

Creare stili bibliografici con **biblatex**: l'esperienza del pacchetto **biblatex-philosophy**

Ivan Valbusa*

Sommario

Lo scopo di questo articolo è descrivere la genesi e le caratteristiche principali degli stili bibliografici e di citazione messi a disposizione dal pacchetto **biblatex-philosophy** e di fornire le indicazioni di base per creare uno stile per il pacchetto **biblatex** di Philipp Lehman. L'articolo richiede conoscenze di base sul funzionamento di **BIBTEX** e di **biblatex**.

Abstract

The aim of this article is to describe the genesis and the main features of the bibliography and citation styles provided by **biblatex-philosophy** package; moreover, it provides the basic concepts to create a style for use with Philipp Lehman's **biblatex** package. This article requires basic knowledge of **BIBTEX** and **biblatex**.

1 Introduzione

Uno dei tanti punti di forza di **L^AT_EX** è la possibilità di gestire la bibliografia in modo estremamente efficiente. **BIBTEX**, assieme ai molti stili e pacchetti attualmente disponibili, risponde alla maggior parte delle esigenze degli utenti **L^AT_EX**. Tutti riconoscono però che i sistemi attuali hanno una natura “rigida”, o meglio, richiedono grandi competenze per essere adattati a richieste particolari.¹

Il pacchetto **biblatex** di Philipp Lehman (LEHMAN, 2010) risolve questo problema, offrendo uno strumento potente, versatile ed estremamente più semplice, per gestire le bibliografie con **L^AT_EX**. Anche se prevede comunque l'uso di **BIBTEX** per la gestione dell'ordinamento delle voci e per generare le etichette, si tratta di una totale reimplementazione del sistema di gestione della bibliografia. I codici degli stili di **biblatex** sono infatti costituiti da macro scritte nel più familiare linguaggio **L^AT_EX**.

Sebbene sia ancora in fase di sviluppo, **biblatex** è perfettamente funzionante, e viene fornito assieme a numerosi stili: numerico, autore-anno,

autore-titolo, con parecchie varianti. Il suo principale punto di forza è la possibilità di personalizzare gli stili bibliografici e gli schemi di citazione forniti dal pacchetto attraverso macro **L^AT_EX** inserite direttamente nel preambolo del proprio documento, ma per esigenze particolari consente anche di creare nuovi stili.²

Il pacchetto **biblatex-philosophy** (VALBUSA, 2010) mette a disposizione un insieme di stili bibliografici con relativo schema di citazione composto da due varianti dello schema di citazione autore-anno (stili **philosophy-classic** e **philosophy-modern**) e una variante per le citazioni estese in nota (stile **philosophy-verbose**).³

Nel seguito di questo articolo mi propongo di spiegare come modificare alcune caratteristiche degli stili standard di **biblatex** e come costruire uno stile bibliografico completo, avendo come punto di riferimento gli stili **philosophy-classic** e **philosophy-modern**. Tenuto conto che **biblatex** è ancora in fase sviluppo e che le macro del pacchetto potranno subire cambiamenti anche profondi nelle prossime versioni, preciso che gli esempi di codice che verranno qui esposti sono compatibili con la versione 0.9a di **biblatex**.

Nel § 2 verrà dato un breve resoconto del cammino che ha portato alla nascita di questi stili, indicandone le caratteristiche principali (§ 2.1) e le motivazioni che hanno determinato la scelta dello schema di citazione autore-anno (§ 2.2). Nel § 3 si darà una descrizione del funzionamento di base degli stili di **biblatex**, relativamente alla gestione della punteggiatura (§ 3.1) e del formato dei campi (§ 3.2). Nel § 4 verranno infine fornite alcune indicazioni su come creare uno stile completo per **biblatex**, con alcune delle caratteristiche degli stili **philosophy**.

2 Breve storia degli stili **philosophy**

La storia degli stili **philosophy** ha inizio sul forum del **G_UIT**. In un primo momento si è trattato di un problema molto specifico: ricreare una bibliografia che assomigliasse a quella illustrata da Umberto Eco nel suo *Come si fa una tesi di laurea* (ECO, 2005, 193).

2. Per un'introduzione a **biblatex** vedi PANTIERI (2009).

3. Il pacchetto **biblatex-philosophy** è disponibile su CTAN all'indirizzo: <http://www.ctan.org/tex-archive/macros/latex/exptl/biblatex-contrib/biblatex-philosophy>.

*Esprimo un particolare ringraziamento a Gustavo Cevolani che mi ha spinto ad elaborare questo articolo, dandomi sempre preziosi consigli. Ringrazio inoltre tutti coloro che hanno partecipato alle molte discussioni sul forum del **G_UIT** e che mi hanno permesso, con le loro osservazioni, di dare agli stili **philosophy** una forma compiuta. Infine desidero ringraziare Enrico Gregorio per la sua grande disponibilità e per l'aiuto che mi ha fornito in più occasioni.

1. Per approfondire vedi HUFFLEN (2007); MORI (2008).

ARISTOTLE

- 1877 *The Rhetoric of Aristotle with a commentary by the late Edward Meredith Cope*, cur. e comm. da Edward Meredith Cope, 3 voll., Cambridge University Press.
- 1907 *De Anima*, a cura di Robert Drew Hicks, Cambridge University Press, Cambridge.
- 1929 *Physics*, trad. da P. H. Wicksteed e F. M. Cornford, G. P. Putnam, New York.
- 1968 *Poetics*, a cura di D. W. Lucas, Clarendon Aristotle, Clarendon Press, Oxford.

KANT, IMMANUEL

- 1968a *Kritik der praktischen Vernunft*, in: *Kants Werke. Akademie Textausgabe*, vol. 5: *Kritik der praktischen Vernunft. Kritik der Urtheilskraft*, Walter de Gruyter, Berlin, pp. 1–163.
- 1968b *Kritik der Urtheilskraft*, in: *Kants Werke. Akademie Textausgabe*, vol. 5: *Kritik der praktischen Vernunft. Kritik der Urtheilskraft*, Walter de Gruyter, Berlin, pp. 165–485.

KNUTH, DONALD E.

- 1984–1986 *Computers & Typesetting*, 5 voll., Addison-Wesley, Reading, Mass.
- 1984 *Computers & Typesetting*, vol. A: *The TeXbook*, Addison-Wesley, Reading, Mass.
- 1986a *Computers & Typesetting*, vol. B: *TeX: The Program*, Addison-Wesley, Reading, Mass.
- 1986b *Computers & Typesetting*, vol. C: *The METAFONTbook*, Addison-Wesley, Reading, Mass.
- 1986c *Computers & Typesetting*, vol. D: *METAFONT: The Program*, Addison-Wesley, Reading, Mass.
- 1986d *Computers & Typesetting*, vol. E: *Computer Modern Typefaces*, Addison-Wesley, Reading, Mass.

NIETZSCHE, FRIEDRICH

- 1988a *Sämtliche Werke. Kritische Studienausgabe*, a cura di Giorgio Colli e Mazzino Montinari, 2ª ed., 15 voll., Deutscher Taschenbuch-Verlag e Walter de Gruyter, München, Berlin e New York.
- 1988b *Sämtliche Werke. Kritische Studienausgabe*, vol. 1: *Die Geburt der Tragödie. Unzeitgemäße Betrachtungen I–IV. Nachgelassene Schriften 1870–1973*, a cura di Giorgio Colli e Mazzino Montinari, 2ª ed., Deutscher Taschenbuch-Verlag e Walter de Gruyter, München, Berlin e New York.

FIGURA 1: Esempio di bibliografia composta con lo stile **philosophy-modern**, sul modello di Eco (2005, 193), rispetto al quale è stato migliorato l'allineamento delle date.

La bibliografia in questione, diffusa presso molte case editrici italiane (per esempio Bompiani, Bollati Boringhieri ed Einaudi), appare come nello scorcio di figura 1. Come si vede, le opere relative ad ogni autore sono raggruppate in un blocco e questo permette di capire a colpo d'occhio qual è il “peso” dell'autore all'interno del documento. È evidente che questo stile è particolarmente efficace con bibliografie corpose, e magari con pochi autori e molte voci bibliografiche.

