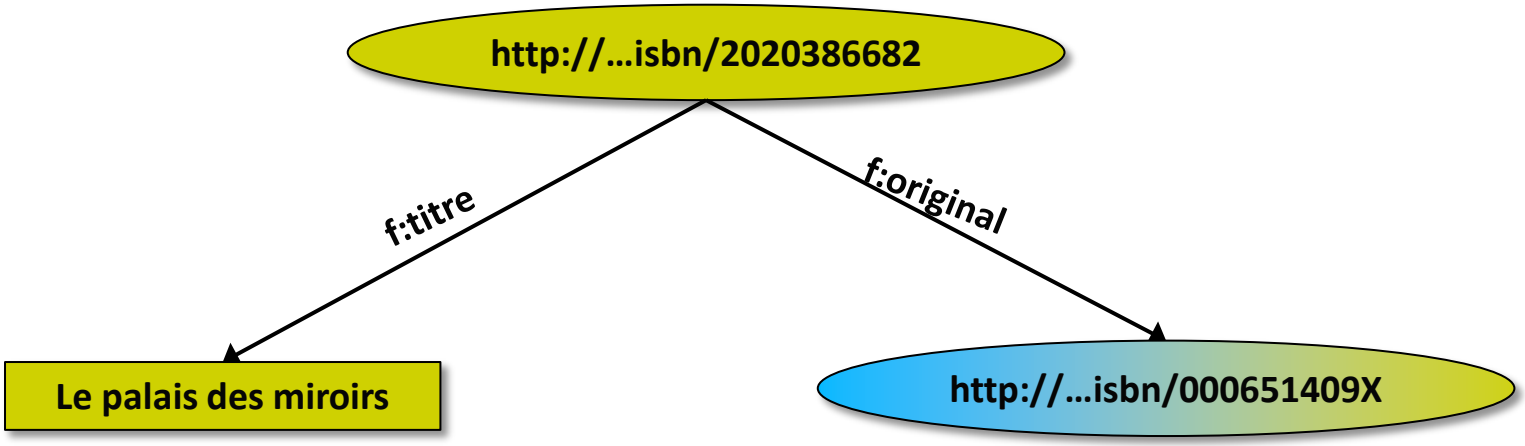
RDF/XML: princìpi (cont.)



- Codificare le risorse (cioè i nodi)

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
   <rdf:Description rdf:about="http://…/isbn/2020386682">
      «Element for original»
         <rdf:Description rdf:about="http://…/isbn/000651409X"/>
       «/Element for f:original»
   </rdf:Description>
<rdf:RDF>
```

RDF/XML: princìpi (cont.)

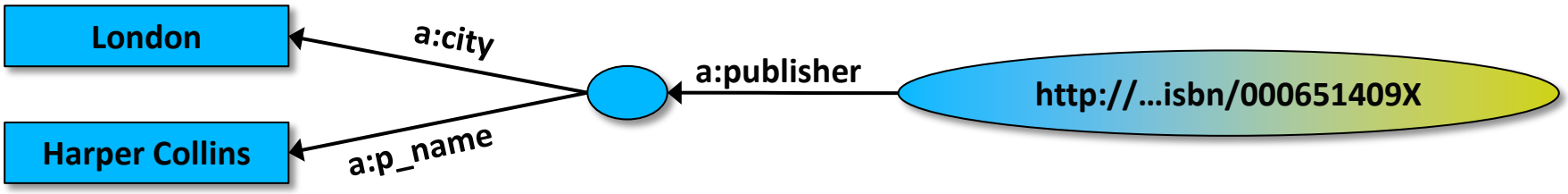


- Codificare le proprietà (cioè gli archi) nei rispettivi namespace

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:f="http://www.editeur.fr">
    <rdf:Description rdf:about="http://…/isbn/2020386682">
      <f:original>
          <rdf:Description rdf:about="http://…/isbn/000651409X"/>
      </f:original>
    </rdf:Description>
<rdf:RDF>
```

Nodi "interni"

- **Consideriamo lo statement:**
  - ✓ **"l'editore è una "cosa" cha ha un nome e un indirizzo"**
- **Abbiamo identificato i nodi mediante URI. Però:**
  - ✓ **…qual è l'URI di "cosa"?**



Una soluzione: creare un URI «extra»

- La risorsa sarà visibile sul web
  - ✓ attenzione a definire URI univoci!

