

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

---

SCUOLA DI SCIENZE  
Corso di Laurea in Informatica

**OCEB: Un'applicazione web per la  
visualizzazione della rete citazionale di  
OpenCitations**

**Relatore:  
Chiar.mo Prof.  
Fabio Vitali**

**Presentata da:  
Lorenzo Massimiliani**

**Correlatore:  
Dott.  
Silvio Peroni**

**Sessione III Anno Accademico 2015/2016**



# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 Strumenti per la presentazione di metadati bibliografici e citazionali</b>	<b>5</b>
1.1 Semantic web . . . . .	6
1.2 Strumenti per la visualizzazione di dati RDF . . . . .	6
1.3 Strumenti per la visualizzazione di metadati bibliografici e citazionali disponibili su triplestore . . . . .	7
1.3.1 Data browser . . . . .	8
1.3.2 BEX (Bibliographic EXplorer) . . . . .	8
1.4 OpenCitations . . . . .	9
1.4.1 Risorse bibliografiche (fabio:Expression) . . . . .	10
1.4.2 Agenti (foaf:Agent) . . . . .	11
1.4.3 Ruoli (pro:RoleInTime) . . . . .	12
1.4.4 Identificativi (datacite:Identifier) . . . . .	12
1.4.5 Formati (fabio:Manifestation) . . . . .	12
1.4.6 Riferimenti bibliografici (biro:BibliographicReference) . . . . .	13
<b>2 OCEB: panoramica</b>	<b>15</b>
2.1 Scopo dell'applicazione . . . . .	15
2.2 Visualizzazione delle entità bibliografiche . . . . .	18
2.2.1 Visualizzazione di risorse bibliografiche . . . . .	19
2.2.2 Visualizzazione di agenti . . . . .	23

2.2.3	Visualizzazione di ruoli . . . . .	24
2.2.4	Visualizzazione di identificativi . . . . .	24
2.2.5	Visualizzazione di formati . . . . .	25
2.2.6	Visualizzazione di riferimenti bibliografici . . . . .	26
2.3	Navigazione attraverso OCEB . . . . .	27
<b>3</b>	<b>OCEB: dettagli implementativi</b>	<b>31</b>
3.1	RSLT (RDF Stylesheet Language Transformations) . . . . .	32
3.2	Descrizione dei template RSLT . . . . .	34
3.2.1	Template per visualizzare le risorse bibliografiche . . . . .	34
3.2.2	Template per visualizzare gli agenti . . . . .	46
3.2.3	Template per visualizzare i ruoli . . . . .	48
3.2.4	Template per visualizzare gli identificativi . . . . .	48
3.2.5	Template per visualizzare i formati . . . . .	49
3.2.6	Template per visualizzare i riferimenti bibliografici . . . . .	50
3.3	Codice javascript utilizzato nell'applicazione . . . . .	51
3.3.1	Funzione per inserire l'identificativo di OpenCitations . . . . .	52
3.3.2	Funzione per ordinare in ordine alfabetico le risorse citate . . . . .	53
3.3.3	Funzioni per rimuovere caratteri speciali . . . . .	53
<b>4</b>	<b>Valutazione</b>	<b>55</b>
4.1	Valutazione dei tempi di caricamento delle pagine HTML . . . . .	55
4.1.1	Valutazione del tempo di caricamento delle risorse bibliografiche . . . . .	57
4.1.2	Valutazione del tempo di caricamento degli agenti . . . . .	59
4.1.3	Valutazione del tempo di caricamento dei ruoli . . . . .	59
4.1.4	Valutazione del tempo di caricamento degli identificativi . . . . .	60
4.1.5	Valutazione del tempo di caricamento dei formati . . . . .	60
4.1.6	Valutazione del tempo di caricamento dei riferimenti bibliografici . . . . .	61
<b>5</b>	<b>Conclusioni</b>	<b>63</b>





# Indice del codice

3.1	Primo esempio di template RSLT . . . . .	32
3.2	Secondo esempio di template RSLT . . . . .	32
3.3	Esempio di query da RSLT . . . . .	33
3.4	Template per riconoscere le diverse risorse bibliografiche . . .	35
3.5	Template principale degli articoli . . . . .	36
3.6	Template per elencare gli autori di una risorsa bibliografica . .	38
3.7	Template per inserire la gerarchia di contenimento . . . . .	40
3.8	Template per inserire il nome dell'editore di una risorsa biblio- grafica . . . . .	42
3.9	Template per inserire una lista di identificativi . . . . .	43
3.10	Template per inserire le risorse che citano la risorsa principale	44
3.11	Template per inserire i riferimenti bibliografici . . . . .	45
3.12	Template principale degli agenti . . . . .	46
3.13	Template principale dei ruoli . . . . .	48
3.14	Template principale degli identificativi . . . . .	48
3.15	Template principale dei formati . . . . .	49
3.16	Template principale dei riferimenti bibliografici . . . . .	50
3.17	Funzione per inserire l'identificativo di OpenCitations . . . . .	52
3.18	Funzione per ordinare le risorse . . . . .	53
3.19	Funzione per sostituire dei caratteri . . . . .	53





# Elenco delle figure

1.1	Schermata di data browser . . . . .	8
1.2	Schermata di BEX . . . . .	9
1.3	Diagramma dell'ontologia di OpenCitations . . . . .	10
2.1	Esempio di visualizzazione di una risorsa in OpenCitations . . . . .	16
2.2	Esempio di visualizzazione di una risorsa con OCEB . . . . .	17
2.3	Esempio di visualizzazione di un articolo . . . . .	19
2.4	Esempio di visualizzazione di un volume . . . . .	21
2.5	Esempio di visualizzazione di una rivista . . . . .	22
2.6	Esempio di visualizzazione di un agente . . . . .	23
2.7	Esempio di visualizzazione di un ruolo . . . . .	24
2.8	Esempio di visualizzazione di un identificativo . . . . .	25
2.9	Esempio di visualizzazione di un formato . . . . .	25
2.10	Esempio di visualizzazione di un riferimento bibliografico . . . . .	26
2.11	Prima schermata di navigazione . . . . .	27
2.12	Seconda schermata di navigazione . . . . .	28
2.13	Terza schermata di navigazione . . . . .	29
4.1	Tempi di esecuzione di RSLT . . . . .	56



# Introduzione

Questa dissertazione presenta OCEB (OpenCitations Entities Browser), un'applicazione web che permette la visualizzazione di metadati presenti nel database di OpenCitations. OpenCitations è un progetto il cui scopo principale è la raccolta di metadati bibliografici e citazionali provenienti dalla letteratura accademica. OCEB visualizza queste informazioni, in modo ordinato, in una pagina HTML e fornisce un meccanismo che consente all'utente di navigare tra i contenuti di OpenCitations.

Il database di OpenCitations, da cui visualizza i metadati OCEB, è precisamente un triplestore, un database costruito appositamente per il salvataggio e il recupero di dati RDF. L'unità base per rappresentare un'informazione in RDF è lo statement ovvero una tripla soggetto – predicato – oggetto. Qualsiasi informazione presente in un triplestore è di difficile consultazione per chi non conosce il semantic web e le tecnologie legate a esso. OCEB è stato sviluppato principalmente per semplificare la visualizzazione di metadati bibliografici e citazionali, espressi in RDF, nascondendo le tecnologie sottostanti all'utente.

Le risorse di OpenCitations sono definite da un URI e sono descritte dai metadati presenti nel triplestore. OCEB è composto da pagine HTML, predisposte alla visualizzazione di quelle risorse. Attraverso un parametro è possibile specificare quale risorsa si vuole visualizzare su una pagina dell'applicazione. OCEB utilizza la libreria RSLT, che permette la trasformazione di dati

RDF tramite template dichiarativi. Questa libreria è utile per ottenere i dati presenti nel triplestore di OpenCitations, attraverso query SPARQL, il linguaggio di interrogazione dei dati RDF. I risultati ottenuti vengono catturati dai template RSLT che li inseriscono all'interno della pagina HTML.

OCEB è composto da più pagine HTML, ognuna indicata per visualizzare un tipo diverso di risorsa. OpenCitations contiene al suo interno entità bibliografiche di sei tipi principali: risorse bibliografiche, agenti, ruoli, identificativi, formati e riferimenti bibliografici. Ogni entità è molto diversa dalle altre, ciò significa che contiene informazioni molto differenti. Per poter applicare un layout diverso e per presentare informazioni particolari per ogni tipo di risorsa è stato scelto di dedicare una pagina alla visualizzazione di ognuna delle sei tipologie di risorse. Ogni pagina di OCEB contiene dei template specifici per la presentazione di un tipo di entità bibliografica. In questo modo una risorsa bibliografica verrà mostrata descrivendone il titolo e gli autori, ad esempio, mentre un agente verrà mostrato descrivendone il nome e il cognome.

L'applicazione, inoltre, costituisce un vero e proprio browser, che permette all'utente di navigare tra le risorse di OpenCitations. Infatti, una visualizzazione di una entità bibliografica non si limita a presentare delle informazioni in una pagina HTML, ma per ogni informazione inserita che fa riferimento a un'altra risorsa è presente un link. Quel link permette di visualizzare quella risorsa, sempre mediante una pagina HTML di OCEB. In questo modo un utente potrà consultare delle informazioni che sono collegate più o meno direttamente con una risorsa. Ad esempio una risorsa bibliografica può avere degli autori e OCEB, se sono presenti, li inserirà nella pagina rappresentandoli con il loro nomi. Questi dati rappresentano il valore testuale di un link che rimanda alla pagina dell'applicazione che visualizza quell'autore.

I tempi di caricamento di una pagina HTML di OCEB dipendono quasi

esclusivamente dal tempo impiegato dai template RSLT a eseguire le query SPARQL. Ci può essere una notevole differenza di tempo tra il caricamento di due risorse diverse in una pagina. Più informazioni contiene una risorsa, più dati dovranno essere presentati e più il tempo aumenta.

La parte restante di questa dissertazione è organizzata come segue. Nel capitolo 1 verrà introdotto l'ambito nel quale è stato sviluppato OCEB. Saranno descritti alcuni strumenti che permettono la visualizzazione di dati RDF, con particolare riferimento ai metadati bibliografici e citazionali e sarà presentato il modello di OpenCitations. Il capitolo 2 proporrà una panoramica dell'applicazione, in cui verranno mostrate e descritte alcune schermate visualizzate dall'applicazione. Nel capitolo 3 verranno esposti alcuni template RSLT, che hanno permesso di sviluppare le schermate del capitolo precedente. Nel capitolo 4 verrà presentata una valutazione quantitativa di OCEB, svolta confrontando il tempo di caricamento di risorse diverse sull'applicazione.



# Capitolo 1

## Strumenti per la presentazione di metadati bibliografici e citazionali

Per comprendere al meglio da quali necessità è nato il bisogno di sviluppare OCEB (OpenCitations Entities Browser), bisogna prima discutere il contesto nel quale nasce tale progetto, ovvero quello della visualizzazione di metadati bibliografici e citazionali. Discuteremo di questo aspetto e si mostreranno alcune delle più significative soluzioni, presenti in letteratura, per la visualizzazione sul browser di questi metadati. Tra queste soluzioni nomineremo Semantic Lancet, che utilizza RSLT, una libreria che permette la visualizzazione di dati RDF, e BEX. Prima di discutere di queste tecnologie, sarà utile approfondire alcuni temi riguardo al semantic web, l'ambito di cui esse fanno parte. Inoltre, in questo capitolo, verrà descritto OpenCitations, un progetto il cui scopo principale è la raccolta di metadati bibliografici e citazionali provenienti dalla letteratura accademica. Questa descrizione ci sarà utile in quanto OpenCitations contiene i metadati che verranno visualizzati attraverso OCEB.

## 1.1 Semantic web

Il semantic web[1] rappresenta un web composto da dati strutturati con un preciso significato semantico e predisposti all'elaborazione automatica. Il W3C[2] ha definito come modello di memorizzazione di questi dati il formato RDF[3] e come linguaggio di interrogazione SPARQL[4]. L'unità base per rappresentare un'informazione in RDF è lo statement. Uno statement è una tripla della forma soggetto – predicato - oggetto, dove il soggetto è una risorsa, il predicato è una proprietà e l'oggetto è un valore o una risorsa. Consideriamo risorsa qualsiasi cosa descrivibile sul web, proprietà relazioni tra valori e risorse, e valori sono, ad esempio, stringhe, interi e booleani. Spesso le relazioni vengono indicate utilizzando un'ontologia, ovvero un insieme di termini condivisi utilizzati per esprimere concetti in un determinato dominio. Per definire o creare una nuova ontologia viene utilizzato il linguaggio OWL (Web Ontology Language) [5] Definiamo un triplestore come un database preposto alla memorizzazione di informazioni in formato RDF. I dati possono essere raccolti in un triplestore per molti motivi, come ad esempio per elaborarli oppure per renderli disponibili alle macchine. Tra questi motivi rientra anche quello di rendere visualizzabili questi dati all'utente e fornire tramite essi delle informazioni.

## 1.2 Strumenti per la visualizzazione di dati RDF

In questa sezione parleremo di alcune tecnologie utilizzate per la visualizzazione di dati RDF, quindi, senza limitarci al dominio dei metadati bibliografici e citazionali. Molto spesso gli statement RDF vengono rappresentati attraverso dei grafi, questo grazie al loro modello soggetto – predicato – oggetto, che li rende particolarmente adatti a una visualizzazione di questo tipo [6]. Esistono diversi software, come RDFViz++[7], e modelli, come quello mostrato in [8], che permettono una visualizzazione di dati RDF mediante



grafi. Solitamente in queste rappresentazioni i nodi dei grafi coincidono con le risorse, mentre gli archi con i predicati [9]. Andando oltre alla rappresentazione di dati RDF mediante grafi, introduciamo una libreria chiamata RSLT [10], che permette la trasformazione di dati, espressi in RDF, tramite template dichiarativi. RSLT, giunta alla versione 1.1 [11], viene utilizzata da Semantic Lancet, che andremo in seguito ad approfondire, per visualizzare dei dati RDF in una pagina HTML. Infatti all'interno di una pagina web è possibile inserire dei template RSLT che eseguiranno delle query SPARQL e i risultati trovati verranno inseriti nella pagina.

### **1.3 Strumenti per la visualizzazione di metadati bibliografici e citazionali disponibili su triplestore**

Descriviamo in questa sezione alcuni strumenti per visualizzare metadati bibliografici e citazionali provenienti dal triplestore di Semantic Lancet. Semantic Lancet [12,13] è un progetto proposto dal Dipartimento di Informatica dell'Università di Bologna che ha lo scopo di creare un dataset aperto (Semantic Lancet Triplestore) di pubblicazioni accademiche e fornire gli strumenti per visualizzare i contenuti del triplestore. Il triplestore corrente di questo progetto è composto da tutti i documenti pubblicati dalla rivista Journal of Web Semantics di Elsevier. Per ogni documento vengono memorizzate informazioni riguardanti i suoi dati biografici come il sommario e le citazioni, espresse in dati RDF. Vengono utilizzate le ontologie SPAR che permettono di descrivere le varie parti di un dominio di pubblicazione, con particolare supporto alle pubblicazioni scientifiche[14]. L'applicazione Semantic Lancet Browser è totalmente basata su RSLT, la libreria che abbiamo descritto precedentemente. Le finalità di questo progetto sono molteplici e vanno dal raccogliere dati citazionali scientifici, all'offrire uno SPARQL endpoint per ottenere i dati, fino al presentare degli strumenti per permettere una visualizza-

zione dei dati del triplestore in maniera semplice, ad alto livello, nascondendo tutta la complessità delle tecnologie sottostanti. Gli strumenti presentati a questo scopo da Semantic Lancet sono due: il data browser e BEX. Andiamo ora a presentarli uno per volta.