Mentre si risolveva questo problema sono sorte però molte altre questioni riguardanti non solamente gli stili bibliografici, e più in generale **biblatex**, ma anche i problemi legati alle convenzioni bibliografiche, agli usi e costumi dei diversi settori disciplinari, ai gusti personali, ecc. Insomma la questione è diventata, come direbbe un filosofo, *eminentemente* stilistica.

Per creare lo stile **philosophy-modern**, che come vedremo risponde alle esigenze emerse durante la lunga discussione sul forum, è stato necessario creare dapprima uno stile senza la complicazione dei “blocchi” ed è così nato lo stile **philosophy-classic**. A questi due stili basati sullo schema autore-anno, si è aggiunto recentemente un altro stile, **philosophy-verbose**, pensato per le citazioni estese fatte nelle note a piè di pagina e diffuse in particolare nell'ambito umanistico.

2.1 Natura degli stili **philosophy**

Gli stili **philosophy-classic** e **philosophy-modern** sono stati concepiti inizialmente per rispondere in particolare alle esigenze degli utenti umanisti, che devono spesso comporre voci bibliografiche molto complesse; ma sono ugualmente (forse più) adatti per l'ambito tecnico-scientifico.

Le caratteristiche principali degli stili **philosophy** sono la possibilità di comporre voci contenenti le informazioni sulla traduzione di un testo straniero o sulla ristampa (vedi § 4.1.3) e voci accompagnate da un'annotazione (vedi § 4.1.1), utili soprattutto in testi con finalità didattiche. Consentono di comporre voci anche molto complesse, utilizzando appositi campi per quasi tutti gli elementi della voce. Eccone un esempio:

Poincaré, Jules-Henri (1968), *La science et l'hypothèse*, Flammarion, Paris, prima edizione 1902; trad. it. *La scienza e l'ipotesi*, testo greco a fronte, a cura di Corrado Sinigaglia, Bompiani, Milano 2003.

Un classico della filosofia della scienza, nel quale vengono trattati i temi fondamentali dell'epistemologia del Novecento, dalla natura dello spazio fisico fino al significato della geometria.

Forniscono infine un certo numero di opzioni per modificare in maniera molto semplice alcune caratteristiche della bibliografia e delle citazioni, come il formato dei nomi degli autori, che possono essere stampati in maiuscolo, e lo stile in cui appaiono l'editore, il luogo e l'anno di edizione. Nell'esempio seguente i campi relativi alla traduzione sono stati racchiusi, grazie a un'apposita opzione, tra parentesi tonde (ma ne esiste una anche per ottenere le parentesi quadre):

MACH, ERNST (1883), *Die Mechanik in ihrer Entwicklung historisch-kritisch dargestellt* (trad. it. *La meccanica nel suo sviluppo storico-critico*, traduzione, introduzione e note di Alfonsina D'Elia, Torino: Bollati Boringhieri, 1977).

Si è discusso molto sul nome da assegnare a questi stili. Inizialmente, scosso dall'entusiasmo di Lorenzo Pantieri, che annunciava con autentica passione *lo stile bibliografico migliore del mondo*,⁴ avevo pensato di chiamare gli stili con l'acronimo **msb**, che sta per “miglior stile bibliografico”. Dopo una lunga discussione la scelta è caduta su **philosophy**. Questo per tre ragioni: essendo cresciuto nell'ambito della filosofia (sebbene di quella cosiddetta *classica tedesca*) ho voluto lasciare un segno di riconoscenza alla disciplina che mi ha permesso di vivere fino ad ora; in secondo luogo, essendo un sostenitore di quella che un tempo si sarebbe chiamata “filosofia scientifica”, ho voluto fare in modo che il nome *philosophy* venisse associato in primo luogo allo schema di citazione autore-anno;⁵ infine, e questa è forse la motivazione principale, ho un certo piacere, quasi un senso di riscatto, nel sapere che uno studente di matematica, di fisica o

4. <http://www.guit.sssup.it/phpbb/viewtopic.php?t=6472>

5. La recente nascita dello stile **philosophy-verbose** ha evidentemente ridimensionato questo intento.

di qualsiasi altra scienza più o meno “esatta” possa avere nel preambolo del suo documento L^AT_EX il termine *philosophy*. D'altra parte non è un affronto così grande per chi ha dimestichezza con la storia della scienza moderna.

2.2 La scelta dello schema autore-anno

Una parte della discussione, animata anche dalla nascita di questi stili, ha riguardato lo schema di citazione autore-anno e la sua presunta inadeguatezza per alcune tipologie di scritto. Gli stili *philosophy-classic* e *philosophy-modern* sono, come già detto, basati su questo schema.

Sebbene vi siano molte ragioni (o motivazioni, giusto per non scomodare il *lógos*) a sostegno dello schema di citazione in forma estesa in campo umanistico, sono dell'avviso che il più versatile (più versatile, non *il* migliore!) sia proprio lo schema autore-anno, per diversi aspetti importanti.

Innanzitutto, esso consente di vedere immediatamente a *chi* si riferisce la citazione. Questo lo rende migliore dello schema numerico, che è molto diffuso in ambito scientifico, e improponibile agli umanisti. D'altra parte, nel contesto scientifico non è importante tanto il conoscere subito *chi* viene citato, ma innanzitutto *che* viene citato qualcuno, e successivamente *cosa* dice l'autore citato. Il ricorso all'autorità è deprecato nelle scienze; e abusato nella filosofia.

Lo schema autore-anno permette inoltre di riportare le citazioni direttamente nel testo e aiuta ad abbandonare la cattiva consuetudine di riempire i testi di lunghe note a piè di pagina, che non senza ragione R. Bringhurst definisce come «l'emblema della meticolosità» (BRINGHURST, 2009, § 4.3.1). Questo comporta di certo un adattamento del proprio stile, ma per uno scopo nobile: la chiarezza espositiva e la cura estetico-formale del testo, che non è mai separata dal contenuto comunicativo.

Questo schema di citazione, inoltre, ci evita di “duplicare” la bibliografia, fatta per esteso in nota e riportata anche alla fine del documento. È evidente, infatti, che la bibliografia finale è teoricamente superflua se si usano le citazioni estese in nota. D'altra parte si rende praticamente inevitabile soprattutto in testi molto ampi.

Infine, è uno schema di citazione diffuso, anzi, direi molto diffuso in campo filosofico, in particolare nella tradizione “analitica” e diffusissimo nel campo delle scienze, dalla chimica alla psicologia fino alla matematica.

Bisogna però ammettere che c'è almeno un limite proprio di questo schema di citazione: non consente di vedere immediatamente da *dove* (quale opera) è stata ripresa la citazione. In alcune tipologie di testi ciò potrebbe portare a qualche disagio. In casi come questi è possibile tuttavia usare delle sigle, come avviene comunemente, per esempio, con le opere di Kant, Aristotele, Platone e molti altri classici del pensiero. In questo modo si

concilia l'essenzialità dello schema autore-anno con il bisogno di contestualizzazione sentito da molti.

Appare evidente che questo tema è strettamente legato a quello delle convenzioni bibliografiche. Come ha giustamente affermato Gustavo Cevolani, «la bibliografia è forse il più tormentato fra gli aspetti tipografici della composizione di un documento. Se in generale esistono poche regole assolute per la tipografia, nel caso della bibliografia non ne esiste sostanzialmente alcuna» (CEVOLANI, 2006, 39). La mancanza di norme precise e di un manuale di stile realmente consolidato ha fatto sì che in Italia non vi sia stata e non vi sia tutt'ora chiarezza sulla gestione della bibliografia (vedi anche LESINA, 2009). I dubbi che continuamente emergono in merito ne sono un'attestazione. Ogni contesto di ricerca solleva le proprie motivazioni a favore di un particolare stile, ma sottoposte a giudizio analitico queste motivazioni appaiono fragili. Quelle che in un primo momento sembrano ragioni si risolvono in semplici abitudini. Alle abitudini ci si affeziona, è vero. Ma fortunatamente possono anche essere cambiate.