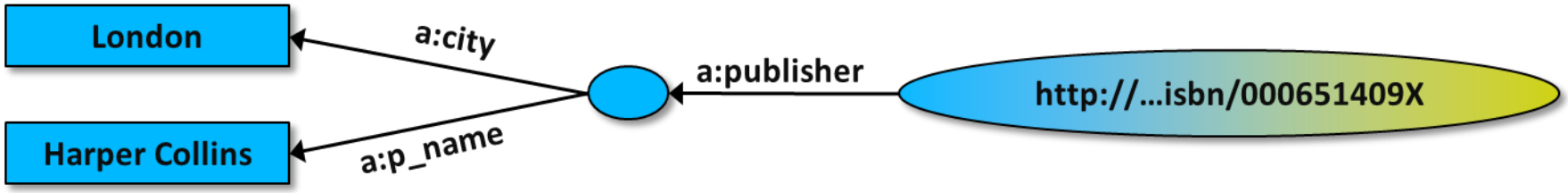
```
<rdf:Description rdf:about="http://…/isbn/000651409X">
   <a:publisher rdf:resource="urn:uuid:f60ffb40-307d-…"/>
</rdf:Description>
<rdf:Description rdf:about="urn:uuid:f60ffb40-307d-…">
   <a:p_name>HarpersCollins</a:p_name>
   <a:city>HarpersCollins</a:city>
</rdf:Description>
```

Identificatori interni (blank node)

- Risorsa non visibile all'esterno

```
<rdf:Description rdf:about="http://…/isbn/000651409X">
   <a:publisher rdf:nodeID="A234"/>
</rdf:Description>
<rdf:Description rdf:nodeID="A234">
   <a:p_name>HarpersCollins</a:p_name>
   <a:city>HarpersCollins</a:city>
</rdf:Description>
```

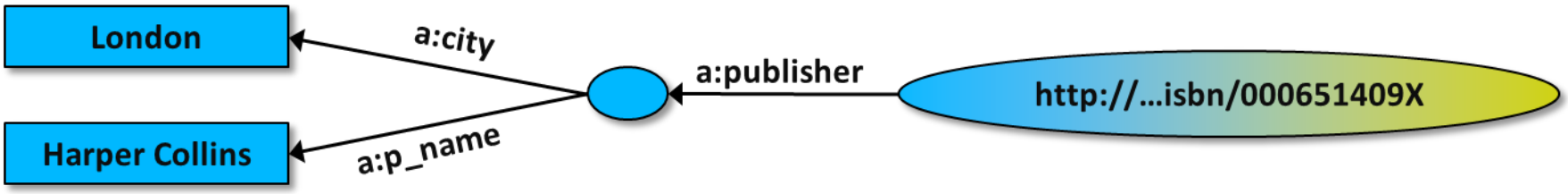
```
<http://…/isbn/2020386682> a:publisher _:A234.
_:A234 a:p_name "HarpersCollins".
```



E se lasciassimo fare tutto al sistema?

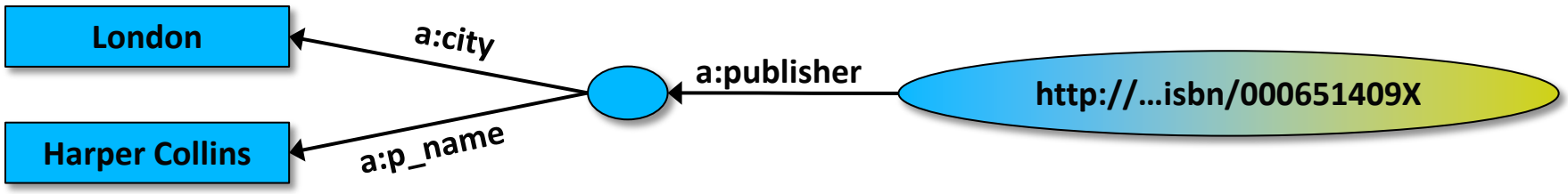
- Il sistema provvederà a creare internamente un "nodeID"
  - ✓ non dovremo preoccuparci del nome…

```
<   <rdf:Description rdf:about="http://…/isbn/000651409X">
  <a:publisher>
      <rdf:Description>
          <a:p_name>HarpersCollins</a:p_name>
          …
      </rdf:Description>
  </a:publisher>
</rdf:Description>
```