### 1.3.1 Data browser

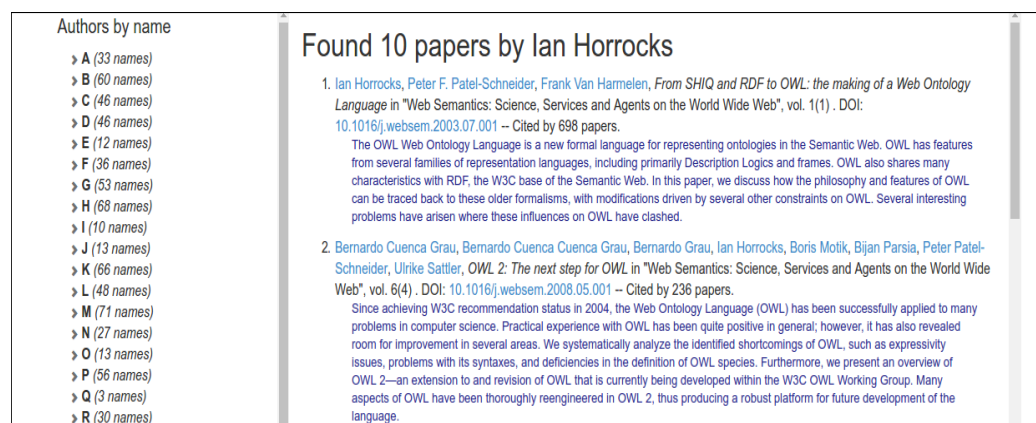


Figura 1.1: Schermata di data browser

La figura 1.1 mostra la pagina web dedicata all'applicazione data browser (<http://www.semanticlancet.eu/browser/>). Possiamo notare a sinistra l'elenco degli autori presenti nel Semantic Lancet Triplestore e a destra un elenco degli articoli scritti da un determinato autore (in figura Ian Horrocks). Questa applicazione può essere utile per reperire informazioni su un determinato autore, come ad esempio l'elenco dei documenti scritti da lui. Inoltre per ogni documento elencato sono fornite informazioni quali la lista degli autori, il titolo, un identificativo, il numero di citazioni ricevute e il testo dell'abstract.

### 1.3.2 BEX (Bibliographic EXplorer)

BEX[15,16,17] rappresenta un'applicazione web interattiva, volta a sostenere l'esplorazione e l'analisi degli articoli e delle citazioni disponibili nel Semantic

Lancet Triplestore. Questa applicazione, che fa parte del progetto Semantic Lancet, è reperibile alla pagina <http://eelst.cs.unibo.it:8089//app/homeSearch>.

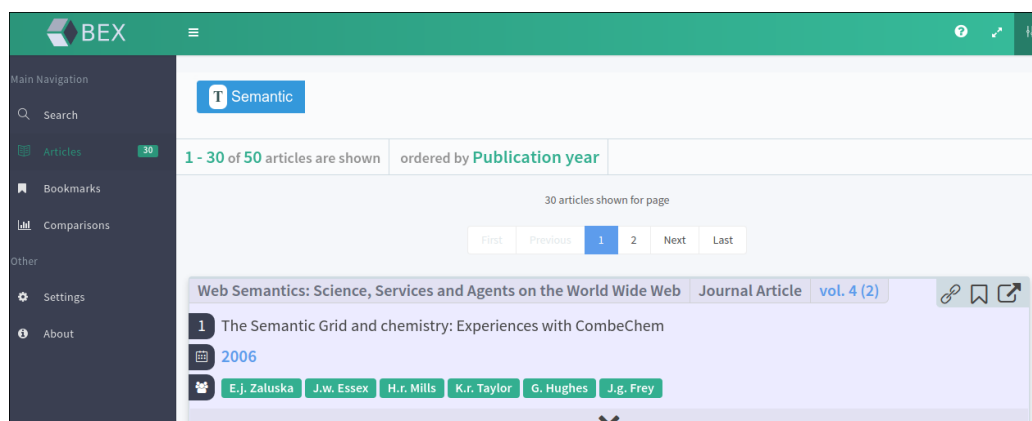


Figura 1.2: Schermata di BEX

Nella figura 1.2 viene mostrato un esempio di ricerca di articoli attraverso una parola chiave presente nel titolo, ovvero "Semantic". Come si può vedere nella figura, è presente un elenco di articoli che rispettano il filtro utilizzato nella ricerca. BEX è stato progettato per essere utilizzato da diverse persone con compiti e scopi differenti. Ad esempio permette a un recensore di valutare la validità di un articolo, a un lettore di cercare collaboratori per un progetto e a un valutatore di valutare l'impatto di un articolo.

## 1.4 OpenCitations

OpenCitations[18] è un progetto il cui scopo principale è la creazione del Open Citation Corpus (OCC), una raccolta di metadati bibliografici e citazionali, in formato RDF, della letteratura accademica. Queste informazioni sono descritte mediante ontologie SPAR, secondo un modello di metadati specifico definito in [19]. Andiamo adesso a descrivere questo modello, in modo da avere una visione più chiara quando nel capitolo 2 parleremo di quello che viene fatto dall'applicazione sviluppata. OpenCitations, infatti, contiene

al suo interno il triplestore che sarà utilizzato da OCEB per visualizzare i metadati.

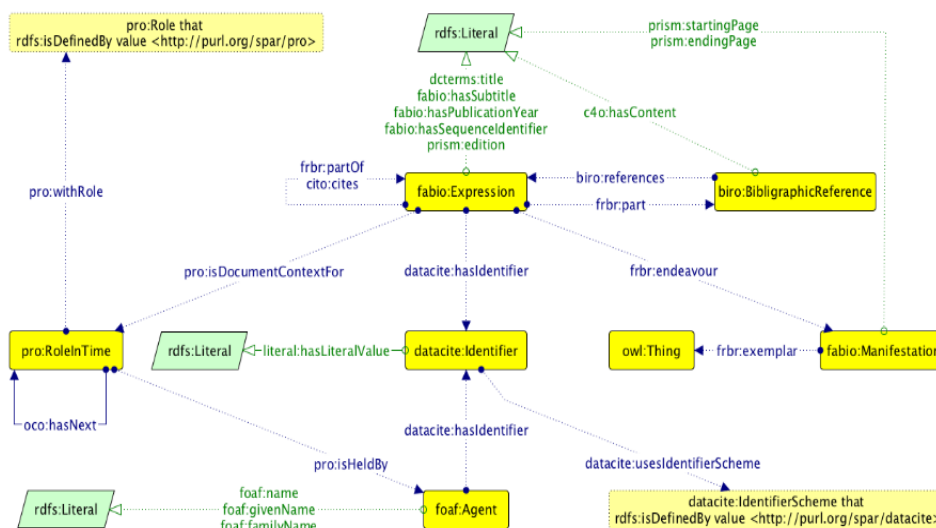


Figura 1.3: Diagramma dell'ontologia di OpenCitations

Nella figura 1.3 viene mostrato un diagramma (generato con Graffoo <http://www.essepuntato.it/graffoo/>) che mostra una panoramica dell'ontologia di OpenCitations. Questa ontologia, chiamata OpenCitations Ontology (OCO) non è una nuova ontologia, ma è un raggruppamento di entità ontologiche già presenti allo scopo di fornire i metadati per OCC. L'Open Citation Corpus contiene principalmente sei tipi di entità: risorse bibliografiche, agenti, ruoli, identificativi, formati e riferimenti bibliografici. Procediamo, ora, descrivendoli singolarmente e dandone più una descrizione discorsiva che formale.

### 1.4.1 Risorse bibliografiche (fabio:Expression)

Le risorse bibliografiche rappresentano le entità centrali di OpenCitations. Possiedono diverse proprietà che hanno come valore stringhe, come il titolo, il sottotitolo, l'anno di pubblicazione e l'identificatore di sequenza. Molte altre proprietà permettono di legare una risorsa bibliografica con altre

risorse, come mostrato nella figura 1.3. Possiamo vedere come la risorsa `fabio:Expression`, che rappresenta una risorsa bibliografica, sia collegata alle altre risorse attraverso le proprietà scritte in blu. E' possibile che una risorsa bibliografica sia legata a un identificativo, che la rappresenta, a un ruolo, che svolge una funzione dentro alla risorsa bibliografica, a un riferimento bibliografico, che contiene, a un formato e a un'altra risorsa bibliografica di cui fa parte. Per ogni risorsa bibliografica è specificato un sottotipo come ad esempio: articolo (`fabio:JournalArticle`), fascicolo (`fabio:JournalIssue`), volume (`fabio:JournalVolume`), rivista (`fabio:Journal`), libro (`fabio:Book`) e capitolo (`fabio:BookChapter`). Il sottotipo definisce quali proprietà sono definite per la risorsa bibliografica. Ad esempio un volume non ha un titolo e un sottotitolo, ma possiede un identificativo di sequenza. Al contrario, un articolo non avrà un identificativo di sequenza, ma bensì un titolo e, eventualmente, un sottotitolo. La proprietà più interessante, definita per le risorse bibliografiche, è sicuramente la proprietà `frbr:partOf` che permette di collegare due risorse bibliografiche, di cui una è contenuta nell'altra. Di solito questa proprietà viene utilizzata per indicare il contenimento di un articolo in un fascicolo, di un fascicolo in un volume, di un volume in una rivista e di un capitolo in un libro.

### 1.4.2 Agenti (`foaf:Agent`)

Le entità di tipo agente rappresentano un individuo oppure una casa editrice. Gli agenti sono relazionati con i ruoli dalla proprietà `pro:isHeldBy`. Tra le proprietà che hanno come soggetto un agente che rappresenta una persona troviamo `foaf:givenName` e `foaf:familyName`, che hanno come valore rispettivamente il nome e cognome della persona. Mentre la proprietà per indicare il nome di una casa editrice è `foaf:name`. Anche per gli agenti, così come per le risorse bibliografiche, è possibile che siano legati a un identificativo dalla proprietà `datacite:hasIdentifier`.

### 1.4.3 Ruoli (**pro:RoleInTime**)

I ruoli mettono in relazione risorse bibliografiche con agenti. La proprietà `pro:isDocumentContextFor`, come si vede nella figura 1.3, collega una risorsa bibliografica con un ruolo, mentre la proprietà `pro:isHeldBy` lega un ruolo a un agente. Il ruolo, che si comporta da intermediario tra due altre entità, è utile perché permette di definire con quale ruolo un agente è parte attiva della risorsa bibliografica. Infatti, la proprietà `pro:withRole` permette di indicare se l'agente che si sta relazionando con la risorsa bibliografica svolga il ruolo di autore o di editore. Inoltre, attraverso la proprietà `oco:hasNext` viene specificato il ruolo successivo legato sempre alla stessa risorsa bibliografica. In questo modo viene specificata tramite una lista di ruoli l'ordine di importanza degli autori. Se consideriamo una lista di ruoli relazionati tra loro dalla proprietà `oco:hasNext`, tutti di tipo autore, il primo ruolo sarà legato, dalla proprietà `pro:isHeldBy` all'autore più importante e così via.

### 1.4.4 Identificativi (**datacite:Identifier**)

Gli identificativi rappresentano un particolare valore che permette di identificare entità come risorse bibliografiche o agenti. La proprietà `datacite:usesIdentifierScheme` ha come soggetto una risorsa di tipo identificativo e come oggetto un valore che indica il tipo dell'identificativo. I principali tipi di identificativi presenti in OpenCitations sono: DOI (<https://www.doi.org/>), URL, PMID, PMCID, ISSN, ORCID e ISBN. L'ultima proprietà che considereremo è `literal:hasLiteralValue` che specifica un determinato valore di un identificativo.

### 1.4.5 Formati (**fabio:Manifestation**)

Un formato indica generalmente da che pagina a che pagina è rappresentato un articolo all'interno di un fascicolo. Le proprietà utilizzate per fare ciò sono `prism:startingPage`, che indica la prima pagina, e `prism:endingPage`,

che indica l'ultima pagina. Questo tipo di risorsa è legata a una risorsa bibliografica dalla proprietà `frbr:embodiment`

#### **1.4.6 Riferimenti bibliografici (`biro:BibliographicReference`)**

Un riferimento bibliografico indica un testo utilizzato da un articolo per citarne un altro. La proprietà `c4o:hasContent` permette di descrivere testualmente il contenuto di questa entità. La proprietà `frbr:part`, che ha come soggetto una risorsa bibliografica e come oggetto un riferimento, definisce quale articolo contiene il riferimento in questione. Mentre `biro:references`, che ha come soggetto un riferimento bibliografico e come oggetto una risorsa bibliografica, permette di indicare a quale articolo si riferisce il riferimento. Per ogni riferimento bibliografico presente in un articolo è presente anche una tripla che indica che l'articolo sta citando un altro (attraverso la proprietà `cito:cites`). Quindi una citazione da una risorsa a un'altra è indicata in due modi: indirettamente tramite la risorsa riferimento bibliografico e direttamente.

In questo capitolo abbiamo parlato di alcune tecnologie che permettono di visualizzare dati RDF in una pagina HTML con particolare riferimento a metadati bibliografici e citazionali. Infine abbiamo descritto `OpenCitations`, che si occupa di raccogliere metadati di questo tipo, e abbiamo rappresentato le principali entità contenute all'interno del suo triplestore.





## Capitolo 2

### OCEB: panoramica

OCEB (OpenCitations Entities Browser) è stato sviluppato per permettere una visualizzazione di dati RDF su browser, in modo chiaro e compatto, facilitandone la lettura e la visione da parte dell'utente. I dati presentati riguardano entità bibliografiche e fanno parte del database di OpenCitations. Come anticipato nel capitolo precedente, OpenCitations è un progetto il cui scopo principale è quello di raccogliere informazioni citazionali di letture accademiche. Per sviluppare il software è stata utilizzata una libreria chiamata RSLT, che permette la trasformazione di dati RDF, tramite template dichiarativi.

Adesso andremo a descrivere i motivi che hanno spinto allo sviluppo di un'applicazione di questo genere. Analizzeremo, inoltre, quali informazioni delle diverse risorse di OpenCitations verranno mostrate e saranno presentate sul browser.