Gli stili *philosophy* sono diventati così anche un'occasione di riflessione sulla possibilità di creare uno standard bibliografico il più possibile razionale che fosse in grado di imporsi alla tipografia italiana, in particolare in ambito accademico; in quest'ottica L^AT_EX sarebbe un ottimo punto di partenza e *biblatex* il suo strumento di elezione.

Prospettare uno standard bibliografico è abbastanza realistico nel contesto delle discipline scientifiche, dove uno standard, in un certo senso, esiste già, o quasi; più arduo nell'ambito umanistico, dove non solo l'attaccamento verso gli stili di citazione estesa è molto più radicato, ma dove esistono anche necessità oggettive più complesse, come la citazione di classici del pensiero, edizioni critiche, traduzioni, ristampe, ecc. Il progetto, tutt'altro che tramontato (almeno nella mente di chi scrive), aspetta quindi che il bacino di utenti umanisti di L^AT_EX si allarghi a sufficienza e che il dibattito si diffonda.

Ora che abbiamo brevemente raccontato la genesi e le motivazioni alla base di questi stili, cerchiamo di descriverne alcune caratteristiche. Per fare questo ci serviranno poche semplici nozioni preliminari su come possono essere gestiti gli stili bibliografici con *biblatex*.

3 Gli stili di *biblatex*

Uno stile bibliografico completo per *biblatex* comprende tre *file*: un file con estensione *.bbx* per la gestione della bibliografia vera e propria (detto anche, in senso stretto, “stile bibliografico”); un file con estensione *.cbx* che gestisce le citazioni (detto “schema di citazione”); un file con estensione *.lbx* per il supporto linguistico (che chiameremo “file di localizzazione”). Questi file hanno

bisogno in particolare di altri due importanti file, `biblatex.sty` e `biblatex.def`, che contengono le definizioni di molte delle macro di base necessarie al funzionamento degli stili.⁶

Il pacchetto `biblatex` mette a disposizione diversi stili bibliografici che possono servire come punto di partenza per la creazione di stili personalizzati.

Ogni tipo di voce bibliografica è definita nel file `standard.bbx` dal comando:

```
\DeclareBibliographyDriver{<type>}{<code>}
```

A titolo di esempio riporto di seguito la macro, che definisce il tipo `@article` (nel seguito “driver `@article`”):

```
\DeclareBibliographyDriver{article}{%
  \usebibmacro{bibindex}%
  \usebibmacro{begentry}%
  \usebibmacro{author/translator+others}%
  \setunit{\labelnamepunct}\newblock
  \usebibmacro{title}%
  \newunit
  \printlist{language}%
  \newunit\newblock
  \usebibmacro{byauthor}%
  \newunit\newblock
  \usebibmacro{bytranslator+others}%
  \newunit\newblock
  \printfield{version}%
  \newunit\newblock
  \usebibmacro{in:}%
  \usebibmacro{journal+issuetitle}%
  \newunit\newblock
  \usebibmacro{byeditor+others}%
  \newunit\newblock
  \usebibmacro{note+pages}%
  \newunit\newblock
  \iftoggle{bbx:isbn}
  {\printfield{issn}}
  {}%
  \newunit\newblock
  \usebibmacro{doi+eprint+url}%
  \newunit\newblock
  \usebibmacro{addendum+pubstate}%
  \newunit\newblock
  \usebibmacro{pageref}%
  \usebibmacro{finentry}}
```

Come si vede, il driver `@article` è composto da righe in cui si richiamano altre macro (`\usebibmacro`), righe in cui si richiede di stampare una lista (`\printlist`) o un campo (`\printfield`), separate dai comandi `\newunit` e `\newblock`. A loro volta le macro richiamate dal driver possono avere al loro interno altre macro e molti altri comandi, in una specie di gioco di scatole cinesi, del tutto naturale per chi è abituato a L^AT_EX.

Molte delle macro usate all'interno dei driver sono definite nel file `biblatex.def` e nel file `biblatex.sty`, altre vengono definite direttamente nei file `.bbx` e `.cbx`. Per creare stili con `biblatex`

6. Il pacchetto viene fornito con tre ulteriori file: `blx-compat.def` e `blx-natbib.def`, che contengono alcune macro di compatibilità generica e per il pacchetto `natbib`; `biblatex.cfg`, nel quale possono essere inserite le proprie personalizzazioni.

bisogna prepararsi a navigare senza timore tra tutti questi file.

3.1 La punteggiatura

Una voce bibliografica non è altro che un insieme di “segmenti” che vengono distinti in *unità*, il segmento più piccolo, e *blocchi*, i segmenti più ampi. Ogni unità, creata dal comando `\newunit`, è costituita di campi, che possono essere liste di nomi, date, intervalli numerici, ecc. Tutti questi elementi possono venire separati da opportuni delimitatori, che sono generalmente i comuni segni di punteggiatura e gli spazi.

All'interno delle macro di `biblatex` i delimitatori delle diverse unità e dei diversi campi della voce bibliografica non vengono inseriti come semplice testo, ma attraverso dei comandi che possono essere ridefiniti dall'utente. Evidentemente ciò ha motivazioni molto serie.

I seguenti sono solo alcuni dei comandi (senza argomento) definiti nel file `biblatex.def`:

`\newunitpunct` il separatore delle diverse unità della voce bibliografica. È il comando eseguito da `\newunit`;

`\bibnamedash` la lineetta che sostituisce il nome nelle liste di voci bibliografiche relative a uno stesso autore;

`\subtitlepunct` il segno che separa il titolo dal sottotitolo (di default un punto fermo).

`\intitlepunct` il segno che segue la stringa “in” (di default i due punti).

Vi sono poi i comandi che stampano uno dei consueti segni di punteggiatura e gli spazi. Il loro nome è sufficientemente eloquente per evitare di descriverli:

<code>\adddot</code>	<code>\addspace</code>	<code>\addcomma</code>
<code>\addsemicolon</code>	<code>\adddotsspace</code>	<code>\addcolon</code>

Questi ultimi comandi non si limitano in realtà ad inserire un determinato simbolo, ma prevengono, per esempio, che vi siano doppie punteggiature e spazi in eccesso. In generale, i comandi che iniziano con `\add` rimuovono automaticamente gli eventuali spazi che li precedono, cosa che avviene internamente attraverso un altro comando importante: `\unspace`.

Altri due comandi molto importanti sono i seguenti:

`\newunit` registra la fine di una unità e inserisce il delimitatore di default, `\newunitpunct`;

`\setunit{<punctuation>}` simile al precedente ma con `<punctuation>`, ovvero uno o più dei delimitatori descritti sopra, al posto di `\newunitpunct`.

Vi sono poi alcuni comandi con un argomento. Il comando `\mkbibemph` stampa il suo argomento in carattere italico, `\mkbibquote` tra virgolette, `\mkbibparens` tra parentesi tonde e `\mkbibbrackets` tra parentesi quadre.

Vediamo ora alcuni esempi d'uso di tali comandi. Una delle prime esigenze che l'utente italiano sente, in particolare nel contesto umanistico, è di avere la virgola come segno di separazione tra le diverse unità della voce bibliografica e il punto fermo come separatore del titolo e del sottotitolo. Per ottenere questo è sufficiente ridefinire i relativi comandi in questo modo:

```
\renewcommand*{\newunitpunct}{\addcomma\space}
\renewcommand*{\subtitlepunct}{\adddot\space}
```

Si noti che non è stato usato il comando `\addspace`, ma il semplice `\space`. In questo caso, infatti, visto che l'ultimo comando eseguito stampa rispettivamente una virgola o un punto, non è necessario che il comando che inserisce lo spazio elimini gli spazi che lo precedono, perché in questo caso non ce ne sono.