...in Turtle

```
<http://…/isbn/000651409X> a:publisher [
    a:p_name "HarpersCollins";
    …
].
```



Riflessioni sui blank node

- Nel processo di merging occorre prestare attenzione ai blank node
  - ✓ i blank node con lo stesso ID in grafi diversi sono differenti
  - ✓ le implementazioni devono considerare attentamente questo fatto …
- Molte applicazioni preferiscono non utilizzare I blank node, e definire nuovi URI “on-the-fly”
- Dal punto di vista della logica, i blank node rappresentano uno statement (proposizione) “esistenziale”
  - ✓ “esiste una risorsa tale che…”

Reification

- Per poter certificare la credibilità di un certo statement, è necessario poter formulare statements about statements
- Questo processo viene detto reification nella comunità che si interessa di Knowledge Representation

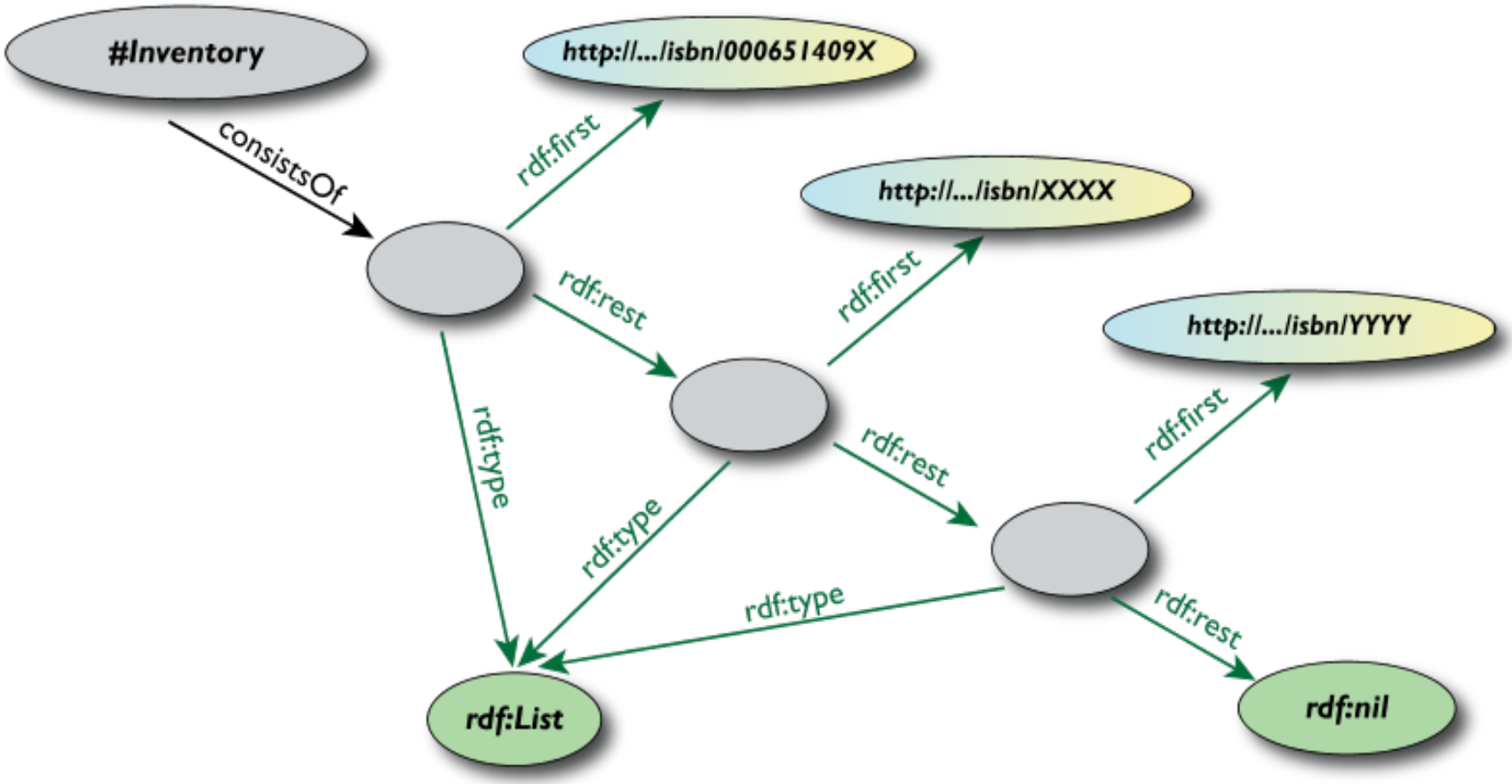
```
exproducts:item10245   exterms:weight   "2.4"^^xsd:decimal .
```

```
exproducts:triple12345   rdf:type        rdf:Statement .
exproducts:triple12345   rdf:subject     exproducts:item10245 .
exproducts:triple12345   rdf:predicate   exterms:weight .
exproducts:triple12345   rdf:object      "2.4"^^xsd:decimal .
```