#### 2.1 Scopo dell'applicazione

Il progetto OpenCitations offre la possibilità, di accedere alle informazioni di una risorsa tramite il suo URI (ad esempio <https://w3id.org/oc/corpus/br/2>, che rappresenta un fascicolo).

bibliographic resource 2 [br/2]	
<a href="https://w3id.org/oc/corpus/br/2">https://w3id.org/oc/corpus/br/2</a>	
<b>is a</b>	document periodical issue
<b>number</b>	1
<b>part of</b>	<a href="https://w3id.org/oc/corpus/br/4">https://w3id.org/oc/corpus/br/4</a>

Figura 2.1: Esempio di visualizzazione di una risorsa in OpenCitations

La figura 2.1 mostra una particolare risorsa di OpenCitations, identificata dalla sigla br/2 e di tipo fascicolo. Questa rappresentazione include esclusivamente le triple che hanno come soggetto la risorsa in questione. In particolare, gli elementi posti sullo sfondo azzurro indicano le proprietà, mentre quelli posti sullo sfondo bianco i valori o le risorse legate a queste proprietà. La rappresentazione in HTML proposta può essere limitante per un utente che vuole ottenere informazioni correlate alla risorsa br/2. Ad esempio esistono degli articoli che fanno parte di questo fascicolo e ciò viene indicato attraverso delle triple di cui i soggetti sono gli articoli e gli oggetti la risorsa br/2. Questo comporta che queste informazioni non siano mostrate nella figura 2.1, in quanto la risorsa principale compare come oggetto. Naturalmente nella visualizzazione di quegli articoli questa indicazione sarà presente, infatti, verrebbero visualizzate tutte le triple in cui gli articoli sono il soggetto. Inoltre, è presente la proprietà "part of" seguita da un URI, il quale ci permette di ottenere informazioni sulla risorsa associata ad esso solo facendoci click sopra e ciò può risultare scomodo. Nasce così l'esigenza di sviluppare un software che sia in grado di offrire al lettore una rappresentazione HTML più semplice, completa, chiara e compatta della risorsa in questione. In aggiunta, si vogliono mostrare tutte quelle informazioni legate a una risorsa che su OpenCitations non vengono mostrate, come ad esempio gli articoli che fanno parte di un fascicolo.

Ora verrà fatto un confronto tra la visualizzazione nativa presente in Open-Citations e quella generata da OCEB, per rendere più chiare le differenze.

[Issue] **International Journal of Gynecological Cancer, 26 (1)**

**Identifiers:**

- OCC ID: br/2.

**Articles:**

- Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martín Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *International Journal of Gynecological Cancer*, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).
- Söelétormos György, Duffy Michael J., Othman Abu Hassan Suher, Verheijen René H.M., Tholander Bengt, Bast Robert C., Gaarenstroom Katja N., Sturgeon Catharine M., Bonfrer Johannes M., Petersen Per Hyltoft, Troonen Hugo, Carlo Torre Gian, Kanty Kulpa Jan, Tuxen Malgorzata K., Molina Raphael. Clinical Use of Cancer Biomarkers in Epithelial Ovarian Cancer: Updated Guidelines From the European Group on Tumor Markers. *International Journal of Gynecological Cancer*, 26 (1): 43-51. 2016. Ovid Technologies (Wolters Kluwer Health).

Figura 2.2: Esempio di visualizzazione di una risorsa con OCEB

La figura 2.2 presenta la stessa risorsa mostrata nella figura 2.1, con una visualizzazione completamente differente, grazie all'applicazione sviluppata. Infatti la visualizzazione della figura 2.2 è stata ottenuta applicando dei template specifici per questo tipo di risorsa, che verranno mostrati più nel dettaglio nel capitolo successivo. In questo modo si ottiene una pagina web in cui le informazioni sono organizzate in modo da rispettare uno schema che permette all'utente di reperire facilmente i dati che gli interessano. Ad esempio, essendo risorsa br/2 di tipo fascicolo, nella parte alta dell'immagine viene mostrata la rivista e il volume che la contengono, seguito dal numero del fascicolo stesso. In questo modo un utente abituato ad avere a che fare con riviste, volumi e fascicoli troverà familiarità con una visualizzazione delle informazioni di questo genere. Un'ulteriore differenza che si nota tra le due figure è che la seconda contiene più informazioni, questo è dovuto al fatto che i template dell'applicazione non si limitano a interpretare le triple del database in cui la risorsa br/2 è il soggetto, ma molte altre. Ad esempio questo permette di inserire nella pagina HTML informazioni riguardanti gli

articoli che fanno parte del fascicolo. Inoltre, avevamo notato che nella figura 2.1 fosse presente un URI, il quale non compare nella figura 2.2. L'URI in questione indica la risorsa, di tipo volume, che contiene il fascicolo br/2. Questa informazione viene sostituita nella figura 2.2, con l'indicazione del numero di quel volume, in questo caso 26, come si può notare nel titolo. Come vedremo in seguito, il numero 26 è comunque il testo di un link, il quale collega questa pagina alla pagina specifica di quel volume indicato con il numero 26, generata sempre dall'applicazione.

Abbiamo illustrato il motivo per cui è stata sviluppato OCEB e che vantaggi offra rispetto alla visualizzazione di risorse già presente in OpenCitations. Nel prossimo capitolo parleremo, più nel dettaglio, di come le informazioni delle diverse risorse bibliografiche vengano presentate e quali motivazioni ci sono alla base di quelle scelte.

## **2.2 Visualizzazione delle entità bibliografiche**

OCEB applica dei template RSLT diversi alle diverse risorse presenti nel triplestore di OpenCitations. I template RSLT ci permettono di fare query SPARQL e presentare i risultati ottenuti nella pagina HTML. Questa scelta è stata fatta per presentare il layout più adatto a ogni entità che può essere contenuta nel database. Ad esempio, non avrebbe senso utilizzare lo stesso layout per visualizzare una risorsa che corrisponde ad un autore e per visualizzarne un'altra che corrisponde a un articolo.

Di seguito illustreremo come vengono presentate dall'applicazione le principali entità bibliografiche, descritte in OpenCitations, ovvero: risorse bibliografiche, agenti, ruoli, identificativi, formati e riferimenti bibliografici.

## 2.2.1 Visualizzazione di risorse bibliografiche

Le risorse bibliografiche rappresentano i componenti centrali di OpenCitations. Ne esistono di moltissimi tipi, con differenze tra loro anche notevoli. A seguire verranno mostrate le visualizzazioni delle principali tipologie.

### Visualizzazione di articoli di una rivista

In questa sezione verrà descritta la visualizzazione di articoli di una rivista, una risorsa bibliografica molto ricca di contenuti.

[Article] **The Electronic Patient Reported Outcome Tool: Testing Usability and Feasibility of a Mobile App and Portal to Support Care for Patients With Complex Chronic Disease and Disability in Primary Care Settings**

By: Steele Gray Carolyn, Gill Ashlinder, Khan Anum Irfan, Hans Parminder Kaur, Kuluski Kerry, Cott Cheryl.

In: JMIR mHealth uHealth - JMIR mHealth and uHealth, 4 (2): e58-e58. 2016. JMIR Publications Inc.

**Identifiers:**

- DOI: [10.2196/mhealth.5331](https://doi.org/10.2196/mhealth.5331).
- URL: <http://dx.doi.org/10.2196/mhealth.5331>.
- PMID: [27256035](https://pubmed.ncbi.nlm.nih.gov/27256035/).
- PMCID: [PMC4911509](https://pubmed.ncbi.nlm.nih.gov/PMC4911509/).
- OCC ID: [br/525205](https://open.citations.isn.org/br/525205).

**Cited by:**

- Steele Gray Carolyn, Wodchis Walter P, Upshur Ross, Cott Cheryl, McKinstry Brian, Mercer Stewart, Palen Ted E, Ramsay Tim, Thavorn Kednapa. Supporting Goal-Oriented Primary Health Care for Seniors with Complex Care Needs Using Mobile Technology: Evaluation and Implementation of the Health System Performance Research Network, Bridgepoint Electronic Patient Reported Outcome Tool. JMIR mHealth uHealth - JMIR mHealth and uHealth, 5 (2): e126-e126. 2016. JMIR Publications Inc. [Show bibliography reference](#)

**References:**

- Abras, C, Maloney-Krichmar, D, Preece, J, Bainbridge, W. User-centered design, Encyclopedia of Human-Computer Interaction, 2004, Thousand Oaks, California, Sage Publications
- Al Ayubi Soleh U, Parmanto, B, Branch, R, Ding, D. A Persuasive and Social mHealth Application for Physical Activity: A Usability and Feasibility Study, JMIR Mhealth Uhealth, 2014, 2, 2, e25, DOI: 10.2196/mhealth.2902, 25099928, PMID: 25099928
- Appelboom, G, Yang, AH, Christophe, BR, Bruce, EM, Slomian, J, Bruyère, O, Bruce, SS, Zacharia, BE, Reginster, J, Connolly, ES. The promise of wearable activity sensors to define patient recovery, J Clin Neurosci, 2014, 7, 21, 7, 1089, 93, DOI: 10.1016/j.jocn.2013.12.003, 24534628, PMID: 24534628
- Bakhshi-Raiez, F, de Keizer N F, Cornet, R, Dorrepaal, M, Dongelmans, D, Jaspers M W M. A usability evaluation of a SNOMED CT based compositional interface terminology for intensive

Figura 2.3: Esempio di visualizzazione di un articolo

La figura 2.3 mostra una pagina web, generata grazie all'applicazione sviluppata, di un articolo, in particolare della risorsa con identificativo br/525205 di OpenCitations. Adesso andiamo a descrivere nel dettaglio la figura presentata, procedendo dall'alto verso il basso. Nel primo paragrafo è presente tra parentesi quadre il tipo di risorsa visualizzato seguito dal titolo e dal sottotitolo dell'articolo. Poi sono visualizzati gli autori del testo, introdotti

da "By", e inseriti seguendo l'ordine definito in OpenCitations. Il terzo paragrafo contiene diverse informazioni, quali la rivista, il volume e il fascicolo che contengono l'articolo, il formato, l'anno di pubblicazione e l'editore. A seguire sono mostrate tre categorie, introdotte da "Identifiers", "Cited By" e "References". La prima mostra l'elenco degli identificativi della risorsa con due tipi di collegamento: un riferimento alla pagina HTML dell'identificativo, facente parte dell'applicazione stessa, e un collegamento esterno per visualizzare la risorsa in un determinato motore di ricerca di quell'identificativo. Ad esempio per il DOI si accede alla pagina "<http://mhealth.jmir.org/>" e si ricerca all'interno di quella pagina la risorsa attraverso il DOI specifico dell'articolo. "Cited by" mostra tutte le risorse bibliografiche che citano l'articolo in questione, descrivendo per ognuna (anche se nell'esempio ne viene mostrata solo una) gli autori, il titolo, la gerarchia di rivista, volume, fascicolo, l'anno di pubblicazione e l'editore. Cliccando sul bottone accanto viene aperto un paragrafo che evidenzia il riferimento utilizzato dalla risorsa citante per citare l'articolo della pagina web. Infine, viene mostrato l'elenco dei riferimenti usati dall'articolo considerato per citare altre risorse, introdotto da "References".

### **Visualizzazione di un fascicolo**

Il fascicolo, contiene articoli scientifici, e a sua volta è contenuto in un volume e in una rivista. Per descriverne la sua visualizzazione, mediante OCEB, faremo riferimento all'esempio di figura 2.2. In alto è presente, così come lo era per gli articoli e lo sarà per le altre risorse, un riferimento tra parentesi quadre al tipo di risorsa presentata. Inoltre, è presente il titolo della rivista e il volume che contengono quelli del fascicolo. Poi vediamo l'elenco degli identificativi e gli articoli che sono contenuti dentro il fascicolo mostrato.

### **Visualizzazione di un volume**

Andremo ora a descrivere come viene presentato un volume, una risorsa bibliografica contenuta in una rivista e che, a sua volta, contiene dei fascicoli.

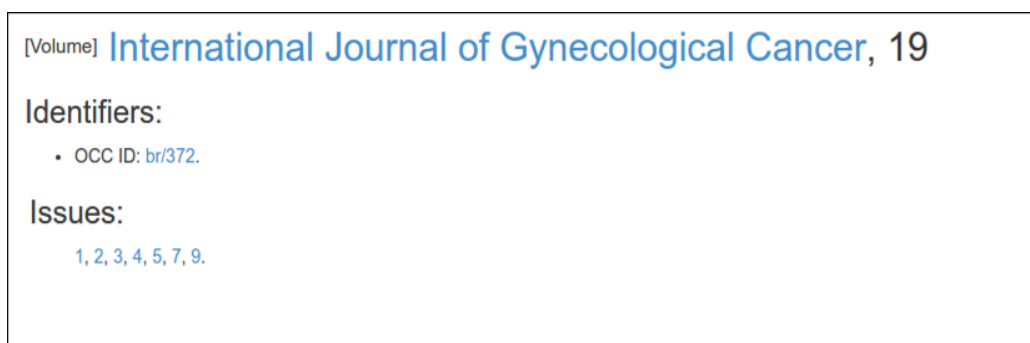


Figura 2.4: Esempio di visualizzazione di un volume

La figura 2.4 mostra un esempio di visualizzazione di una risorsa di tipo volume, in particolare della risorsa [br/372](#) di OpenCitations. In alto nell'immagine viene mostrato il titolo della rivista che lo contiene, seguito dal numero del volume che si sta considerando. Come per le risorse precedentemente descritte, sono elencati gli identificativi. L'ultimo elemento presentato è l'elenco dei fascicoli contenuti nel volume, presentati in ordine numerico crescente.

### **Visualizzazione di una rivista**

Seguendo la gerarchia di contenimento delle risorse bibliografiche, presentiamo la visualizzazione di una rivista.

[Journal] **International Journal of Gynecological Cancer**

**Identifiers:**

- ISSN: 1048-891X.
- OCC ID: br/3.

**Volumes:**

3, 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26.

**Articles:**

- Mabuchi Seiji, Matsumoto Yuri, Hamasaki Toshimitsu, Kawano Mahiru, Hisamatsu Takeshi, Mutch David G., Kimura Tadashi. Elevated White Blood Cell Count at the Time of Recurrence Diagnosis Is an Indicator of Short Survival in Patients With Recurrent Cervical Cancer. *International Journal of Gynecological Cancer*: 1-1. 2012. Ovid Technologies (Wolters Kluwer Health).
- Yamada Yoshihiko, Shigetomi Hiroshi, Onogi Akira, Haruta Shoji, Kawaguchi Ryuji, Yoshida Shozo, Furukawa Naoto, Nagai Akira, Tanase Yasuhiro, Tsunemi Taihei, Oi Hidekazu, Kobayashi Hiroshi. Redox-Active Iron-Induced Oxidative Stress in the Pathogenesis of Clear Cell Carcinoma of the Ovary. *International Journal of Gynecological Cancer*: 1-1. 2011. Ovid Technologies (Wolters Kluwer Health).
- Yamamoto Takuro, Mori Taisuke, Sawada Morio, Kuroboshi Haruo, Tatsumi Hiroshi, Yoshioka Takashi, Matsushima Hiroshi, Iwasaku Kazuhiro, Kitawaki Jo. Estrogen-Related Receptor- $\gamma$  Regulates Estrogen Receptor- $\alpha$  Responsiveness in Uterine Endometrial Cancer. *International Journal of Gynecological Cancer*: 1-1. 2012. Ovid Technologies (Wolters Kluwer Health).
- Zhang Jingjing, Zhao Jin, Zhang Wenjing, Liu Guanyuan, Yin Dongmei, Li Jian, Zhang Sumei, Li Hongxia. Establishment of Paclitaxel-Resistant Cell Line and the Underlying Mechanism on Drug Resistance. *International Journal of Gynecological Cancer*: 1-1. 2012. Ovid Technologies (Wolters Kluwer Health).