Nelle voci `@article` di bibliografie italiane, il nome della rivista non è generalmente preceduto dall'espressione "in:", la quale compare invece nelle voci `@incollection`, spesso senza i due punti. Questa stringa viene stampata dalla macro `in:`, definita nel file `biblatex.def` in questo modo:

```
\newbibmacro*{in:}{%
  \printtext{%
    \bibstring{in}\intitlepunct}}
```

Come si può facilmente vedere, dopo aver stampato l'espressione "in" attraverso il comando `\bibstring`, viene eseguito il comando `\intitlepunct`, che stampa di default i due punti. È pertanto opportuno eliminare dal driver `@article` visto sopra la riga in cui viene caricata questa macro. Se poi vogliamo eliminare i due punti, lo possiamo fare molto semplicemente ridefinendo il comando che li genera (`\intitlepunct`):

```
\renewcommand*{\intitlepunct}{\nopunct\addspace}
```

È necessaria qui un'avvertenza generale: quando si ridefiniscono le macro di *biblatex* bisogna porre particolare attenzione all'uso dei simboli di commento (%). La mia modesta esperienza personale consiglia a chi vuole scrivere stili per *biblatex* di abbondare con i commenti, per evitare spiacevoli conseguenze. L'assenza di un commento là dove dovrebbe esserci potrebbe portare alla presenza indesiderata di uno spazio nell'output finale e, peggio ancora, potrebbe richiedere contorsioni per evitarlo (per esempio attraverso `\unspace`), con il rischio di non riuscirci. Se però si conoscono molto bene le finenze della sintassi di *LATEX* si può essere più parsimoniosi.

3.2 Il formato dei campi

Ogni campo della voce bibliografica ha un suo proprio stile o formato. In particolare alcuni campi, come quello per il titolo di un volume, sono in corsivo, altri in tondo. Altri ancora potrebbero essere in neretto o in maiuscoletto, come quello per i nomi degli autori. Questo formato, a seconda che si tratti di campi letterali, liste o nomi può essere modificato grazie ai comandi:

```
\DeclareFieldFormat[<entry type>]{<format>}{<code>}
\DeclareListFormat[<entry type>]{<format>}{<code>}
\DeclareNameFormat[<entry type>]{<format>}{<code>}
```

dove *<entry type>* è il tipo di voce bibliografica (`book`, `article`, ecc.); *<format>* è il campo che deve essere formattato; *<code>* è il codice che deve essere eseguito per quel campo e quel tipo di voce bibliografica. Se *<entry type>* non viene specificato, il formato verrà ridefinito per tutti i tipi di voce bibliografica.

Il campo `url`, per esempio, è definito nel file `biblatex.def` in questo modo:

```
\DeclareFieldFormat{url}{%
  \mkbibacro{URL}\addcolon\space url{#1}}
```

Il comando `\mkbibacro` è a sua volta definito in modo da stampare il suo argomento (un acronimo) in maiuscoletto, sempre che nel font in uso sia disponibile. Come si vede, la sintassi è molto più intuitiva di quella dei file `.bst` e per avere semplicemente l'indirizzo web, quindi, è sufficiente modificare il formato del campo `url` in modo che abbia effetto sui campi `url` di tutti i tipi di voce bibliografica, ovvero senza indicare alcun tipo di voce nel campo opzionale *<type>*, in questo modo:

```
\DeclareFieldFormat{url}{\url{#1}}
```

Gli stili bibliografici standard di *biblatex* prevedono poi che i titoli delle voci `@inbook` vengano stampati in tondo e tra virgolette, ovvero il formato del campo `title` è definito in questo modo:⁷

```
\DeclareFieldFormat[inbook]{title}{%
  \mkbibquote{#1}\isdot}}
```

Se si vuole modificare il comportamento solo relativamente a questo tipo di voce ed avere i titoli in corsivo anziché tra virgolette servirà pertanto il codice seguente:

```
\DeclareFieldFormat[inbook]{title}{\mkbibemph{#1}}
```

Tipicamente, in una raccolta di opere di uno stesso autore (che va registrata come `@book` e che generalmente è una edizione critica o comunque un'importante edizione di riferimento per quell'autore), possono comparire sia volumi singoli sia articoli apparsi originariamente su rivista, oltre a scritti inediti, lettere ecc.

7. Quali virgolette dipende dall'opzione passata al pacchetto `csquotes`, che con *biblatex* è sempre consigliabile caricare.

Per ciascuna di queste voci si potrebbe voler usare una formattazione distinta. Oltre al driver `@inbook`, `biblatex` prevede il driver `@suppbook`, per voci bibliografiche relative a prefazioni, introduzioni, ecc., e il driver `@bookinbook`, per le voci raccolte in un `book` e pubblicate originariamente come volume singolo. Questi driver sono trattati di default come alias del driver `@inbook`, ma vi si possono naturalmente assegnare formattazioni distinte. Se volessimo mantenere il formato standard per il titolo delle voci `inbook` visto sopra, e stampare in corsivo il titolo delle voci `bookinbook` e `suppbook` sarebbe sufficiente il seguente codice:

```
\DeclareFieldFormat[bookinbook]{title}{%
\mkbibemph{#1}}
\DeclareFieldFormat[suppbook]{title}{%
\mkbibemph{#1}}
```

Ora che ci siamo fatti un'idea, anche se vaga, di alcuni aspetti del funzionamento di `biblatex` possiamo tentare di descrivere come realizzare uno stile bibliografico personalizzato. Ci faranno da sfondo, di nuovo, gli stili `philosophy`.

4 Creare uno stile con `biblatex`

4.1 Lo stile bibliografico

Tutti i codici che abbiamo descritto finora possono essere usati direttamente nel preambolo. È facile intuire che andando avanti di questo passo il preambolo diventerebbe alquanto ingombrante. In casi come questo è opportuno riflettere sulla possibilità di creare un proprio stile bibliografico che contenga tutte le impostazioni che ci servono. Per convenzione, nel file di stile bibliografico viene inserito anche il codice per gestire gli aspetti globali dello stile, e questo è il motivo per cui i file `.bbx` sono in genere molto più ampi dei file `.cbx`.

Va detto che creare uno stile per `biblatex` è molto semplice, ma creare uno stile per `biblatex`, che risponda a certe esigenze, invece, può essere anche molto complesso, e in alcuni casi addirittura impossibile. Di seguito vedremo come creare uno stile che comprende alcune caratteristiche degli stili `philosophy-classic` e `philosophy-modern`. Partiremo quindi dalla struttura dello stile `authoryear`, su cui si basano gli stili.

Creiamo quindi un file che chiameremo `mystyle.bbx` e che contenga le seguenti righe:

```
\ProvidesFile{mystyle.bbx}
\RequireBibliographyStyle{authoryear}

\endinput
```

A questo punto il nostro stile bibliografico è già utilizzabile ma non sarebbe diverso dallo stile `authoryear` che viene richiamato dal comando `\RequireBibliographyStyle`.⁸ Tutti i codici che

8. Gli stili `philosophy` sono basati in realtà sullo stile `authoryear-comp`, ma, relativamente alla bibliografia, questo stile è identico ad `authoryear`.

seguiranno la seconda riga saranno quindi ridefinizioni di macro e di comandi usati dallo stile `authoryear`, che richiama a sua volta lo stile `standard`, oppure codici completamente nuovi che implementano nuove funzioni non presenti nello stile `authoryear`.

All'interno del file di stile bibliografico appena creato, possiamo inserire, per esempio, il driver `@article` opportunamente modificato, la macro `in`: ridefinita e i comandi per modificare i formati dei campi che abbiamo visto precedentemente. Di seguito verranno descritti altri codici che potranno arricchire il nostro nuovo file di stile bibliografico e che verrà riportato integralmente alla fine di questa sezione.

4.1.1 Come creare una nuova opzione

Uno stile bibliografico può anche essere corredato da opzioni che permettono all'utente di personalizzare alcuni elementi dello stile in maniera molto semplice, senza dover ridefinire le macro o i formati dei campi. Con `biblatex` si possono creare sia opzioni globali sia opzioni che possono essere inserite all'interno del record bibliografico e che valgono quindi solo per la voce che le contiene. Ci soffermeremo su quelle del primo tipo.

I nomi in maiuscolo

Vediamo ora come creare una nuova opzione che permetta di ottenere i nomi che compaiono nella voce bibliografica in maiuscolo.