Collezioni (liste)

- **Potremmo avere degli statement del tipo:**
  - ✓ **“Il catalogo dei libri è una “cosa” che consiste di: <http://…/isbn/000651409X>, <http://…/isbn/000XXXX>, …”**
- **Ma vogliamo elencare gli elementi esattamente in quell’ordine**
- **I “blank node” non sono adeguati**

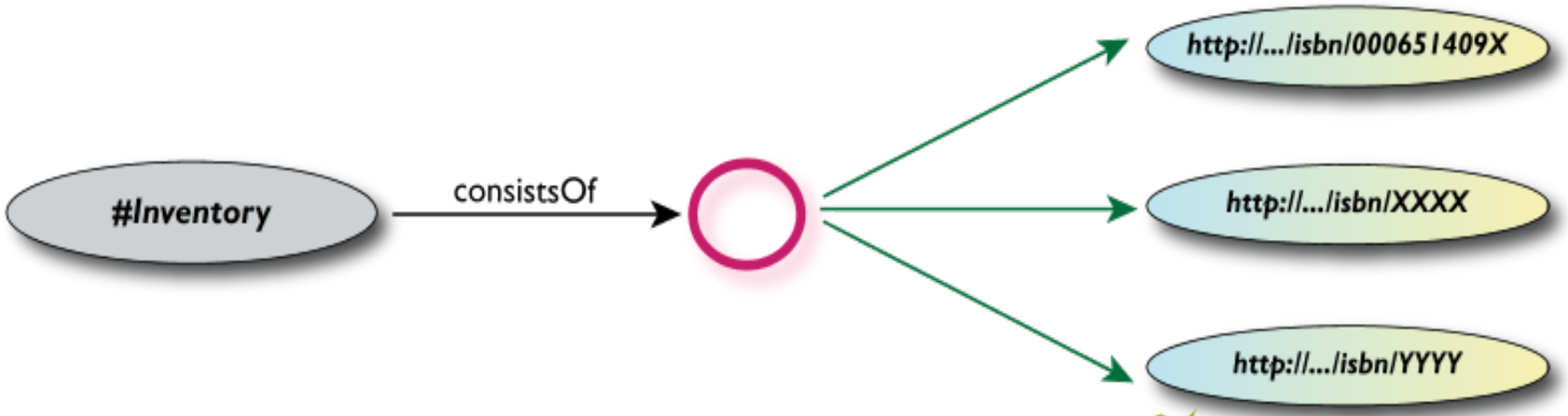
Collezioni (liste) (cont.)