Figura 2.5: Esempio di visualizzazione di una rivista

Osservando la figura 2.5, che mostra la visualizzazione della risorsa br/3, si nota che il layout di questo tipo di risorsa è simile ai precedenti. Le differenze che si possono notare riguardano la presenza di un elenco di volumi e di articoli. I volumi sono ordinati, in maniera numericamente crescente, e rappresentano, naturalmente, i volumi contenuti all'interno della rivista. Sono presenti anche degli articoli nella figura, anche se non corrispondono a tutti gli articoli presenti nella rivista. Infatti, quelli mostrati rappresentano gli articoli direttamente inseriti dentro alla rivista, senza rispettare la gerarchia articolo-fascicolo-volume-rivista. Si è scelto di non elencare tutti gli articoli presenti, a qualsiasi livello di contenimento, dentro a una rivista, per evitare di presentare decine di migliaia di articoli, che avrebbero appesantito il caricamento della pagina web.

### Visualizzazione delle altre risorse bibliografiche

La visualizzazione delle altre risorse bibliografiche è abbastanza simile alle precedenti e descrivere ogni altro tipo di risorsa singolarmente, sarebbe stato abbastanza superfluo. Per questo si è scelto di accennare come vengono presentate le altre risorse, senza scendere nello specifico. Procedendo sempre dall'alto verso il basso vengono visualizzati il titolo, gli autori, la risorsa che



contiene quella che stiamo presentando, l'elenco degli identificativi, l'elenco dei documenti che citano e che vengono citati dalla risorsa in questione. Infine per alcune risorse vengono inseriti gli elementi che ne fanno parte. Ad esempio per i libri vengono presentati, se sono presenti, i capitoli di cui sono composto.

## 2.2.2 Visualizzazione di agenti

Ora, vediamo come una risorsa di tipo agente venga visualizzata attraverso OCEB.



[Author] Nicoletta Colombo

Identifiers:

- OCC ID: ra/1.

Author of:

- Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martín Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *International Journal of Gynecological Cancer*, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).

Figura 2.6: Esempio di visualizzazione di un agente

Consideriamo come esempio di visualizzazione di una risorsa di tipo agente, la figura 2.6, che mostra la risorsa ra/1, che rappresenta un autore. Oltre agli autori, come sottotipo di agente, sono presenti anche gli editori, che non mostreremo perché il layout rimane molto simile a quello mostrato nella figura. E' possibile osservare che per gli autori viene mostrato il nome e il cognome in alto. Più in basso sono presenti l'elenco degli identificativi e la lista degli articoli di cui il soggetto è autore. Questi articoli sono descritti presentando gli autori, il titolo e il sottotitolo, la rivista, il volume e il fascicolo che lo contengono, l'anno di pubblicazione e l'editore.

### 2.2.3 Visualizzazione di ruoli

Il ruolo è un'entità, definita in OpenCitations, che collega un agente con che esercita una certa funzione (ad esempio autore o editore) a una risorsa bibliografica. Vediamo, mediante un esempio, come venga presentato un ruolo.

[Role] Nicoletta Colombo as author of [ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer](#)

Resource: Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martin Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *International Journal of Gynecological Cancer*, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).

Role: author.

Identifiers:

- OCC ID: ar/1.

Figura 2.7: Esempio di visualizzazione di un ruolo

La figura 2.7 mostra un esempio di visualizzazione di una risorsa di tipo ruolo, in particolare della risorsa ar/1 di OpenCitations. Osservando la figura 2.7 si nota che nel primo paragrafo è indicato il nome e il cognome della persona che svolge il ruolo di autore di un determinato articolo, di cui è menzionato il titolo. Più in basso viene inserita la risorsa bibliografica, già presentata con il titolo, descrivendola in maniera completa. Viene poi ribadito il tipo di ruolo e alla fine, è presente il solito elenco di identificativi.

### 2.2.4 Visualizzazione di identificativi

Procediamo a descrivere la visualizzazione della successiva risorsa: gli identificativi.

[doi] [10.1097/igc.0000000000000609](https://doi.org/10.1097/igc.0000000000000609)

Identifier of: Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martin Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. International Journal of Gynecological Cancer, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).

Identifiers:

- OCC ID: id/1.

Figura 2.8: Esempio di visualizzazione di un identificativo

Nella figura 2.8 vediamo come viene presentata la risorsa id/1. In particolare la risorsa id/1 rappresenta un identificativo di tipo DOI. Nella parte superiore dell'immagine troviamo tra le parentesi quadre il tipo di identificativo seguito dal suo valore. Viene poi specificata l'entità che ha quel particolare identificativo, e infine, vengono presentati gli identificativi associati a questo DOI. A ogni risorsa di tipo identificativo è sempre associato un OCC (Open Citations Corpus) ID, che specifica l'URI di OpenCitations relativo alla risorsa visualizzata.

## 2.2.5 Visualizzazione di formati

Il formato indica principalmente da che pagina a che pagina di un fascicolo è presente una determinata risorsa bibliografica.

[Format] [ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: 2-30](#)

Format of: Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martin Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. International Journal of Gynecological Cancer, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).

Identifiers:

- OCC ID: ref1.

Figura 2.9: Esempio di visualizzazione di un formato

La figura 2.9 mostra la visualizzazione di una risorsa di tipo formato, più precisamente della risorsa re/1 di OpenCitations. Nel primo paragrafo è indicato il titolo della risorsa bibliografica che ha quel determinato formato, seguito dalla pagina di inizio e di fine. A seguire, viene descritta meglio e in modo completo la risorsa bibliografica e, alla fine, è presente il consueto elenco degli identificativi.

## 2.2.6 Visualizzazione di riferimenti bibliografici

Concludiamo questa parte descrivendo la visualizzazione dell'ultimo tipo di risorsa, ovvero il riferimento bibliografico.

[Bibliographic Reference] Lee, NK, Cheung, MK, Shin, JY. Prognostic factors for uterine cancer in reproductive-aged women. *Obstet Gynecol.* 2007; 109: 655–662. PMID: 17329517

Contained in: Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martín Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *International Journal of Gynecological Cancer*, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).

References: Lee Nita Karnik, Cheung Michael K., Shin Jacob Y., Husain Amreen, Teng Nelson N., Berek Jonathan S., Kapp Daniel S., Osann Kathryn, Chan John K.. Prognostic Factors for Uterine Cancer in Reproductive-Aged Women. *International Journal of Gynecological Cancer*, 109 (3): 655-662. 2007. Ovid Technologies (Wolters Kluwer Health).

Identifiers:

- OCC ID: be/2.


Figura 2.10: Esempio di visualizzazione di un riferimento bibliografico

La figura 2.10 ci mostra come viene presentato un riferimento bibliografico, visualizzando la risorsa be/2. In alto vediamo il testo, vero e proprio, del riferimento bibliografico. Viene poi indicata la risorsa che contiene questo particolare riferimento e la risorsa a cui esso si riferisce. Concludiamo come sempre, con l'elenco degli identificativi

In questa sezione abbiamo parlato della visualizzazione delle diverse entità bibliografiche presenti in OpenCitations, utilizzando come esempi dei contenuti di pagine HTML generati con l'applicazione.

## 2.3 Navigazione attraverso OCEB

L'applicazione, oltre a offrire la possibilità di visualizzare un'entità bibliografica, offre la possibilità di navigare da una risorsa a un'altra, costituendo un browser per i dati di OpenCitations. Infatti, tutti gli elementi testuali colorati di blu, mostrati nelle immagini precedenti, costituiscono dei link alle risorse descritte dal testo. Ad esempio, visualizzando una pagina relativa a un articolo, è possibile fare click su un autore per caricare la pagina HTML dell'applicazione riguardante quell'autore. Questo permette all'utente, di ricevere altre informazioni, indirettamente collegate alla risorsa che sta visualizzando. Vediamo ora un esempio di una possibile navigazione tra le pagine HTML dell'applicazione.



[Article] **ACOG Practice Bulletin #65: Management of Endometrial Cancer**  
In: *Obstetrics & Gynecology*, 106 (2): 413-425. 2005. Ovid Technologies (Wolters Kluwer Health).

Identifiers:

- DOI: [10.1097/00006250-200508000-00050](https://doi.org/10.1097/00006250-200508000-00050)
- URL: <http://dx.doi.org/10.1097/00006250-200508000-00050>
- PMID: [16055605](https://pubmed.ncbi.nlm.nih.gov/16055605/)
- OCC ID: [br/11](#)

Cited by:

- Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martín Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *Obstetrics & Gynecology*, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health). [Show bibliography reference](#)

Figura 2.11: Prima schermata di navigazione

Consideriamo di stare visualizzando la pagina dell'applicazione relativa alla risorsa br/11, mostrata nella figura 2.11. Oltre a prendere visione delle informazioni presenti nell'immagine potremmo essere interessati a scoprire altri dati non direttamente collegati con la risorsa principale. Come possiamo vedere, l'articolo in questione è citato da un altro articolo e nella sua descrizione sono inseriti i suoi autori. Quindi, potremmo essere interessati a capire quali altri articoli sono stati scritti da uno di quegli autori. Ad esempio, consideriamo che l'autore a cui siamo interessati sia il quarto, ossia

Tjalling Bosse. Non ci resta che fare click sul suo nome per visualizzare la pagina HTML dedicata a lui.

[Author] **Tjalling Bosse**

**Identifiers:**

- ORCID: 0000-0002-6881-8437. [g](#)
- OCC ID: ra/4.

**Author of:**

- Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martín Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana.ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *International Journal of Gynecological Cancer*, 26 (1): 2-30. 2016. Ovid Technologies (Wolters Kluwer Health).
- Bosse Tjalling, ter Haar Natajja T, Seeber Laura M, Diest Paul J v, Hes Frederik J, Vasen Hans FA, Nout RA, Creutzberg Carien L, Morreau Hans, Smit Vincent THBM.Loss of ARID1A expression and its relationship with PI3K-Akt pathway alterations, TP53 and microsatellite instability in endometrial cancer. *International Journal of Gynecological Cancer*, 26 (11): 1525-1535. 2013. Ovid Technologies (Wolters Kluwer Health).
- Bosse Tjalling, Nout R.A., Stelloo E., Dreef E., Nijman H.W., Jürgenliemk-Schulz I.M., Jobsen J.J., Creutzberg C.L., Smit V.T.H.B.M.L1 cell adhesion molecule is a strong predictor for distant recurrence and overall survival in early stage endometrial cancer: Pooled PORTEC trial results. *International Journal of Gynecological Cancer*, 50 (15): 2602-2610. 2014. Ovid Technologies (Wolters Kluwer Health).
- Bosse Tjalling, Peters Elke E.M., Creutzberg Carien L., Jürgenliemk-Schulz Ina M., Jobsen Jan J., Mens Jan Willem M., Lutgens Ludy C.H.W., van der Steen-Banasik Elzbieta M., Smit Vincent T.H.B.M., Nout Remi A..Substantial lymph-vascular space invasion (LVSI) is a significant risk factor for recurrence in endometrial cancer – A pooled analysis of PORTEC 1 and 2 trials. *International Journal of Gynecological Cancer*, 51 (13): 1742-1750. 2015. Ovid Technologies (Wolters Kluwer Health).

Figura 2.12: Seconda schermata di navigazione

Dopo aver cliccato sull'autore, si aprirà una pagina HTML, rappresentata nella figura 2.12. Questa figura mostra in particolare la risorsa ra/4 di OpenCitations. In questa pagina è presente una lista di articoli di cui Tjalling Bosse è autore e il primo di essi è lo stesso articolo che compariva nella figura 2.11. Ora che ci troviamo in questa pagina potremmo essere interessati a visualizzare la pagina relativa a un articolo, per esempio consideriamo che sia l'ultimo tra quelli elencati.

[Article] **Substantial lymph-vascular space invasion (LVSI) is a significant risk factor for recurrence in endometrial cancer – A pooled analysis of PORTEC 1 and 2 trials**

By: Bosse Tjalling, Peters Elke E.M., Creutzberg Carien L., Jürgenliemk-Schulz Ina M., Jobsen Jan J., Mens Jan Willem M., Lutgens Ludy C.H.W., van der Steen-Banasik Elzbieta M., Smit Vincent T.H.B.M., Nout Remi A.

In: *European Journal of Cancer*, 51 (13): 1742-1750. 2015. Elsevier BV.

**Identifiers:**

- DOI: [10.1016/j.ejca.2015.05.015](https://doi.org/10.1016/j.ejca.2015.05.015).
- URL: <http://dx.doi.org/10.1016/j.ejca.2015.05.015>.
- PMID: [26049688](https://pubmed.ncbi.nlm.nih.gov/26049688/).
- OCC ID: br/379.

**Cited by:**

- Colombo Nicoletta, Creutzberg Carien, Amant Frederic, Bosse Tjalling, González-Martín Antonio, Ledermann Jonathan, Marth Christian, Nout Remi, Querleu Denis, Mirza Mansoor Raza, Sessa Cristiana. ESMO-ESGO-ESTRO Consensus Conference on Endometrial Cancer: Diagnosis, Treatment and Follow-up. *European Journal of Cancer*, 26 (1): 2-30. 2016. Elsevier BV.

[Show bibliography reference](#)

Figura 2.13: Terza schermata di navigazione

Dopo aver fatto click sul titolo dell'articolo che volevamo visualizzare, ci comparirà la pagina HTML relativa a quell'articolo, illustrata nella figura 2.13. La schermata presentata fa riferimento alla risorsa br/379.

Quindi, navigando tramite i link presenti nelle pagine web, siamo passati da una pagina dedicata a un articolo, a un'altra riferita a un autore di un articolo che cita il precede, fino alla pagina rappresentante un articolo di quell'autore. Questo esempio di navigazione mostra come l'applicazione sviluppata costituisca un browser vero e proprio di dati presenti in OpenCitations.

In questo capitolo abbiamo mostrato quali funzioni svolga OCEB, come vengano visualizzate le diverse risorse bibliografiche di OpenCitations e abbiamo motivato il perché l'applicazione costituisca un browser.





## Capitolo 3

# OCEB: dettagli implementativi

In questo capitolo verrà spiegato, nel dettaglio, in che modo le informazioni citazionali di OpenCitations vengano visualizzate attraverso i template RSLT dell'applicazione. Si procederà descrivendo le principali funzionalità di RSLT che sono state utilizzate per realizzare OCEB, mediante degli esempi per rendere più chiare le spiegazioni. Verranno poi illustrati i principali template RSLT utilizzati da OCEB e, infine, si parlerà di alcune funzioni javascript utilizzate da supporto ai template.

Per una migliore gestione dei template e per semplicità è stato scelto di dividere la visualizzazione delle diverse entità bibliografiche in sei diverse pagine HTML. Attraverso un parametro, chiamato "var", dichiarato nell'URL di una pagina dell'applicazione è possibile inserire la risorsa di OpenCitations che si vuole visualizzare. Ad esempio la pagina HTML che gestisce risorse bibliografiche si chiama "Bibl.html" e se volessimo visualizzare in quella pagina la risorsa "https://w3id.org/oc/corpus/br/1" dovremmo specificare nel parametro var quella stringa. Quindi, l'URL per visualizzare quella risorsa in quella pagina sarà "Bibl.html?var=https://w3id.org/oc/corpus/br/1". Una funzione javascript permetterà di ottenere il valore del parametro var che verrà memorizzato in una variabile chiamata rootName. Questa variabile permetterà di identificare la risorsa principale che sarà la prima a essere

utilizzata dai template.