Innanzitutto va dichiarata una nuova variabile booleana, cioè una variabile che può assumere due valori, vero o falso:

```
\newbool{bbx:scauthorsbib}
```

La prima parte del nome di questa variabile (`bbx:`) è semplicemente un'etichetta convenzionale per ricordarsi che la variabile viene definita all'interno di un file di stile bibliografico.

Il secondo passaggio consiste nella dichiarazione dell'opzione, che chiameremo `scauthorsbib`, con il comando `\DeclareBibliographyOption`:

```
\DeclareBibliographyOption{scauthorsbib}{true}
{\csuse{bool#1}{bbx:scauthorsbib}}
```

Come si vede, questo comando ha un argomento opzionale. Si tratta del valore di default (in questo caso `true`) assegnato all'opzione `scauthorsbib` nel caso non venga indicato esplicitamente alcun valore. Il che significa che in questo modo sarà equivalente scrivere `scauthorsbib=true` o semplicemente `scauthorsbib`.

Per fare in modo che questa opzione non valga di default è necessario assegnare alla variabile il valore falso. Con `biblatex` si ha a disposizione un apposito comando per inizializzare le opzioni dello stile ovvero decidere quali opzioni impostare di default:

```
\ExecuteBibliographyOptions{scauthorsbib=false}
```

Fortunatamente *biblatex* ha previsto la possibilità di modificare il formato dei nomi, sebbene per scopi più complessi questa procedura non possa essere di aiuto. Nel formato che gestisce la stampa dei nomi, che avviene tramite il comando `\printnames`, le varie parti dei nomi (il nome proprio, il cognome, il prefisso, ecc.) sono state racchiuse in appositi comandi. Questi comandi vengono definiti nel file `biblatex.def` in questo modo:

```
\newcommand*{\mkbibnamefirst}[1]{#1}
\newcommand*{\mkbibnamelast}[1]{#1}
\newcommand*{\mkbibnameprefix}[1]{#1}
\newcommand*{\mkbibnameaffix}[1]{#1}
```

Per ottenere quello che ci siamo proposti è quindi sufficiente ridefinire questi comandi. Ora abbiamo tutto quello che ci serve per creare la nuova opzione.

Concettualmente il problema è molto semplice. Dobbiamo dire a LATEX che se la variabile `bbx:scauthorsbib` assume il valore *vero* i comandi sopra riportati devono essere ridefiniti in modo da stampare il loro argomento in maiuscoletto. Ciò può essere fatto attraverso attraverso il condizionale `\ifbool`:

```
\ifbool{bbx:scauthorsbib}%
  {\renewcommand*{\mkbibnamelast}[1]{\textsc{#1}}}%
  {\renewcommand*{\mkbibnamefirst}[1]{\textsc{#1}}}%
  {\renewcommand*{\mkbibnameprefix}[1]{\textsc{#1}}}%
  {\renewcommand*{\mkbibnameaffix}[1]{\textsc{#1}}}%
  {}%
```

Vogliamo però fare in modo che questa opzione valga solo all'interno della bibliografia e non influisca sulle citazioni. Per questo è sufficiente inserire il codice precedente nell'argomento di

```
\AtBeginBibliography{<code>}
```

La nostra nuova opzione è così pronta per essere passata a *biblatex*:

```
\usepackage[bibstyle=mystyle,
             citestyle=authoryear,
             scauthorsbib=true]{biblatex}
```

e sarà valida solo all'interno della bibliografia.

Naturalmente questa opzione permette di stampare in maiuscoletto *tutti* i nomi che compaiono in bibliografia, compresi quelli degli eventuali traduttori, curatori, ecc. Gli stili *philosophy* sono da questo punto di vista più sobri perché forniscono un'opzione per rendere in maiuscoletto solo i nomi che compaiono all'inizio della voce bibliografica. Sebbene l'effetto possa apparire più semplice, la soluzione è molto più complicata, perché richiede di modificare opportunamente il formato `sortname`,

definito nel file `biblatex.def`, responsabile del formato dei nomi all'interno della bibliografia.⁹

Qui ci troviamo a un livello abbastanza profondo del pacchetto *biblatex* e va oltre le pretese di questo articolo spiegarne i dettagli, che richiederebbero l'illustrazione di un certo numero di macro tutt'altro che semplici. Dal punto di vista concettuale, tuttavia, non c'è niente di diverso da quanto visto finora: si tratta di modificare opportunamente alcune macro.

Bibliografie commentate

Alcuni stili bibliografici utilizzano il campo `annotation` per produrre bibliografie commentate, molto utili sebbene poco diffuse. Generalmente si tratta di bibliografie in cui dopo ogni voce bibliografica viene inserita una breve descrizione del testo, come nell'esempio riportato a p. 40. Questo è il caso degli stili *philosophy*.

Vogliamo quindi fare in modo che alla fine della voce bibliografica e in un nuovo paragrafo venga stampato il contenuto del campo `annotation` e creare una nuova opzione per poter decidere se stampare o no tale campo, analoga all'opzione `scauthorsbib` vista sopra.

Vediamo il primo punto. Il campo `annotation` deve essere stampato alla fine della voce bibliografica. L'ultima macro usata da ogni driver è la macro `finentry`, definita nel file `standard.bbx` in questo modo:

```
\newbibmacro*{finentry}{\finentry}
```

Questa macro si limita quindi ad eseguire semplicemente il comando `\finentry`, il quale a sua volta esegue `\finentrypunct` (ovvero, di default, stampa un punto fermo).

La macro `finentry` andrà pertanto ridefinita in modo da stampare il campo `annotation` solo se viene dichiarata l'opzione `annotation=true`:

```
\renewbibmacro*{finentry}{%
  \ifbool{bbx:annotation}
    {\iffieldundef{annotation}
     {\finentry}
     {\setunit{\addperiod\par\nobreak}
      \printtext[annotation]{%
        \printfield{annotation}\finentry\par}}
     {\finentry}}
```

Cerchiamo ora di spiegarla. Il primo condizionale (`\ifbool`) valuta se è attiva o meno l'opzione `annotation`. Se l'opzione è attiva entra in gioco il condizionale

```
\iffieldundef{<field>}{<true>}{<false>}
```

che esegue `<true>` se `<field>` non è definito, `<false>` altrimenti.

Nel nostro caso, se non è definito il campo `annotation` viene eseguito il comando `\finentry`,

9. Lo stile *philosophy-verbose* richiede anche di modificare il formato `labelname`, che gestisce il formato dei nomi in tutte le citazioni.

ovvero, in altri termini, se l'opzione `annotation` è attiva e non viene usato il campo `annotation` la macro si comporta come la sua definizione di default. Se invece è stato utilizzato il campo `annotation` viene stampato un punto fermo e si va a capo. Il comando `\nobreak` serve per evitare che la pagina inizi con un'annotazione.

A questo punto viene eseguito il comando `\printtext`. Questo comando esegue il codice che ha come argomento, ovvero stampa il campo `annotation` ed esegue il comando `\finentry`.

Per la sua natura, è bene poi che il campo `annotation` sia distinto dalle voci bibliografiche, usando un font diverso o un corpo minore. Ciò è naturalmente possibile attraverso `\DeclareFieldFormat`, magari definendo un nuovo comando:

```
\newcommand{\annotationfont}{\sffamily}
\DeclareFieldFormat{annotation}{\annotationfont #1}
```

In questo modo abbiamo definito un nuovo formato (`annotation`) che è quello passato all'argomento opzionale di `\printtext`. Se ora vogliamo cambiare lo stile con il quale viene stampato il campo `annotation` è sufficiente ridefinire il comando `\annotationfont`. Ugualmente, se vogliamo comporre tale campo in corpo minore. È per casi come quest'ultimo che si rende necessario il secondo `\par`, al fine di ottenere l'interlinea corretta all'interno del campo `annotation`.