...in RDF/XML e Turtle

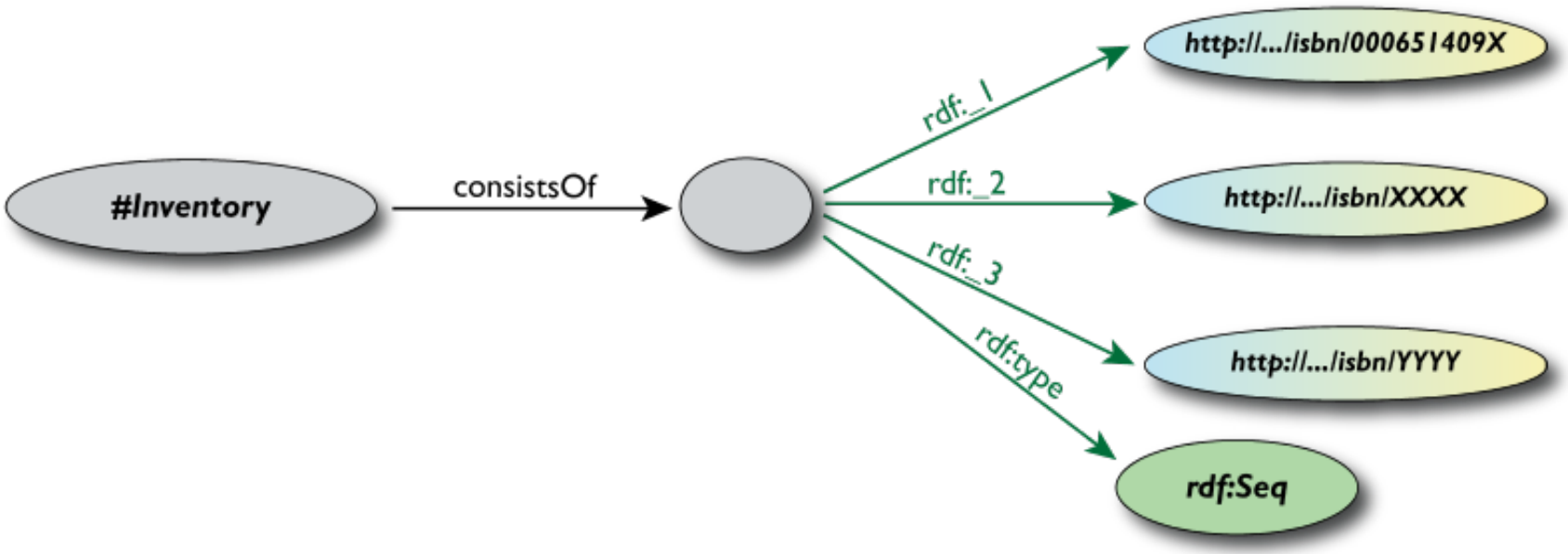
```
<rdf:Description rdf:about="#Inventory">
    <a:consistsOf rdf:parseType="Collection">
        <rdf:Description rdf:about="http://.../isbn/000651409X"/>
        <rdf:Description rdf:about="http://.../isbn/XXXX"/>
        <rdf:Description rdf:about="http://.../isbn/YYYY"/>
    </a:consistsOf>
</rdf:Description>
```

```
:Inventory a:consistsOf
(<http://.../isbn/000651409X> <http://.../isbn/XXXX> ...) .
```



Sequence

- Usare i costrutti predefiniti:
  - ✓ RDF Schema class Seq
  - ✓ RDF properties rdf:_1, rdf:_2, …
- L'accordo sulla semantica è di contenuto sequenziale



Serializzazione delle Sequence

In RDF/XML:

```
<rdf:Description rdf:about="#Inventory">
  <a:consistsOf>
      <rdf:Description>
          <rdf:type rdf:resource="http:...rdf-syntax-ns#Seq">
          <rdf:_1 rdf:resource="http://.../isbn/000651409X">
          ...
      </rdf:Description>
  </a:consistsOf>
</rdf:Description/>
```

In Turtle

```
:Inventory
  a:consistsOf [
    rdf:type <http:...rdf-syntax-ns#Seq>;
    rdf:_1   <http://.../isbn/000651409X>;
          ...
  ].
```

Sequences (RDF/XML semplificato)