### 3.1 RSLT (RDF Stylesheet Language Transformations)

Descriviamo, ora, le funzionalità di RSLT, limitandoci a descrivere quelle utilizzate per lo sviluppo di OCEB. Procediamo per esempi in modo da rendere più semplice la spiegazione di quello che fanno i template RSLT.

```
1 <template name="start">
2   <calltemplate name="template2"> </calltemplate>
3 </template>
4
5 <template name="template2">
6   <h1> Template RSLT </h1>
7 </template>
```

codice 3.1: Primo esempio di template RSLT

Il codice 3.1 mostra due template. Il template con name="start" è il primo ad essere eseguito, infatti la parola chiave "start" ci dà la sicurezza che quel template sia quello di partenza. Al suo interno troviamo un elemento calltemplate, con l'attributo name="template2". Questo elemento permette di eseguire il template con il nome uguale a template2. Quindi una volta chiamato il secondo template esso inserirà un titolo h1 con scritto Template RSLT.

Supponiamo che nel triplestore che stiamo interrogando siano presenti queste due triple, descritte in formato Turtle (<https://www.w3.org/TR/turtle/>).

```
1 <http://example.com/risorsa> rdf:type articolo .
2 <http://example.com/risorsa> titolo "Tesi RSLT" .
```

Consideriamo ora un esempio più complesso del precedente, mostrato nel codice 3.2.

```
1 <template name="start">
```

```

2   <at select="??a rdf:type articolo" mode="template2"> </at>
3 </template>
4
5 <template match="?a -> articolo" mode="template1">
6   <h1> Non compare </h1>
7 </template>
8
9 <template match="?a -> articolo" mode="template2">
10  <h1> {{a["titolo"]}} </h1>
11 </template>

```

codice 3.2: Secondo esempio di template RSLT

Il primo template a essere eseguito è quello con name="start", il quale attraverso l'elemento at (Apply Template) esegue una query SPARQL. L'elemento at, infatti, permette di fare una query SPARQL in cui verranno trovate tutte le triple che hanno come soggetto la variabile preceduta dai due punti interrogativi e che rispettano la query inserita all'interno del select. In questo caso la query SPARQL che verrà fatta sarà questa:

```

1 SELECT ?a ?p ?o
2   WHERE {
3     ?a rdf:type articolo .
4     ?a ?p ?o .
5   }

```

codice 3.3: Esempio di query da RSLT

In questo modo come risultato otterremo tutte le triple presenti nel nostro triplestore. L'attributo mode dell'elemento at ci permette di filtrare i template in grado di catturare i risultati, considerando solo quelli che possiedono l'attributo mode="template2". Il secondo template del codice 3.2 non viene eseguito perché il valore del suo attributo mode non coincide con quello dell'apply template. Il terzo template, invece, è in grado di catturare i risultati. L'attributo match specifica una variabile, in questo caso "a", che è in grado di fare match con i risultato di un apply template il cui tipo (rdf:type) coincida con quello dichiarato dopo la freccia ( $\rightarrow$ ). Quindi l'ultimo template riesce a catturare i risultati della query SPARQL, visto che fa match con risorse

di tipo articolo. Viene eseguito quel template ed esso inserirà un titolo h1 al cui interno sarà presente il valore "Tesi RSLT". Questo avviene perché grazie alla variabile definita del match del template si può accedere ai valori delle proprietà della risorsa trovata, seguendo lo schema a["proprietà"].

In questa sezione abbiamo parlato di RSLT e illustrato attraverso due esempi le funzionalità dei template, limitandoci a presentare le principali che sono state utilizzate per OCEB. Procediamo ora illustrando i più significativi template utilizzati dall'applicazione web.

## **3.2 Descrizione dei template RSLT**

I template RSLT, utilizzati per ottenere informazioni dal database e per presentarle, svolgono un ruolo centrale nell'applicazione sviluppata. Essi, infatti, permettono di fare delle query SPARQL direttamente al database di OpenCitations, scaricare le informazioni ottenute, in formato XML, e utilizzarle o per fare altre query a partire da esse o per presentare dei dati in una certa sezione della pagina HTML. Andiamo a vedere come si comportano i template, descrivendone alcuni presenti nelle pagine HTML di OCEB.

### **3.2.1 Template per visualizzare le risorse bibliografiche**

I template che andremo a descrivere sono utilizzati per visualizzare risorse bibliografiche, come ad esempio articoli, fascicoli, volumi e riviste.

#### **Template per riconoscere il tipo di risorsa bibliografica**

Questi template vengono utilizzati nella pagina web preposta alla presentazione di risorse bibliografiche. Il loro scopo è di identificare che tipo di risorsa bibliografica debba essere rappresentata, in modo da applicare tem-

plate diversi a seconda del tipo. Ad esempio, è utile per applicare template e presentazioni diverse se si vuole visualizzare un fascicolo o un articolo.

```
1 <template name="main" >
2   <at select="{{rootName}}" mode="main"/>
3 </template>
4
5 <template match = "?a -> fabio:JournalArticle" mode="main">
6   <h2>[ Article]</h2>
7   <!--vengono applicati template specifici
8   per gli articoli -->
9 </template>
10
11 <template match = "?a -> fabio:JournalIssue" mode="main">
12   <h2>[ Issue]</h2>
13   <!-- vengono applicati template specifici
14   per i fascicoli -->
15 </template>
16
17 <template match = "?a -> fabio:Journal" mode="main">
18   <h2>[ Journal]</h2>
19   <!-- vengono applicati template specifici
20   per le riviste -->
21 </template>
```

codice 3.4: Template per riconoscere le diverse risorse bibliografiche

Il codice 3.4 mostra i template relativi a questa operazione. Il primo template chiamato nel codice 3.4 è quello di nome "main", il quale applica un altro template, attraverso l'elemento `ap`, ovvero `apply template`. Esso effettua una query dove seleziona `rootName`, che corrisponde alla risorsa specificata dal parametro "var" nell'indirizzo della pagina. L'attributo `select` di un template ci permette di fare una query SPARQL i cui risultati saranno catturati da un altro template. Per precisione, il template che cattura quei risultati è quello che possiede un attributo `match` il cui tipo specificato dopo la freccia corrisponde al tipo del risultato trovato. Quindi, verranno trovate le triple che

hanno come soggetto la risorsa specificata da `rootName`. A seconda del tipo della risorsa, indicato dalla proprietà `rdf:type` nel database di `OpenCitations`, uno dei template sottostanti catturerà il risultato della prima query. Se la risorsa principale sarà un articolo verrà fatto il match con il secondo template, se sarà un fascicolo con il terzo e se sarà una rivista con il quarto. In questo modo è possibile assegnare dentro a ogni template delle azioni differenti, ad esempio ogni template inserisce un titolo diverso e applica a sua volta template differenti. Bisogna sottolineare che nel codice 3.4 sono stati elencati solo tre template che gestiscono risorse differenti, ossia articoli, fascicoli e riviste, mentre nel codice dell'applicazione è presente un template per ogni tipo di risorsa bibliografica definita in `OpenCitations`.

## Template utilizzati per la visualizzazione di un articolo

Andiamo adesso a descrivere il template RSLT principale che viene applicato a una risorsa bibliografica di tipo articolo.

```
1 <template match = "?a -> fabio:JournalArticle"
2   mode="main" priority="2">
3   <h2>
4     <font>[ Article ]</font>
5     <span ng-if="a['fabio:hasSubtitle']">:</span>
6     {{a['fabio:hasSubtitle']}}
7   </h2>
8   <calltemplate name="autori"> </calltemplate>
9   <p>
10    <calltemplate name="part"> </calltemplate>.
11    {{a['fabio:hasPublicationYear']}}.
12    <at select ="{{a['uri']}} pro:isDocumentContextFor ??c"
13      preload="??c pro:withRole
14        pro:publisher."mode="publisher"> </at>
15  </p>
16  <calltemplate name="id"> </calltemplate>
17  <calltemplate name="cit"> </calltemplate>
18  <calltemplate name="ref"> </calltemplate>
```

### codice 3.5: Template principale degli articoli

Questo template viene applicato esclusivamente agli articoli, questo grazie al template precedente, che consente di riconoscere la risorsa bibliografica che si vuole visualizzare nella pagina HTML. Osservando il codice 3.5 si vede che alcune informazioni vengono già presentate all'interno del template, mentre per ricavarne altre vengono applicati, a loro volta, altri template. Procediamo descrivendo nel dettaglio quello che avviene nel codice. All'interno del tag `h2` vengono già presentati dei dati, infatti, è presente il testo `[Article]` seguito dal titolo e dal sottotitolo. Infatti la dicitura `a[‘dcterms:title’]` ci permette di accedere al contenuto della proprietà `dcterms:title` dell'articolo in questione e in modo speculare verrà inserito il sottotitolo. L'elemento `span` con l'attributo `ng-if=’a[fabio:hasSubtitle]’` è utile per inserire tra il titolo e il sottotitolo il carattere dei due punti solo se effettivamente è presente il sottotitolo. Infatti, risulterebbe sgradevole, negli articoli sprovvisti di sottotitolo, visualizzare il titolo terminare con i due punti. Vengono poi chiamati diversi template, attraverso l'elemento `calltemplate` di RSLT, che verranno usati per inserire all'interno della pagina HTML gli autori, la gerarchia di contenimento, gli identificativi, le citazioni fatte e quelle ricevute dall'articolo. Dopo la chiamata al template `”part”` viene inserito l'anno di pubblicazione della risorsa e viene applicato un template per ricercare l'editore dell'articolo.

Ora che abbiamo introdotto il template principale che gestisce gli articoli, andiamo a descrivere i template che vengono chiamati al suo interno.

### **Template per inserire l'elenco degli autori di una risorsa bibliografica**

In questa sezione andremo a descrivere i template usati per inserire gli autori di una risorsa bibliografica. Come abbiamo visto nella sezione precedente questo template viene invocato per la presentazione di un articolo. Comunque, gli articoli non sono le uniche risorse bibliografiche che utilizzano questo

template, infatti, anche risorse come articoli su atti di conferenza, capitoli e libri lo utilizzano. In generale è stato scelto di inserire dei template specifici nelle sole risorse che possono attraverso essi ricevere delle informazioni aggiuntive. Ritornando al tema di questa sezione, risorse come fascicoli, volumi e riviste non avendo autori non necessitano l'uso di questo template, al contrario, per tutte le altre risorse, che possono avere quei dati, viene utilizzato.

```

1 <template name=" autori">
2   <at select=" {{a['uri']}} pro:isDocumentContextFor ??c"
3     preload="??c pro:withRole pro:author .
4     FILTER NOT EXISTS{ ?x oc:hasNext ??c }"
5     mode=" ruoli"> </at>
6 </template>
7
8 <template match= "?c -> pro:RoleInTime" mode=" ruoli">
9   <p id="by">By:
10     <at select = "<{{c['pro:isHeldBy']}}>" /> </at>
11     <at select = " {{c['uri']}} oc:hasNext ??b"
12       mode=" ruoli2"> </at>.
13   </p>
14 </template>
15
16 <template match= "?c -> pro:RoleInTime" mode=" ruoli2">
17   <at select = "<{{c['pro:isHeldBy']}}>" /> </at>
18   <at select = " {{c['uri']}} oc:hasNext ??b"
19     mode=" ruoli2"> </at>
20 </template>
21
22 <template match= "?c -> foaf:Agent" >
23   <a href=" Agent.html?var={{c['uri']}} ">
24     {{c['foaf:familyName']}}
25     <span ng-if="c['foaf:givenName']">
26       {{c['foaf:givenName']}}
27     </span>
28   </a>

```



codice 3.6: Template per elencare gli autori di una risorsa bibliografica

Entrando nello specifico, possiamo osservare il codice 3.6, che mostra ben quattro diversi template utilizzati per elencare tutti gli autori di una risorsa. Descriviamo il codice dall'alto verso il basso, il che rispetta anche l'ordine di chiamata dei diversi template. Nel codice 3.5 è presente un elemento `calltemplate` con attributo `name="autori"` e questo significa che verrà chiamato il template con `name="autori"` il quale è il primo template mostrato nel codice 3.6. Esso si limita a fare una query per determinare il primo autore. Nello specifico la query fa una `select` sulla risorsa "c" che corrisponde a un ruolo relazionato con la risorsa bibliografica principale e attraverso l'attributo `preload` selezioniamo l'unico ruolo di tipo autore che non è il successivo di nessun altro ruolo. In questo modo siamo sicuri di aver identificato il primo ruolo, perché tutti gli altri hanno un ruolo che li precede. L'attributo `mode="ruoli"`, dell'elemento `at`, ci permette di essere sicuri che il secondo template del codice 3.6 sarà l'unico in grado di catturare i risultati della prima query. Il template con `mode="ruoli"` non fa altro che inserire un paragrafo al cui interno eseguirà due query: la prima per trovare il nome e cognome associato al ruolo che ha fatto il match con il template; la seconda per determinare il ruolo successivo definito dalla proprietà `oc:hasNext`. È facile notare che i risultati della prima query del secondo template verranno catturati dall'ultimo template, essendo l'unico che fa match con risorse di tipo `Agent`, mentre i risultati della seconda query verranno catturati dal terzo template. Esso è molto simile al precedente template, infatti farà le medesime due query, la differenza è che non inserisce un nuovo paragrafo.

### Template per inserire la gerarchia di contenimento

Il template utilizzato per inserire la gerarchia di contenimento è utile per ottenere informazioni sulle risorse bibliografiche che contengono la risorsa che stiamo visualizzando. È applicato a tutte quelle risorse che fanno parte

parte di un'altra risorsa, come ad esempio i capitoli di un libro, gli articoli di una rivista e i fascicoli di un volume.

```
1 <template name="part">
2   <span id="in">In: </span>
3   <at select = "{{a['uri']}} core:partOf ??d"
4     mode="part2"> </at>
5 </template>
6
7 <template match= "?d -> fabio:JournalIssue" mode="part2"
8   priority="2">
9   <at select= "{{d['uri']}} core:partOf ??c" mode="part2">
10  </at>
11  <a class="issue" href= "./Bibl.html?var={{d['uri']}}">
12    ({{d['fabio:hasSequenceIdentifier']}})
13  </a>
14 </template>
15
16 <template match= "?d -> fabio:JournalVolume" mode="part2"
17   priority="2">
18   <at select= "{{d['uri']}} core:partOf ??c" mode="part2">
19   </at>,
20   <a class="volume" href= "./Bibl.html?var={{d['uri']}}">
21     {{d['fabio:hasSequenceIdentifier']}}
22   </a>
23 </template>
24
25 <template match= "?d -> fabio:Journal" mode="part2"
26   priority="2">
27   <a class="journal" href= "./Bibl.html?var={{d['uri']}}">
28     {{d["dcterms:title"]}}
29   </a>
30 </template>
```

codice 3.7: Template per inserire la gerarchia di contenimento

Il codice 3.7 descrive come è composto questo template, anche se, per sinteticità, si è scelto di mostrare la sola gestione della gerarchia articolo, fascicolo, volume e rivista. Il template con name="part" è esattamente quello chiama-

to dal codice 3.5, il quale inserisce "In: ", che specifica che stiamo inserendo una risorsa che contiene quella di partenza e questo verrà fatto grazie all'apply template (at) che è posto successivamente con attributo mode = "part2". Il risultato del primo at può fare match con il secondo, terzo o quarto template mostrato dal codice 3.7, a seconda di che tipo di risorsa abbiamo ottenuto. Nello specifico, il secondo template fa match con risorse di tipo fascicolo, il terzo con risorse di tipo volume e il quarto con risorse di tipo rivista. Questi template presentano un valore che indica il numero del fascicolo o del volume oppure il titolo della rivista. E' interessante osservare che i template che catturano fascicoli e volumi, inoltre, applicano un template che esegue una query per determinare se a loro volta esiste una risorsa contenitiva per quel fascicolo o per quel volume. In pratica, questo permette a partire da un articolo di ottenere informazioni sul fascicolo che lo contiene, e a sua volta sul volume che contiene il fascicolo e, infine, sulla rivista che contiene il volume. Per rendere meglio l'idea possiamo vedere quali template verranno applicati a un articolo. Il primo template chiamato, del codice 3.7, è sicuramente quello con name="part", infatti viene chiamato direttamente dal template che gestisce la risorsa articolo. Questo template inserisce il testo "In" e applica un template che cerca quale risorsa lo contenga. Nella situazione più comune un articolo è contenuto in un fascicolo, poi in un volume e in una rivista. Quindi avremmo come risultato di quella query un fascicolo che farà match con il secondo template ed esso farà una query per trovare la risorsa che contiene quel fascicolo e dopo di che inserirà la sua sequenza. La risorsa trovata dall'elemento at sarà un volume, che farà match con il terzo template, il quale farà la consueta query per determinare la risorsa che lo contiene, e inserirà, anche esso, la sequenza specifica del volume. Per finire, l'ultima risorsa trovata sarà una rivista che farà match con il quarto template, che inserirà il suo titolo. Applicati tutti questi template, otteniamo, partendo da un articolo, il titolo della rivista, il numero di volume e il numero dal fascicolo che lo contengono. Il risultato sarebbe molto simile se fossimo partiti da un fascicolo o da un volume, semplicemente, si sarebbero applicati meno

template e si sarebbero ottenute meno informazioni. Da notare come nel codice appena mostrato siano presenti dei link, questi offrono la possibilità di visualizzare la pagina HTML, generata dai template presenti nelle pagine di OCEB, di quelle risorse descritte dal testo dei link.