4.1.2 Miglioramento dell'opzione `backref`

L'opzione `backref` di `biblatex` permette di stampare nella bibliografia i riferimenti alle pagine in cui il testo è stato citato all'interno del proprio documento.¹⁰

Gli stili `philosophy` permettono di ottenere i riferimenti tra parentesi tonde, in un nuovo periodo, e in una forma meno sintetica: "(Citato a p. 5.)".¹¹ Ciò è stato possibile ridefinendo le stringhe `backrefpage` e `backrefpages` nel file di localizzazione (vedi § 4.3) e modificando opportunamente la macro che gestisce questi riferimenti, ovvero `pageref`, definita nel file `biblatex.def` in questo modo:

```
\newbibmacro*{pageref}{%
  \iflistundef{pageref}
  {}
  {\printtext[parens]{%
    \ifnumgreater{\value{pageref}}{1}
    {\bibstring{backrefpages}\ppspace}
    {\bibstring{backrefpage}\ppspace}%
  }}
  \printlist[pageref][-{\value{listtotal}}]{pageref}}}
```

Dobbiamo innanzitutto fare in modo che prima dei riferimenti venga stampato un punto e uno spazio. Questa operazione è semplice, perché basta

10. Ricordo che il pacchetto `backref` è incompatibile con `biblatex`.

11. Il punto è all'interno della parentesi proprio perché, dal punto di vista sintattico, questi riferimenti costituiscono di fatto un periodo.

aggiungere `\setunit{\addperiod\space}` prima di `\printtext`. Ecco la macro a cui dovremmo arrivare:

```
\renewbibmacro*{pageref}{%
  \iflistundef{pageref}
  {}
  {\setunit{\addperiod\space}%
  \printtext[backrefparens]{%
    \ifnumgreater{\value{pageref}}{1}
    {\bibstring{backrefpages}\ppspace}
    {\bibstring{backrefpage}\ppspace}%
  }}
  \printlist[pageref][-{\value{listtotal}}]{pageref}%
  \adddot\NOPUNCT}}
```

Questa macro controlla innanzitutto se vi sono delle pagine in cui una voce è stata citata. In caso affermativo aggiunge un punto fermo e uno spazio. Quindi esegue il comando `\printtext` e al suo interno, attraverso il condizionale `\ifnumgreater`, valuta i due casi possibili. Infine stampa la lista delle pagine e aggiunge un punto. Il comando `\NOPUNCT` successivo serve per evitare che all'esterno della parentesi di chiusura vi sia il punto di fine periodo che chiude generalmente la voce bibliografica. Il punto, infatti, in questo caso è già (deve essere) all'interno della parentesi.

L'argomento opzionale del comando `\printtext` definisce il formato delle parentesi da usare in questi riferimenti ed è stato introdotto per evitare che una ridefinizione del comando `\mkbibparens` modifichi anche queste parentesi. L'opzione `square` fornita dagli stili `philosophy-classic` e `philosophy-modern`, che permette di ottenere le citazioni con parentesi quadre anziché tonde, non ha infatti alcun effetto sulle parentesi di questi riferimenti proprio grazie al nuovo formato `backrefparens`. Un modo semplice per definire questo formato è il seguente:

```
\DeclareFieldFormat{backrefparens}{( #1 )}
```

Qui si è fatto uso delle parentesi, inserendole come puro testo. Negli stili `philosophy` si è seguita una strada un po' più lunga e corretta.

4.1.3 I campi per la traduzione

Veniamo ora alla caratteristica principale degli stili `philosophy` che consiste nella possibilità di gestire voci bibliografiche che comprendano le eventuali informazioni sulla traduzione italiana di un testo straniero o sulla ristampa.

Una tipica voce bibliografica in cui si faccia uso di questi campi è simile alla seguente:

Brighurst, Robert, *The Elements of Typographic Style*, 2a ed., Hurtley & Marks Publisher, Vancouver 1996; trad. it. *Gli elementi dello stile tipografico*, Sylvestre Bonnard, Milano 2009.

L'idea alla base degli stili `philosophy` è di utilizzare i campi `origtitle`, `origpublisher`, `origlocation` e `origdate` per inserirvi le informa-

zioni sulla traduzione italiana.¹² Il nome di questi campi potrebbe creare qualche disagio “linguistico”, almeno in un primo tempo. Si è scelto tuttavia di non invertire i ruoli dei campi per rispondere anche alle esigenze di chi non compone spesso voci bibliografiche simili all’esempio fatto, i quali possono in questo modo continuare ad utilizzare i propri archivi bibliografici anche con gli stili *philosophy*.

Consideriamo ora il caso più semplice di una voce *book* che faccia uso di questi campi. Ecco una parte della macro con cui viene definito il driver *@book*:

```
\DeclareBibliographyDriver{book}{%
  \usebibmacro{bibindex}%
  ...
  \usebibmacro{addendum+pubstate}%
  \newunit\newblock
  \usebibmacro{pageref}%
  \usebibmacro{finentry}}
```

Le informazioni sulla traduzione devono essere stampate alla fine della voce bibliografica subito dopo gli eventuali campi *addendum* e *pubstate*. La cosa più semplice è quindi ridefinire la macro *addendum+pubstate* in modo che includa il codice per gestire le informazioni sulla traduzione. Poiché questa macro viene caricata da tutti i driver, questo ci eviterà di modificare ogni singolo driver.

Prima di tutto dobbiamo creare la macro necessaria per gestire i campi relativi alla traduzione, che chiameremo *origfields*:

```
\newbibmacro*{origfields}{%
  \iffielddundef{origtitle}{-}{%
    \setunit{\addsemicolon\space}%
    \bibstring{transtolang}\setunit{\addspace}%
    \printfield{origtitle}{origtitle}%
    \setunit{\addcomma\space}%
  \iflistundef{origpublisher}%
    {\printlist{origlocation}%
     \setunit*\addspace}%
    \printorigdate}%
  {\printlist{origpublisher}%
   \setunit*\addcomma\space}%
  \printlist{origlocation}%
  \setunit*\addspace}%
  \printorigdate}}
```

Questa macro controlla innanzitutto che sia presente il campo *origtitle*, ovvero il titolo della traduzione; in questo caso stampa un punto e virgola seguito dallo spazio, la stringa *transtolang*, che verrà definita nel file di localizzazione con “trad. it.” (vedi § 4.3) e il campo *origtitle* seguito da una virgola. Poi controlla se è definito il campo *origpublisher*; in caso affermativo stampa la lista contenuta in *origpublisher*, aggiunge una virgola e lo spazio, e stampa la data della traduzione (*\printorigdate*). Nel caso non sia definito *origpublisher* viene stampato il campo *origlocation*, lo spazio e la data della traduzione.

12. Qui ci limitiamo al caso di traduzioni italiane, tuttavia gli stili sono compatibili con qualsiasi lingua riconosciuta da *babel*.

A questo punto possiamo ridefinire la macro *addendum+pubstate*, richiamando al suo interno la macro *origfields* appena creata:

```
\renewbibmacro*{addendum+pubstate}{%
  \printfield{addendum}%
  \newunit\newblock
  \printfield{pubstate}%
  \newunit\newblock
  \usebibmacro{origfields}}
```

4.1.4 Lo stile *philosophy-modern*

Nel breve resoconto storico degli stili *philosophy* (nelle due versioni *classic* e *modern*) abbiamo spiegato come questi stili siano nati per produrre una bibliografia a “blocchi”, come quella di figura 1. È arrivato il momento di mostrare come sia possibile arrivare a (una parte di) quel risultato.

In questo tipo di bibliografia ogni gruppo di voci bibliografiche può far riferimento a un autore o a un curatore, ma ci sono anche voci che vengono indicate attraverso un’etichetta convenzionale (il campo *label*), nel caso la voce stessa non abbia né autore né curatore.