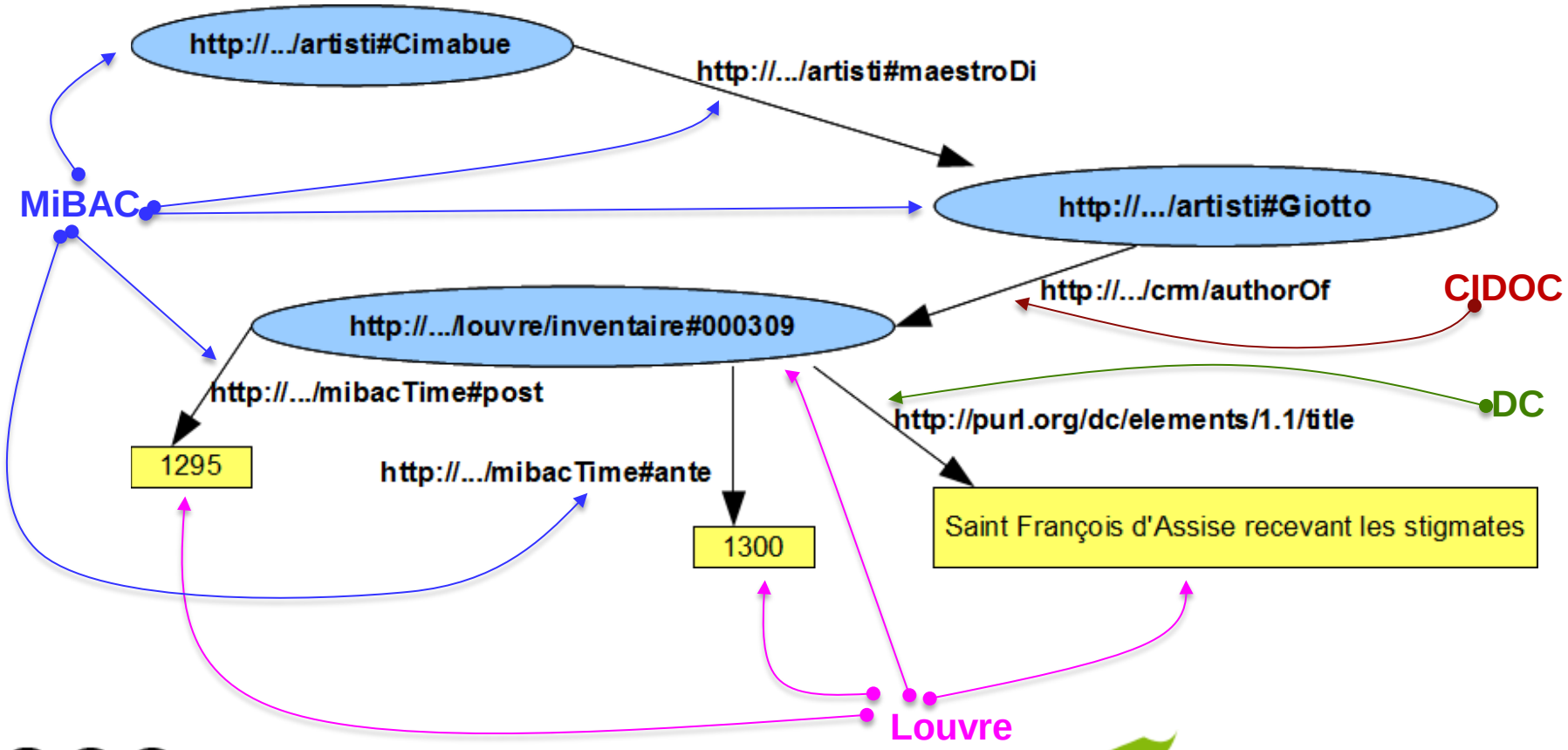
```
<rdf:Description rdf:about="#Inventory">
    <a:consistsOf>
        <rdf:Seq>
            <rdf:li rdf:resource="http://.../isbn/000651409X">
            ...
        </rdf:Seq>
    </a:consistsOf>
</rdf:Description/>
```

Altri contenitori

- **Definiti anche in RDFS**
  - ✓ **rdf:Bag**
    - **contenitore generico, senza una particolare semantica**
  - ✓ **rdf:Alt**
    - **accordo sulla semantica: solo uno dei costituenti è "valido"**
- **Nota: la definizione semantica di questi contenitori è incompleta:**
  - ✓ **meglio non utilizzarli…**
  - ✓ **per i bag, utilizzare predicati ripetuti**
  - ✓ **per le sequenze, usare le liste**

Un grafo RDF (WorldWide!)

...un insieme di triple s-p-o (subject-predicate-object)



Conclusioni

- Il Web è nato per condividere conoscenza
- RDF è la base
- Le tecnologie del Semantic Web costituiscono il quadro di riferimento

Grazie per l'attenzione!

(Nobody's perfect!)

?

Domande

Slide a: www.orestesignore.eu/education/lda/slides/lda1.pdf