## Template per inserire l'editore

Discutiamo ora del template utilizzato per inserire l'editore di una risorsa nella pagina HTML dell'applicazione.

```
1 <template match= "?c -> pro:RoleInTime" mode="publisher">
2   <at select = "<{{c ['pro:isHeldBy ' ]}}>" mode="publisher2" />
3 </template>
4
5 <template match= "?c -> foaf:Agent" mode="publisher2">
6   <span class="publisher">
7     <a href="Agent.html?var={{c ['uri ' ]}}">
8       {{c ['foaf:name ' ]}}
9     </a>.
10  </span>
11 </template>
```

codice 3.8: Template per inserire il nome dell'editore di una risorsa bibliografica

I template mostrati nel codice 3.8 hanno la finalità di inserire il nome dell'editore nella pagina della risorsa visualizzata attraverso l'applicazione. Il primo template mostrato è il primo a essere eseguito e semplicemente ricerca un elemento di tipo agente collegato alla risorsa bibliografica mediante la proprietà `pro:isHeldBy`. Il secondo template fa match con i risultati del precedente apply template e inserisce un link che ha come valore il nome dell'editore. Naturalmente, facendo click su quel link verrà caricata la pagina, generata dall'applicazione, relativa all'editore.

## Template per inserire gli identificativi

Questo template permette di inserire una lista di identificativi associati a un'entità bibliografica o a un agente.

```
1 <template name="id">
2   <h3>Identifiers:</h3>
3   <ul> <li id="doi">
4     <at select = "{{a['uri']}} datacite:hasIdentifier ??d"
5       preload="??d datacite:usesIdentifierScheme
6         datacite:doi." mode="id_doi"> </at>
7   </li>
8   <li class="url">
9     <at select = "{{a['uri']}} datacite:hasIdentifier ??d"
10      preload="??d datacite:usesIdentifierScheme datacite:url"
11      mode="id_url"> </at>
12   </li> </ul>
13 </template>
14
15 <template match= "?d -> datacite:Identifier" mode="id_doi">
16   <span ng-if=" first">DOI:</span>
17   <span ng-if="! first">, </span>
18   <a href=" Ident.html?var={{d['uri']}}">{{d['essepuntato:
19     literalreification/hasLiteralValue ']]}}</a>.
20   <a href=" http://dx.doi.org/
21     {{d['essepuntato:literalreification/
22       hasLiteralValue ']]}}">
23     
25   </a>
26 </template>
27
28 <template match= "?d -> datacite:Identifier" mode="id_url">
29   <span ng-if=" first">URL: </span>
30   <span ng-if="! first">, </span>
31   <a href=" Ident.html?var={{d['uri']}}">{{d['essepuntato:
32     literalreification/hasLiteralValue ']]}}</a>.
33   <a href=" {{d['essepuntato:literalreification
34     /hasLiteralValue ']]}}">
```

```

35     
37     </a>
38 </template>

```

codice 3.9: Template per inserire una lista di indentificativi

Consideriamo i template presenti nel codice 3.9. Il primo template, con name="id", è il primo ad essere chiamato e inserisce nella pagina HTML il titolo "Identifiers:" e poi ricerca due tipi di identificativi, ovvero DOI e URL, facendo la select attraverso due elementi at di RSLT. Gli identificativi trovati di tipo DOI fanno match con il secondo template, mentre quelli di tipo URL con il terzo template. Questi ultimi aggiungono una dicitura per indicare di che tipo di identificativo si tratti (DOI o URL) e poi aggiungono il valore dello stesso. Nel secondo elemento "a" di ognuno di essi viene inserito un collegamento a un motore di ricerca per quel particolare tipo di identificativo. Facendo click sull'immagine inserita dentro all'elemento "a", quindi, potremmo visualizzare la risorsa ricercandola in un sito attraverso un suo specifico identificativo. Naturalmente, anche il codice 3.9, è stato semplificato per rendere al meglio l'idea alla base di questi template, infatti, vengono presentati soltanto due tipi di identificativi, ma in realtà sono molti di più.

### **Template per inserire l'elenco delle risorse bibliografiche che citano la risorsa che stiamo visualizzando**

Andremo ora a presentare e a discutere dei template utilizzati per elencare le risorse bibliografiche che citano la risorsa principale della pagina HTML.

```

1 <template name="cit">
2     <ul id="citedBy">
3         <at select="??b biro:references {{a['uri']}}"
4             mode="be1"> </at>
5     </ul>
6 </template>
7

```

```

8 <template match= "?c ->biro:BibliographicReference" mode="be1">
9   <span ng-if="first">
10     <h3 id="citedBy" class="left"> Cited by: </h3>
11   </span>
12   <li class="cites">
13     <at select="??b core:part {{c['uri']}}" mode="cites">
14     </at>
15   </li>
16 </template>

```

codice 3.10: Template per inserire le risorse che citano la risorsa principale

Considerando il codice 3.10 notiamo che il primo template viene chiamato direttamente dal codice 3.5. Esso crea una lista, utilizzando l'elemento "ul", e all'interno di essa verrà fatta una query, specificata dall'attributo select dell'elemento apply template (at). La query non farà altro che trovare i riferimenti bibliografici diretti verso la risorsa principale della pagina dalla proprietà biro:references. In questo modo tutti i risultati della query faranno il match con il secondo template del codice 3.10. L'elemento span con proprietà ng-if="first" ci permette di aggiungere il titolo "Cited by:" solo una volta, ovvero quando verrà fatto il match con la prima bibliographicReference trovata. Proseguendo la lettura del codice, vediamo che più in basso viene aggiunto un elemento della lista e all'interno di esso viene fatta una query per ottenere le risorse bibliografiche che possiedono quella particolare reference. Non andremo ad approfondire su quale template farà match con i risultati dell'ultima query, diciamo solo che verranno applicati dei template per presentare in ordine: la lista degli autori, il titolo e sottotitolo, la gerarchia contenitiva, il formato, l'anno e l'editore di ogni risorsa trovata.

### Template per inserire l'elenco dei riferimenti bibliografici

Questo template ha l'obiettivo di inserire i riferimenti bibliografici della risorsa che stiamo visualizzando.

```

1 <template name="ref">
2   <ul id="ref">

```

```

3     <at select = "{{a['uri']}} core:part ??c" mode="ref2" >
4     </at> </ul>
5 </template>
6
7 <template match= "?d->biro:BibliographicReference" mode="ref2">
8     <li>
9         <a href="Bibl.html?var={{d['biro:references']}}">
10            {{d['c4o:hasContent']}}
11        </a>
12    </li>
13 </template>

```

codice 3.11: Template per inserire i riferimenti bibliografici

Il codice 3.11 ci mostra due template utilizzati per inserire nella pagina HTML l'elenco dei riferimenti bibliografici di una risorsa. Il template con name="ref" è il primo a essere chiamato, crea una lista e applica un template ricercando le risorse legate dalla proprietà core:part alla risorsa bibliografica. Il secondo template fa match con i risultati trovati e viene inserito, per ognuno di essi, un elemento della lista con un link, il cui contenuto è il testo della references e il collegamento è alla pagina web della risorsa a cui si riferisce la reference. Il link, come si può vedere, è alla pagina Bibl.html, che è la pagina dell'applicazione che gestisce risorse bibliografiche, con un parametro "var" posto uguale all'uri della risorsa a cui è collegato il riferimento bibliografico.

### 3.2.2 Template per visualizzare gli agenti

Procediamo ora con la descrizione dei template presenti sulla pagina HTML dedicata alla visualizzazione di agenti.

```

1 <template name="main">
2     <at select = "{{rootName}}}" mode="author"> </at>
3 </template>
4
5 <template match= "?a -> foaf:Agent" mode="author" >
6     <span ng-if="a['foaf:familyName']">
7         <h2> <font>[Author]</font>{{a["foaf:givenName"]}}

```



```

8         {{a[" foaf:familyName"]}} </h2>
9     </span>
10
11     <span ng-if="!a[' foaf:familyName ']">
12         <h2> <font> [Publisher]</font> {{a[' foaf:name']}}
13         </h2>
14     </span>
15
16     <calltemplate name="id"> </calltemplate>
17     <calltemplate name="author"> </calltemplate>
18 </template>
19
20 <template name="author">
21     <span ng-if="a[' foaf:familyName ']">
22         <h3> Author of: </h3>
23         <ul><at select="??c pro:isDocumentContextFor ?b"
24             preload="?b pro:isHeldBy {{a[' uri ' ]}}.
25             ??b pro:withRole pro:author." mode="cites"> </ul>
26     </span>
27 </template>

```

codice 3.12: Template principale degli agenti

Il codice 3.12 mostra i template utilizzati per gestire una risorsa di tipo agente. Il primo template è il primo a essere applicato e semplicemente ne applica un altro per trovare le triple in cui la risorsa principale è soggetto. Queste triple vengono catturate dal secondo template che, grazie all'attributo `ng-if` dell'elemento `span`, ci divide in due casistiche. O la risorsa è un autore e quindi vengono inseriti nella pagina web il suo nome e cognome oppure se è un editore viene inserito il suo nome. Vengono poi chiamati due template, `id`, che abbiamo già visto, e `author` che inserisce gli articoli di cui l'agente è autore. Infatti il terzo template del codice è proprio questo, il quale determina prima se la risorsa che ha fatto match sia un autore e non un editore, filtrando sulle risorse che possiedono l'attributo `foaf:familyName`, ovvero solo gli autori. L'ultimo apply template ricerca le risorse bibliografiche scritte eventualmente dall'autore centrale della pagina HTML.

### 3.2.3 Template per visualizzare i ruoli

Descriviamo ora quali template vengano utilizzati per visualizzare una risorsa di questo tipo.

```
1 <template name="main">
2   <at select="{{rootName}}" mode="role" />
3 </template>
4
5 <template match="?j -> pro:RoleInTime" mode="role">
6   <h2> <font>[Role]</font>
7   <at select="{{j['uri']}} pro:isHeldBy ??b"
8     mode="agent"> </at> as {{j["pro:withRole"]}} of
9   <at select="??a pro:isDocumentContextFor {{j['uri']}}"
10     mode="title" /> </at> </h2>
11   <p> Resource: <at select="{{j['uri']}}
12     ^pro:isDocumentContextFor ??a" mode="cites"> </at> </p>
13   <p> Role: <span id="role2"> </span> </p>
14 </template>
```

codice 3.13: Template principale dei ruoli

Considerando il codice 3.13 si evince che il risultato del primo apply template faccia match con il secondo template. Esso inserisce la dicitura [Role] e applica un template che inserirà il nome dell'agente che svolge quel ruolo, poi verrà specificato il ruolo e viene applicato un secondo template per determinare il titolo della risorsa bibliografica a cui si riferisce il ruolo.

### 3.2.4 Template per visualizzare gli identificativi

Andiamo a presentare i template principali utilizzati per presentare un risorsa di tipo identificativo.

```
1 <template name='start' >
2   <calltemplate name="main" renderTo=".body"> </calltemplate>
3   <calltemplate name="body" renderTo=".body"> </calltemplate>
4 </template>
5
6 <template name="main">
```

```

7     <at select = "{{rootName}}" mode="self"> </at>
8 </template>
9
10 <template match= "?a -> datacite:Identifier" mode="self" >
11     <h2>
12         [{{a['datacite:usesIdentifierScheme ']]}}]
13         {{a['esse:hasLiteralValue ']]}}
14     </h2>
15 </template>
16
17 <template name="body">
18     <p>Identifier of:
19         <at select="??a datacite:hasIdentifier {{rootName}}"
20             mode="cites">
21     </at> </p>
22 </template>

```

codice 3.14: Template principale degli identificativi

Il codice 3.14 è composto da un template principale con name="start", il quale chiama due template "main" e "body". Il template main fa una query per trovare le triple di cui la risorsa principale è soggetto. Questo risultato fa match con il terzo template, presentato nel codice 3.14, che inserisce nella pagina HTML il tipo di identificativo in questione, ad esempio DOI, e il suo valore. Il template con name="body" inserisce la scritta "Identifier of: " e a seguire la descrizione della risorsa bibliografica rappresentata dall'identificativo in questione.

### 3.2.5 Template per visualizzare i formati

```

1 <template name="main">
2     <at select = "{{rootName}}" mode="self"> </at>
3 </template>
4
5 <template match= "?a -> fabio:Manifestation" mode="self" >
6     <h2>
7         <font>[Format]</font>

```

```

8     <at select="??b core:embodiment {{a['uri']}}"
9         mode="title"> </at>: {{a['basic:startingPage']}}
10    <span ng-if="a['basic:endingPage']">
11        -{{a['basic:endingPage']}}
12    </span>
13 </h2>
14 <p> Format of :
15     <at select="??b core:embodiment {{a['uri']}}"
16         mode="cites"> </at>
17 </p>
18 </template>

```

codice 3.15: Template principale dei formati

Il codice 3.15 presenta il consueto template principale, con name="main", che applica un template per ricercare le triple legate al riferimento bibliografico visualizzato dalla pagina HTML. Queste triple vengono catturate dal secondo template che aggiunge un'intestazione, con il tipo di risorsa, il titolo della risorsa bibliografica a cui si riferisce il formato, la pagina di inizio e, se presente, la pagina di fine. L'ultimo paragrafo specifica nuovamente la risorsa bibliografica legata al formato.