Le macro responsabili della gestione delle liste di autori e curatori si chiamano rispettivamente *author* ed *editor*. Vediamo come modificare la macro *author* per ottenere il risultato mostrato in figura 1. La macro *author* viene definita nel file *biblatex.def* e ridefinita nel file *authoryear.bbx* come segue:

```
\renewbibmacro*{author}{%
  \ifthenelse{\ifuseauthor\AND\NOT
  \ifnameundef{author}}%
  {\ifthenelse{\iffieldequals{fullhash}
  {\bbx@lasthash}\AND
  \NOT\iffirstonpage\AND
  \(\NOT\boolean{bbx@inset}\OR
  \iffieldequalstr{entrysetcount}{1}\)}}%
  {\bibnamedash}%
  {\usebibmacro{bbx:savehash}%
   \printnames{author}%
   \iffielddundef{authortype}
   {\setunit{\addspace}%
    \setunit{\addcomma\space}}}%
  \iffielddundef{authortype}
  {}
  {\usebibmacro{authorstrg}%
   \setunit{\addspace}}}%
  {\global\undef\bbx@lasthash
   \usebibmacro{labeltitle}%
   \setunit*\addspace}%
  \usebibmacro{date+extrayear}}
```

Prima di tutto va definita una nuova lunghezza elastica e un nuovo comando:

```
\newlength{\postnamesep}
\setlength{\postnamesep}{0.5ex plus 2pt minus 1pt}
\newcommand{\postsep}{%
  \par\nobreak\vskip\postnamesep%
  \hskip-\bibhang\ignorespaces}
```

Il comando *\postsep*, che genera un nuovo capoverso e aggiunge uno spazio verticale del valore di *\postnamesep*, deve essere eseguito dopo il co-

mando `\printnames`, che in questo caso stampa la lista degli autori.

La macro `author` originaria, infine, stampa una lineetta (`\bibnamedash`) al posto del nome dell'autore per le voci successive alla prima relative a uno stesso autore. Questa lineetta va naturalmente eliminata in una bibliografia come quella che stiamo costruendo, perché verrà sostituita dall'etichetta dell'anno.

Siamo così riusciti a far sì che dopo la lista degli autori la voce bibliografica vada a capo. Non ci rimane ora che fare in modo che l'etichetta dell'anno sia inserita in una "colonna" di larghezza predeterminata a sinistra della voce bibliografica. La macro che gestisce l'etichetta dell'anno si chiama `date+extrayear`, definita nel file `authoryear.bbx`. La cosa più semplice è ridefinirla racchiudendo la data in un box della larghezza del rientro sinistro della bibliografia (`\bibhang`):

```
\renewbibmacro*{date+extrayear}{%
  \iffieldundef{year}%
  {%
    {\makebox[\bibhang]{r}{%
      \printtext{\printfield{labelyear}%
        \printfield{extrayear}%
        \hskip\yeararticle}}}}
```

All'interno di questo box l'etichetta dell'anno viene allineata a destra. Per aggiungere un po' di spazio è stata definita la lunghezza `\yeararticle`, questa volta rigida, che può eventualmente essere modificata dall'utente a seconda delle sue esigenze.¹³ Per riservare un po' di spazio in più all'etichetta dell'anno è inoltre necessario impostare la lunghezza `\bibhang` a un valore di almeno `4\parindent`

Negli stili `authoryear`, dopo l'etichetta dell'anno viene eseguito il comando `\labelnamepunct` (di default equivalente a `\newunitpunct`). Questo comando va pertanto ridefinito in modo che non stampi nulla, nemmeno lo spazio:

```
\renewcommand*{\labelnamepunct}{\unspace}
```

Ci siamo limitati ad intervenire su voci bibliografiche con un autore. Per quelle con curatore (generalmente `collection`) dovrà essere modificata la macro `bbx:editor` e in questo caso sono richiesti ulteriori aggiustamenti, dei quali il lettore interessato potrà rendersi conto leggendo il file `philosophy-modern.bbx`.

Rimangono solo da sistemare le macro che entrano in gioco con voci che usano il campo `label`, per le quali è necessario modificare la macro `labeltitle` e aggiungere un semplice trucco nel driver `@collection`. Riporto infine di seguito il file dello stile bibliografico `mystyle.bbx` che abbiamo creato con i codici precedenti:

13. Gli stili `philosophy` forniscono anche l'opzione `yearleft`, per ottenere l'allineamento a sinistra, come avviene nell'esempio di Eco (2005, 193). Per evidenti motivi estetici, sconsiglio tuttavia l'uso di quest'opzione.

```
\ProvidesFile{mystyle.bbx}
\RequireBibliographyStyle{authoryear}

\DeclareLanguageMapping{italian}{italian-mystyle}

\newbool{bbx:scauthorsbib}
\newbool{bbx:annotation}

\DeclareBibliographyOption{scauthorsbib}[true]
  {\csuse{bool#1}{bbx:scauthorsbib}}
\DeclareBibliographyOption{annotation}[true]{%
  \csuse{bool#1}{bbx:annotation}}

\ExecuteBibliographyOptions{%
  scauthorsbib=false,annotation=false}

\newlength{\yeararticle}
\newlength{\postnamesep}
\setlength{\yeararticle}{0.8em}%
\setlength{\postnamesep}{0.5ex plus 2pt minus 1pt}
\setlength{\bibitemsep}{\postnamesep}
\setlength{\bibnamesep}{1.5ex plus 2pt minus 1pt}
\setlength{\bibhang}{4\parindent}

\newcommand{\postsep}{\par\nobreak\vskip\postnamesep}
\hskip-\bibhang\ignorespaces}
\newcommand{\annotationfont}{\sffamily}
\renewcommand*{\labelnamepunct}{\unspace}
\renewcommand*{\intitlepunct}{\nopunct\addspace}

\DeclareFieldFormat{url}{\url{#1}}
\DeclareFieldFormat[bookinbook]{title}{\mkbibemph{#1}}
\DeclareFieldFormat[supbook]{title}{\mkbibemph{#1}}
\DeclareFieldFormat[origtitle]{\mkbibemph{#1}}
\DeclareFieldFormat{annotation}{\annotationfont #1}
\DeclareFieldFormat{backrefparens}{(#1)}

\AtBeginBibliography{\ifbool{bbx:scauthorsbib}%
  {\renewcommand*{\mkbibnamelast}[1]{\textsc{#1}}}%
  {\renewcommand*{\mkbibnamefirst}[1]{\textsc{#1}}}%
  {\renewcommand*{\mkbibnameprefix}[1]{\textsc{#1}}}%
  {\renewcommand*{\mkbibnameaffix}[1]{#1}}}%
  {}%

\renewbibmacro*{date+extrayear}{%
  \iffieldundef{year}%
  {%
    {\makebox[\bibhang]{r}{%
      \printtext{\printfield{labelyear}%
        \printfield{extrayear}%
        \hskip\yeararticle}}}}
```

```
\renewbibmacro*{addendum+pubstate}{%
  \printfield{addendum}%
  \newunit\newblock
  \printfield{pubstate}%
  \newunit\newblock
  \usebibmacro{origfields}}

\renewbibmacro*{finentry}{%
  \ifbool{bbx:annotation}%
  {\iffieldundef{annotation}%
    {\finentry}%
    {\setunit{\addperiod\par\nobreak}%
      \printtext[annotation]{%
        \printfield{annotation}\finentry\par}}}%
    {\finentry}}

\renewbibmacro*{pageref}{%
  \iflistundef{pageref}
  {}
  {\setunit{\addperiod\space}%
    \printtext[backrefparens]{%
      \ifnumgreater{\value{pageref}}{1}
      {\bibstring{backrefpages}\ppspace}
      {\bibstring{backrefpage}\ppspace}%
      \printlist[pageref]{-\value{listtotal}}{pageref}%
      \adddot\NOPUNCT}}

\newbibmacro*{origfields}{%
  \iffieldundef{origtitle}{%
    {\setunit{\addsemicolon\space}%
      \bibstring{transtolang}\setunit{\addspace}%
      \printfield[origtitle]{origtitle}%
      \setunit{\addcomma\space}}%
    {\iflistundef{origpublisher}%
      {\printlist{origlocation}%
        \setunit{\addspace}}%
```

```

\printorigdate}%
{\printlist{origpublisher}%
\setunit*{\addcomma\space}%
\printlist{origlocation}%
\setunit*{\addspace}%
\printorigdate}}

\renewbibmacro*{author}{%
\ifthenelse{\ifuseauthor\AND\NOT\ifnameundef{author}}
{\ifthenelse{\iffieldequals{fullhash}{\bbx@lasthash}\AND
\NOT\iffirstonpage\AND
\(\NOT\boolean{bbx@inset}\OR
\iffieldequalstr{entrysetcount}{1}\)}}
{}%MOD
{\usebibmacro{bbx:savehash}%
\printnames{author}%
\postsep%ADD
\iffieldundef{authortype}
{\setunit{\addspace}}
{\setunit{\addcomma\space}}}%
\iffieldundef{authortype}
{}
{\usebibmacro{authorstrg}%
\setunit{\addspace}}}%
\global\undef\bbx@lasthash
\usebibmacro{labeltitle}%
\setunit*{\addspace}}%
\usebibmacro{date+extrayear}}
\endinput