### 3.2.6 Template per visualizzare i riferimenti bibliografici

Ora, descriviamo i template utilizzati per presentare una risorsa di tipo riferimento bibliografico

```

1 Come ultimi template, presentiamo quelli utilizzati
2   da l1 applicazione per visualizzare le risorse di tipo
3   riferimento bibliografico.
4
5 <template name="main">
6     <at select ="{{rootName}}" mode="self"> </at>
7 </template>
8
9 <template match= "?a->biro:BibliographicReference" mode="self" >
10    <h3> <font>[Bibliographic Reference]</font>

```

```

9      {{a["c4o:hasContent"]}}
10     </h3>
11     <p>Contained in: <at select="??b core:part {{a['uri']}}"
12     mode="cites">
13     </at> </p>
14     <p>References: <at select="{{a['uri']] biro:references ??b"
15     mode="cites"> </at> </p>
</template>

```

codice 3.16: Template principale dei riferimenti bibliografici

Il template principale è il template con name="main", il quale, come sempre, trova le triple legate direttamente alla risorsa principale. Il risultato dell'apply template viene catturato dal secondo template, il quale presenta il tipo di risorsa che si sta visualizzando e il contenuto del riferimento, attraverso la proprietà c4o:hasContent. Vengono poi aggiunti due template per descrivere la risorsa bibliografica che contiene quel riferimento e quella a cui è diretto.

In questa sezione abbiamo presentato alcuni template utilizzati per visualizzare le diverse entità bibliografiche definite in OpenCitations. I template presentati sono una piccola parte di quelli utilizzati in OCEB, ma, comunque, danno l'idea di come si comporti l'applicazione nella visualizzazione di una risorsa. Per facilitare la lettura del codice presentato, si è scelto di non inserire all'interno dei tag HTML il codice javascript che in realtà l'applicazione presenta. Per compensare a questa mancanza andremo nella sezione successiva a descrivere alcune delle funzioni javascript principali presenti nell'applicazione.

### 3.3 Codice javascript utilizzato nell'applicazione

A supporto dei template è stato utilizzato del codice javascript che permette di aggiungere informazioni e risolvere qualche problema. Le funzioni che

andremo a descrivere vengono invocate nelle pagine HTML dell'applicazione, all'interno dell'elemento script, che ci permette di mescolare javascript a HTML.

### 3.3.1 Funzione per inserire l'identificativo di OpenCitations

```
1 function addOcc() {  
2     var foo = getParameterByName('var');  
3     var res = foo.slice(27);  
4     var element = "<li>OCC ID: <a href='"+foo+"'> " + res +  
5         + "</a> . </li>";  
6     $("#listID").append(element);  
7 }
```

codice 3.17: Funzione per inserire l'identificativo di OpenCitations

La funzione mostrata nel codice 3.17 è utilizzata per inserire nella lista degli identificativi di ogni entità bibliografica l'identificativo di OpenCitations della stessa. Come abbiamo visto ogni risorsa ha un proprio identificativo su OpenCitations e questa funzione, appunto, lo inserisce nella pagina HTML generata dall'applicazione. Ad esempio la risorsa <https://w3id.org/oc/corpus/br/1> ha come identificativo di OpenCitations la sigla br/1. Vediamo come si comporta la funzione sopraelencata. GetParameterByName ci permette di ottenere il valore del parametro var contenuto nell'indirizzo della pagina, che specifica la risorsa centrale che stiamo visualizzando. Il metodo foo.slice(27) ci permette di tagliare la stringa che abbiamo ricevuto e di considerare solo gli ultimi caratteri, quindi se la variabile foo contiene <https://w3id.org/oc/corpus/br/1>, la variabile res conterrà solamente br/1. La variabile element crea un elemento lista di HTML contenente, a sua volta, un link all'indirizzo corrispondente al parametro dell'URL avente come valore testuale solamente la variabile res, che corrisponde all'identificativo. Nell'ultimo passaggio viene aggiunto alla lista degli identificativi l'elemento creato in precedenza.

### 3.3.2 Funzione per ordinare in ordine alfabetico le risorse citate

```
1 function order3(idList , h3) {
2     var mylist = $(idList);
3     var listitems = $(idList + " li");
4
5     listitems.sort(function(a, b) {
6         return $(a).text().toUpperCase().localeCompare
7             ($(b).text().toUpperCase());
8     })
9
10    $.each(listitems , function(idx, itm) { mylist.append(itm);
11    });
12 }
```

codice 3.18: Funzione per ordinare le risorse

La funzione descritta dal codice 3.18 svolge il compito di ordinare alfabeticamente una lista di elementi HTML. Viene utilizzata dall'applicazione per presentare in ordine alfabetico la lista delle risorse citate. Utilizza la variabile `mylist` per memorizzare l'elemento lista, mentre `listitems` per memorizzare i componenti della lista. Viene poi definita una funzione per ordinare gli elementi della lista, che viene applicata a ogni elemento di cui è composta.

### 3.3.3 Funzioni per rimuovere caratteri speciali

Questa funzione che stiamo per presentare è utilizzata per rimuovere il carattere `"$amp;"` e sostituirlo con `"$"`. E' utile nell'applicazione perché i caratteri `"$"` presenti nel triplestore vengono sostituiti da `"$amp;"` dopo aver fatto le query SPARQL e ottenuti i dati XML.

```
1 function clean3(id) {
2     $(id).each(function () {
3         var mod = $(this).text();
4         mod = mod.replace(/&/g, '&');
5         $(this).text(mod);
6     });
7 }
```

7 }

### codice 3.19: Funzione per sostituire dei caratteri

Il codice della funzione è molto semplice, riceve come parametro un selettore CSS, poi ottiene il testo degli elementi che si ottengono da quel selettore e avviene la sostituzione del carattere. Questa funzione non è l'unica che si occupa di sostituire caratteri speciali, ma è un buon esempio per tutta questa tipologia.

In questo capitolo abbiamo elencato e descritto i principali template RSLT utilizzati da OCEB. Inoltre sono state presentate alcune funzioni javascript che supportano e in qualche modo estendono le funzionalità dei template.



# Capitolo 4

## Valutazione

L'utilizzo dei template RSLT permette di visualizzare dati, in formato RDF, presenti nel triplestore di OpenCitations. Il progetto OpenCitations raccoglie dati riferiti a entità bibliografiche provenienti dalla letteratura accademica. Attraverso un parametro, chiamato "var", presente nell'URL della pagina dell'applicazione, possiamo specificare quale risorsa vogliamo visualizzare. In questo capitolo andremo a valutare il tempo di caricamento di risorse diverse, appartenenti a OpenCitations, nelle pagine HTML dell'applicazione OCEB. I tempi di caricamento della pagina web dipendono fortemente da quale risorsa si vuole visualizzare. Infatti, alcune risorse possono contenere molte più informazioni rispetto ad altre. Ad esempio un articolo può non avere citazioni oppure averne molte e questo condiziona molto il tempo di caricamento della pagina. Infatti se sono presenti più informazioni significa che devono essere scaricati più dati dal database di OpenCitations e questo comporta un inevitabile allungamento dei tempi.

### **4.1 Valutazione dei tempi di caricamento delle pagine HTML**

OpenCitations Entities Browser (OCEB) è composto da sei diverse pagine HTML per gestire i tipi di entità principali presenti in OpenCitations (risorse

bibliografiche, agenti, ruoli, identificativi, formati e riferimenti bibliografici). Valuteremo il tempo di caricamento di alcune risorse diverse su ciascuna pagina di OCEB. In questo modo verrà posta enfasi sulla differente tempistica di caricamento e analizzeremo da cosa essa dipenda. Considereremo come tempo di caricamento della pagine HTML il tempo impiegato dall'esecuzione dei template RSLT. Infatti, le pagine non avranno altre risorse da caricare o altri processi da eseguire, oltre ai template.

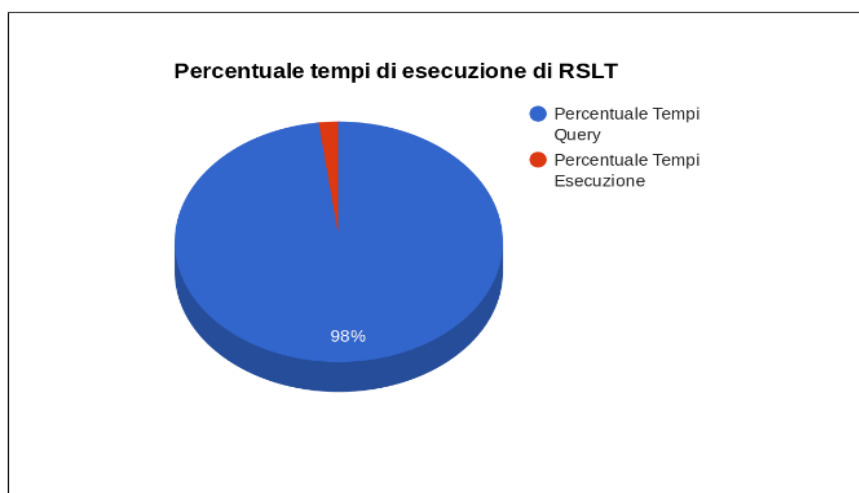


Figura 4.1: Tempi di esecuzione di RSLT

Un'ulteriore semplificazione che possiamo fare è di considerare come tempo di esecuzione dei template RSLT esclusivamente il tempo di esecuzione delle query SPARQL dai template. A dimostrazione di questo possiamo vedere l'immagine 4.1(presa da [11]), che mostra come i tempi di esecuzione di RSLT dipendano quasi esclusivamente dai tempi di esecuzione delle query. Ricapitolando, quindi, considereremo come tempo di caricamento di una risorsa in una pagina HTML il solo tempo dovuto all'esecuzione delle query dei template RSLT.

Tutti i test che andremo a presentare sono stati fatti sulla stessa macchina,

la cui architettura è composta da una CPU i3-3227U, da una memoria RAM da 4 GB e il computer è dotato del sistema operativo Ubuntu 16. Inoltre, come browser è stato utilizzato Google Chrome, versione 53. Presentiamo ora una tabella che riassume i risultati dei test che verranno poi descritti nelle sezioni successive.

Tipo	Risorsa 1	Risorsa 2	Risorsa 3
Riviste	2,70 s	5,10 s	11,25 s
Articoli	2,53 s	3,61 s	4,34 s
Agenti	5,14 s	14,83 s	40,54 s
Ruoli	2,71 s	2,70 s	2,52 s
Identificativi	2,85 s	2,75 s	2,69 s
Formati	2,65 s	2,62 s	1,81 s
Riferimenti bib.	2,95 s	3,27 s	2,73 s

La tabella mostra i tempi di caricamento di alcune risorse sulle pagine di OCEB. La prima colonna contiene il tipo di risorsa che è stato considerato, le altre tre colonne contengono i risultati effettuati su diverse risorse del tipo definito dalla prima colonna. I tempi di alcune risorse dello stesso tipo variano anche molto, questo perché i test sono stati effettuati considerando risorse con un carico di informazioni da presentare molto differente.

#### 4.1.1 Valutazione del tempo di caricamento delle risorse bibliografiche

In questa sezione faremo un confronto fra il tempo di caricamento di alcune risorse bibliografiche, che saranno presentate sulla pagina di OCEB dedicata a queste entità, chiamata "Bibl.html". Un primo confronto sarà fatto tra tre diverse risorse di tipo rivista, mentre un secondo verrà fatto tra tre altre risorse di tipo articolo.

Per il primo confronto consideriamo le risorse di tipo rivista:

<https://w3id.org/oc/corpus/br/8> (br/8), <https://w3id.org/oc/corpus/br/3> (br/3)

e <https://w3id.org/oc/corpus/br/19> (br/19). OCEB per ognuna di queste risorse mostra il titolo, gli identificativi, i volumi contenuti al suo interno e gli articoli che sono presenti direttamente al loro interno, senza essere inclusi in un volume o in un fascicolo. L'unica differenza che emerge tra le tre è la seguente: la rivista br/8 contiene al suo interno un solo articolo, la rivista br/3 ne contiene 4, mentre la rivista br/19 ne contiene ben 13. La pagina "Bibl.html", ossia quella di OCEB preposta alle risorse bibliografiche, impiega 2.70 secondi per caricare br/8, 5.10 secondi per caricare br/3 e 11,25 secondi per caricare br/19. Come si può facilmente notare, i tre tempi sono molto differenti e aumentano in modo quasi costante all'aumentare degli articoli da presentare. Possiamo considerare che la pagina HTML impieghi circa 2 secondi per caricare tutte parti di una rivista escludendo l'elenco degli articoli e che per ogni articolo caricato vengano impiegati circa 0,7 secondi. Un articolo per essere inserito nella pagina impiega molto tempo perché devono essere trovate molte informazioni a suo riguardo e, quindi, devono essere applicati molti template. Infatti, vengono applicati dei template per trovare tutti gli autori, per trovare il titolo e sottotitolo, per trovare l'editore e per trovare il formato. L'applicazione di tanti diversi template comporta così ad avere un tempo di presentazione delle informazioni di un articolo così grande.

Ora procediamo con il prossimo confronto tra tre risorse di tipo articolo. Consideriamo le risorse <https://w3id.org/oc/corpus/br/37431> (br/37431), <https://w3id.org/oc/corpus/br/680127> (br/680127) e <https://w3id.org/oc/corpus/br/1> (br/1). Le informazioni mostrate da OCEB per ognuna di queste risorse sono il titolo, il sottotitolo, gli autori, la rivista che contiene l'articolo, la lista degli identificativi e la lista dei riferimenti bibliografici. Le principali differenze tra la visualizzazione delle tre risorse riguardano il fatto che la risorsa br/37431 contiene solamente 17 riferimenti bibliografici, la risorsa br/680127 ne contiene 59 e la risorsa br/1 ne contiene 211. Il tempo di caricamento dell'articolo con 17 riferimenti bibliografici è di 2,53 secondi, dell'articolo con 59 riferimenti è di 3,61 secondi e dell'ar-

titolo con 211 riferimenti è di 4,34 secondi. Naturalmente anche in questo confronto c'è una certa differenza di tempo di caricamento tra le risorse che contengono quantità di informazioni diverse. Però in questo caso, a differenza del confronto precedente, la differenza di tempo non è così marcata. Ciò è dovuto al fatto che tutti i riferimenti necessitano di una sola query per essere presentati, ovvero viene fatta un'unica query SPARQL che trova tutti i riferimenti bibliografici di un articolo e esiste un template che farà match con tutti questi risultati e li inserirà nell'elenco dei riferimenti bibliografici. Nel confronto fatto tra le riviste, invece, per ogni articolo che si doveva inserire venivano applicati più template e di conseguenza fatte più query.

### **4.1.2 Valutazione del tempo di caricamento degli agenti**

Confrontiamo ora tre risorse diverse di tipo agente. Consideriamo questi tre autori: <https://w3id.org/oc/corpus/ra/516591> (ra/516591), <https://w3id.org/oc/corpus/ra/25709> (ra/25709), <https://w3id.org/oc/corpus/ra/51659> (ra/51659). Ora valutiamo il tempo di caricamento di queste risorse nella pagina Agent.html di OCEB. Per ognuno di questi agenti vengono mostrate delle informazioni che indicano il nome e cognome della persona, gli identificativi a lui collegati e una lista di articoli di cui lui è autore. Il numero di articoli presenti nell'elenco condizionano molto il tempo di caricamento di una risorsa di tipo agente, così come lo era per le riviste. L'agente ra/516591 ha scritto 4 articoli, l'agente ra/25709 ha scritto 10 articoli, mentre l'agente ra/51659 ha scritto 36 articoli. Il tempo di caricamento di queste risorse è rispettivamente di 5,14 secondi, 14,83 secondi e 40,54 secondi.

### **4.1.3 Valutazione del tempo di caricamento dei ruoli**

Ora confronteremo tre risorse di tipo ruolo, che sono:

<https://w3id.org/oc/corpus/ar/1> (ar/1), <https://w3id.org/oc/corpus/ar/2> (ar/2)

e <https://w3id.org/oc/corpus/ar/20> (ar/20). I tempi di caricamento di queste quattro risorse nella pagina Role.html di OCEB sono 2,71 secondi per ar/1, 2,70 secondi per ar/2 e 2,52 secondi per ar/20. Come si può osservare il tempo di caricamento di queste risorse è molto simile, le differenze che ci sono dipendono dalla presentazione dell'articolo a cui è legato il ruolo. Infatti, ar/1 e ar/2 si riferiscono allo stesso articolo e il tempo di caricamento delle due risorse è molto simile, mentre ar/20 si riferisce a un'altra risorsa bibliografica.

#### **4.1.4 Valutazione del tempo di caricamento degli identificativi**

In questa sezione, consideriamo il tempo di caricamento di tre identificativi diversi, mediante la pagina HTML di OCEB preposta alla presentazione di identificativi (Ident.html). Le risorse su cui faremo un confronto sono tre identificativi della risorsa br/1, che abbiamo visto precedentemente. Essi sono: <https://w3id.org/oc/corpus/id/1> (id/1), <https://w3id.org/oc/corpus/id/3> (id/3), <https://w3id.org/oc/corpus/id/6> (id/6). Le visualizzazioni di queste risorse sono molto simili tra loro e le differenze principali tra queste riguardano il tipo di identificativo che rappresentano. Id/1 è un identificativo di tipo DOI, id/3 di tipo URL e id/6 di tipo pmid. Il tempo di caricamento di queste risorse è rispettivamente di 2,85 secondi, 2,75 secondi e 2,69 secondi. Questo sta a significare che il tempo di caricamento di risorse diverse di tipo identificativo è abbastanza simile.

#### **4.1.5 Valutazione del tempo di caricamento dei formati**

Considereremo il tempo di caricamento di tre risorse di tipo formato sulla pagina Format.html, indicata alla presentazione di risorse di tipo formato. Confronteremo le risorse <https://w3id.org/oc/corpus/re/1> (re/1), <https://w3id.org/oc/corpus/re/2> (re/2) e <https://w3id.org/oc/corpus/re/3>

(re/3) Il tempo di caricamento di queste risorse è rispettivamente di 2,65 secondi, 2,62 secondi e 1,81 secondi. Anche in questo caso le piccole differenze di tempo dipendono dalla presentazione della risorsa bibliografica legata al formato che si sta visualizzando.

#### **4.1.6 Valutazione del tempo di caricamento dei riferimenti bibliografici**

Infine, in questa sezione andremo a confrontare il tempo di caricamento di tre risorse di tipo riferimento bibliografico in una pagina HTML di OCEB. Quelle che andremo a confrontare sono tre riferimenti presenti all'interno della risorsa br/1 e sono: <https://w3id.org/oc/corpus/be/1> (be/1), <https://w3id.org/oc/corpus/be/2> (be/2), <https://w3id.org/oc/corpus/be/3> (be/3). I tempi di caricamento nella pagina di OCEB preposta alla visualizzazione di riferimenti (Refer.html) sono di 2,95 secondi per be/1, 3,27 per be/2 e 2,73 per be/3. In questo caso, le differenze sono minime e sono date dal numero di informazioni che devono essere presentate della risorsa bibliografica a cui si riferisce il riferimento. Nella visualizzazione di ognuno di questi tre riferimenti sono mostrate delle informazioni sulla risorsa che contiene quel riferimento (avendo preso dei riferimenti bibliografici tutti presenti in br/1 queste non cambiano), e delle informazioni sulla risorsa a cui si riferisce il riferimento (queste cambiano da riferimento a riferimento).

In questo capitolo abbiamo valutato il tempo impiegato dall'applicazione OCEB a visualizzare alcune risorse presenti nel dataset di OpenCitations. E' emerso che le entità diverse, di tipo risorsa bibliografica e agente, abbiano dei tempi di caricamento molto diversi tra loro, dipendenti dal carico di informazioni che contengono. Le altre entità, invece, hanno un tempo di caricamento molto costante.





# Capitolo 5

## Conclusioni

La raccolta di dati in un qualsiasi database ha tra i tanti scopi quello di rendere facilmente visualizzabili e presentabili quei dati all'utente. OCEB permette di visualizzare i metadati bibliografici e citazionali presenti nel triplestore di OpenCitations. Per fare ciò l'applicazione utilizza delle pagine HTML con all'interno dei template RSLT, che permettono di eseguire delle query SPARQL e presentare i risultati ottenuti nella pagina. Di seguito verranno introdotti alcuni possibili lavori futuri che porteranno a migliorare OCEB a livello di prestazioni, funzionalità e versatilità.

Nel capitolo 4 abbiamo valutato le prestazioni di OCEB ed è emerso che il tempo impiegato da una pagina dell'applicazione a caricare le informazioni legate a una risorsa dipenda fortemente dal numero di dati che devono essere presentati per quella risorsa. Comunque i tempi di esecuzione delle query SPARQL potrebbero sicuramente essere diminuiti facendo delle query più efficienti. Soprattutto sarebbe positivo uniformare il tempo di caricamento tra le diverse risorse, evitando che una risorsa che abbia molti dati da dover caricare impieghi molto più tempo di un'altra che ne abbia meno. Sicuramente il divario non potrà essere eliminato del tutto, però potrebbe essere limitato. Anche per l'esperienza utente dover aspettare dei tempi molto diversi per il caricamento di due risorse simili può risultare sgradevole.

Sarebbe comodo utilizzare un'unica pagina HTML per presentare tutte le diverse entità di OpenCitations. Come abbiamo detto nei capitoli precedenti, OCEB è composto da sei pagine HTML, ognuna preposta a presentare un tipo di entità differente di OpenCitations. Questa scelta è stata fatta per semplificare la presentazione delle diverse risorse e per non inserire troppi template in un'unica pagina. La riduzione a un'unica pagina permetterebbe di rendere più elegante questo sistema di visualizzazione ed eviterebbe di riportare gli stessi template più volte nelle sei diverse pagine. Infatti, molti template sono stati copiati da una pagina all'altra per poterli utilizzare su più pagine, ma questo ha comportato un'inevitabile ridondanza di codice tra tutte le pagine HTML. Un ulteriore vantaggio della riduzione a una pagina sarebbe quello di poter visualizzare nella stessa pagina tutte le risorse di OpenCitations, senza dover cercare la pagina preposta al tipo di risorsa che si vuole visualizzare. Ora, ad esempio, per visualizzare una risorsa di tipo agente bisogna utilizzare la pagina di OCEB indicata per quel tipo di risorsa. Infatti, su un'altra pagina la visualizzazione non sarebbe adeguata, perché ogni pagina contiene dei template specifici per un tipo di risorsa.

Un'estensione possibile sarebbe quella di inserire dei grafi e dei diagrammi nelle pagine HTML di OCEB. In questo modo si avrebbe una visione più immediata di alcune informazioni riguardanti una risorsa. Ad esempio, si potrebbe inserire nelle pagine di OCEB, predisposte alla visualizzazione di articoli, un grafo che corrisponderebbe alla rete di citazione di un articolo. Il nodo centrale rappresenterebbe l'articolo principale, mentre gli altri nodi rappresenterebbero altri articoli che sono citati o che hanno citato il nodo centrale. Naturalmente gli archi, meglio se orientati, indicherebbero una relazione tra un articolo e un altro. Oppure si potrebbero inserire dei diagrammi per visualizzare il numero di citazioni nell'arco del tempo di un articolo o altre statistiche interessanti.

Sicuramente una possibile modifica in futuro può essere data dall'aumento delle informazioni presentate in una visualizzazione di una risorsa per mezzo di OCEB. Quando si visualizza una risorsa in una pagina vengono visualizzate un certo numero di informazioni che la riguardano, direttamente o indirettamente. Un aumento delle informazioni presentate sarebbe sicuramente un vantaggio per l'utente che, avendo a disposizione più dati, potrebbe aumentare le sue conoscenze a riguardo di quella risorsa. Consideriamo, ad esempio, una risorsa di tipo volume che contiene al suo interno dei fascicoli che, a loro volta, contengono degli articoli. La visualizzazione di questa risorsa presenterà una lista di tutti i fascicoli che essa contiene, mentre non verranno mostrati gli articoli contenuti nei fascicoli. In questo modo un utente per poter accedere a un articolo dovrà prima fare click su uno dei fascicoli elencati nella pagina del volume, in modo da aprire la pagina di un fascicolo. Poi, in quella visualizzazione potrà accedere alle informazioni riguardanti gli articoli che sono contenuti in quel fascicolo. In questo modo, però, risulta scomodo per l'utente accedere agli articoli indirettamente passando sempre per una pagina di un fascicolo. Una possibile soluzione è quella di elencare nella pagina di una risorsa di tipo volume direttamente tutti gli articoli che sono contenuti nei fascicoli di quel volume. Una scelta di questo tipo ha un rovescio della medaglia, ovvero il tempo di caricamento di una pagina che dovrà presentare più informazioni inevitabilmente aumenterà.

Un'ulteriore estensione possibile è quella di offrire agli utenti una funzione che permetta di segnalare un'informazione presente in una visualizzazione di una risorsa che ritiene scorretta o imprecisa. Magari attraverso un modale l'utente potrebbe segnalare quale è il dato che trova scorretto e il perché. Dopo la segnalazione si cercherà di capire se l'errore dipenda da OCEB o è un errore presente nei dati raccolti in OpenCitations e si tenterà, ad ogni modo, di correggerlo.

OCEB attualmente è stato valutato quantitativamente, ma non qualitativa-

mente. Una valutazione qualitativa, realizzata intervistando diverse persone dopo che hanno utilizzato l'applicazione per qualche scopo, potrebbe essere molto utile. Ad esempio, si potrebbero assegnare come compiti, alle persone intervistate, quelli di trovare tutti gli articoli di alcuni autori oppure di visualizzare tutti gli articoli contenuti in una rivista. In questo modo potrebbero essere indicati dei limiti e dei difetti che gli utenti hanno riscontrato e l'applicazione potrebbe essere migliorata seguendo queste indicazioni.

# Bibliografia

- [1] Berners-Lee, T., Hendler, J., Lassila O. (2001).The Semantic Web, Scientific American.
- [2] <https://www.w3.org/>.
- [3] Guha, R., Brickley, D. (2014). RDF Schema 1.1. W3C Recommendation. <http://www.w3.org/TR/2014/REC-rdf-schema-20140225/>. W3C.
- [4] The W3C SPARQL Working Group. SPARQL 1.1 Overview. W3C Recommendation. <https://www.w3.org/TR/2013/REC-sparql11-overview-2013032/>.W3C, mar. 2013.
- [5] <https://www.w3.org/TR/owl-features/>.
- [6] Florenzano F., Parra D., Reutter J., Venegas1 F. (2016). An interactive visualisation for RDF data, Proceedings of the ISWC 2016 Posters Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016).
- [7] Athanassiades, A., Kontopoulos, E., Bassiliades N. (2009). Visualizing RDF Documents.
- [8] Angles, R. (2009). A Nested Graph Model for Visualizing RDF Data, Proceedings of the 3rd Alberto Mendelzon International Workshop on Foundations of Data Management. Arequipa, Peru.

- [9] Liu, S., Cede, J., Candan, K., Sapino, M., Huang, S., Li, X. (2009). R2DB: A System for Querying and Visualizing Weighted RDF Graphs, 2012 IEEE 28th International Conference on Data Engineering.
- [10] Peroni, S., Vitali, F. (2015). RSLT: R. D. F. Stylesheet Language Transformations . In: Proceedings of the ESWC Developers Workshop 2015 co-located with the 12th Extended Semantic Web Conference (ESWC 2015), Portorož, Slovenia. A cura di Ruben Verborgh e Miel Vander Sande. Vol. 1361. CEUR Workshop Proceedings. CEUR-WS.org, 2015, pp. 7–13. url: <http://ceur-ws.org/Vol-1361>.
- [11] Quadrelli, D. (2015) RSLT: trasformazione di Open Linked Data in testi in linguaggio naturale tramite template dichiarativi Tesi di laurea, Scuola di Scienze, Corso di Laurea in Informatica, Alma Mater Studiorum Università di Bologna, 2015/2016. II.
- [12] Bagnacani, A., Ciancarini, P., Di Iorio, A., Nuzzolese, A. G., Peroni, S., Vitali, F. (2015). The Semantic Lancet Project: a Linked Open Dataset for Scholarly Publishing In Proceedings of Satellite Events of EKAW 2014, Lecture Notes in Artificial Intelligence 8982: 101-105 Berlin, Germany: Springer.
- [13] Bagnacani, A., Ciancarini, P., Di Iorio, A., Nuzzolese, A. G., Peroni, S., Vitali, F. (2015). Knowledge management of scholarly products: Semantic Lancet Project Proceedings of 16th European Conference on Knowledge Management (ECKM 2015). Reading, UK: Academic Conferences and Publishing International.
- [14] Di Iorio, A., Giannella, R., Poggi, F., Vitali, F. (2015). Exploring bibliographies for research-related tasks.
- [15] Di Iorio, A., Giannella, R., Poggi, F., Peroni, S., Vitali, F. (2015). Exploring Scholarly Papers Through Citations In Proceedings of the 2015 ACM Symposium on Document Engineering (DocEng '15). ACM, New York, NY, USA, 107-116.

- [16] Giannella, R. (2015) BEX: un ambiente user-friendly per esplorare articoli scientifici e bibliografie Tesi di laurea, Scuola di Scienze, Corso di Laurea in Informatica per il Management, Alma Mater Studiorum Università di Bologna, 2013/2014. III.
- [17] Graziosi, A. (2015) Navigazione di articoli scientifici e citazioni: un ambiente scalabile e flessibile Tesi di laurea, Scuola di Scienze, Corso di Laurea in Informatica per il Management, Alma Mater Studiorum Università di Bologna, 2015/2016. I.
- [18] Peroni, S., Dutton, A., Gray, T., Shotton, D. (2015). Setting our bibliographic references free: towards open citation data. *Journal of Documentation*, 71 (2): 253-277. <http://dx.doi.org/10.1108/JD-12-2013-0166>, OA at <http://speroni.web.cs.unibo.it/publications/peroni-2015-setting-bibliographic-references.pdf>.
- [19] Shotton, D. (2013). Open citations. *Nature*, 502 (7471): 295-297. <http://dx.doi.org/10.1038/502295a>.





# Ringraziamenti

Ringrazio i miei genitori per tutto il supporto e l'incoraggiamento che mi hanno dato in questi anni.

Inoltre, ringrazio i miei amici Alessio, Andrea B., Andrea C., Andrea N., Davide e Simone per essere sempre stati presenti.