```

4.2 Lo schema di citazione

Abbiamo visto come creare uno stile bibliografico. Vediamo ora come costruire il relativo schema di citazione, che chiameremo `mystyle.cbx`. Il nostro schema di citazione, a dire il vero molto semplice, sarà basato sullo schema `authoryear-comp`:

```

\ProvidesFile{mystyle.cbx}
\RequireCitationStyle{authoryear-comp}

\newbool{cbx:scauthorscite}

\DeclareBibliographyOption{scauthorscite}[true]{%
\csuse{bool#1}{cbx:scauthorscite}}

\ExecuteBibliographyOptions{scauthorscite=false}

\AtEveryCite{%
\ifbool{cbx:scauthorscite}%
{\renewcommand*{\mkbibnamelast}[1]{\textsc{#1}}%
\renewcommand*{\mkbibnamefirst}[1]{\textsc{#1}}%
\renewcommand*{\mkbibnameprefix}[1]{\textsc{#1}}%
\renewcommand*{\mkbibnameaffix}[1]{\textsc{#1}}}%
{}%
}
\endinput

```

Questo codice dovrebbe essere ormai abbastanza eloquente. A differenza dello stile bibliografico, in questo caso dobbiamo richiamare uno schema di citazione attraverso il comando

```
\RequireCitationStyle{style}
```

Viene successivamente definita un'ulteriore opzione, `scauthorscite`, analoga a `scauthorsbib` ma valida solo per le citazioni. Questo si ottiene attraverso il condizionale `\ifbool` passato come argomento di `\AtEveryCite` il quale, come suggerisce il nome, permette di eseguire il codice in esso contenuto per ogni citazione.

Abbiamo così introdotto due opzioni, `scauthorsbib` e `scauthorscite`, per ottenere i nomi in maiuscolo nella bibliografia e nelle citazioni, indipendentemente l'una dalle altre. Naturalmente è anche possibile creare un'opzione globale `scauthors`, come avviene negli stili `philosophy`, che attiva entrambe le opzioni precedenti:

```

\newbool{bbx:scauthors}
\DeclareBibliographyOption{scauthors}[true]{%
\csuse{bool#1}{bbx:scauthors}}
\AtBeginDocument{%
\ifbool{bbx:scauthors}
{\booltrue{bbx:scauthorsbib}
\booltrue{cbx:scauthorscite}}{}

```

4.3 Il file di localizzazione

Le macro di biblatex fanno uso di stringhe che vengono definite nei file di localizzazione (`.lbx`). Questo file viene caricato automaticamente in base alla lingua principale del documento. Ogni stringa ha due definizioni, una per la forma estesa e una per la forma abbreviata. Sebbene sia prevista l'opzione globale `abbreviate=false` per utilizzare la forma estesa delle stringhe, potrebbe essere necessario agire localmente, per usare la forma estesa solo e in ogni caso per alcune stringhe.

All'interno dei file di localizzazione le stringhe vengono definite attraverso il comando `\DeclareBibliographyStrings`. Per quanto riguarda la lingua italiana (file `italian.lbx`) ecco la definizione di alcune stringhe:

```

\DeclareBibliographyStrings{%
...
editor = {{curatore}{cur\addot}},
editors = {{curatori}{cur\addot}},
backrefpage = {{citato a pagina}
{cit\addot\space a p\addot}},
backrefpages = {{citato alle pagine}
{cit\addot\space alle pp\addot}},
...
}

```

Vogliamo ora fare in modo che indipendentemente dall'opzione `abbreviate`, la stringa `editor` venga sempre interpretata con "(a cura di)", inoltre vorremmo sempre una versione semi-estesa delle stringhe `backrefpage` e `backrefpages`, ovvero "Citato a p." e "Citato alle pp.". Queste modifiche possono essere fatte direttamente nel preambolo attraverso il comando `\DefineBibliographyStrings`, con un codice come il seguente:

```

\DefineBibliographyStrings{italian}{%
editor = {{a cura di}},
editors = {{a cura di}},
backrefpage = {{citato a p\addot}},
backrefpages = {{citato alle pp\addot}}
}

```

Con questo codice l'opzione `abbreviate` non avrà alcun effetto sulle stringhe `editor`, `editors`, `backrefpage` e `backrefpages`, che verranno stampate sempre nella forma estesa.

Fatta questa premessa, l'ultimo passaggio per ottenere uno stile completo per biblatex è la creazione di un file di localizzazione che contenga le definizioni delle stringhe usate dallo stile. Ecco il codice finale del file `italian-mystyle.lbx`:

```

\ProvidesFile{italian-mystyle.lbx}
\InheritBibliographyExtras{italian}
\NewBibliographyString{transtolang}

```

```
\DeclareBibliographyStrings{%
  transtolang = {{traduzione italiana}
                 {trad\adddotspace it\adddot}},
  editor = {{(a cura di)}{(a cura di)}},
  editors = {{(a cura di)}{(a cura di)}},
  backrefpage = {{Citato a pagina}{Citato a p\adddot}},
  backrefpages = {{Citato alle pagine}
                  {Citato alle pp\adddot}}}
\endinput
```

Il comando

```
\InheritBibliographyStrings{italian}
```

copia le stringhe definite nel file di localizzazione standard `italian.lbx` all'interno del nuovo file, evitando in questo modo di ridefinirle una per una.

Il successivo comando

```
\NewBibliographyString{transtolang}
```

permette di definire una nuova stringa, che può essere utilizzata all'interno delle macro con l'apposito comando `\bibstring`. Questo comporterà l'incompatibilità dello stile con le altre lingue, che evidentemente non riconosceranno la nuova stringa, a meno che non venga definita nel relativo file di localizzazione. Gli stili `philosophy` (versione 0.7a) forniscono i file di localizzazione per la lingua italiana e per la lingua inglese.

Il comando `\DeclareBibliographyStrings` definisce quindi ogni stringa nelle due forme possibili. Quest'ultimo comando, a differenza di `\DefineBibliographyStrings` può essere usato esclusivamente nei file `.lbx`.

Si noti che nella definizione della stringa `transtolang` è stato utilizzato il comando `\adddotspace`. Questo comando non si limita semplicemente ad aggiungere un punto e uno spazio, ma dopo lo spazio viene inserita una penalità stabilita dal contatore `abbrvpenalty`, che di default è impostato al valore di `\hyphenpenalty`. Ciò fa in modo che l'espressione "trad. it." possa venire spezzata, se necessario. Tuttavia se si volesse che questa e altre simili espressioni usate in bibliografia non vengano in nessun caso spezzate è sufficiente impostare globalmente il valore di `abbrvpenalty` a 10000, evitando così di aggiungere molti spazi inseparabili in ogni espressione.

Arrivati a questo punto, il file di localizzazione `mystyle.lbx` appena creato dovrà essere richiamato all'interno del file `mystyle.bbx` (vedi sopra):

```
\DeclareLanguageMapping{italian}{italian-mystyle}
```

5 Conclusione

Con questo articolo abbiamo cercato di dare un'idea di cosa significhi creare uno stile con `biblatex`.

Soltanto un'idea. Molte cose, la maggior parte, sono rimaste *non dette*, fatto che sarebbe apprezzato da certa filosofia post-ermeneutica. Ma il motivo non pretende di essere qui così profondo. Come in altri campi del linguaggio `LaTeX`, anche con `biblatex` ci sono molte vie per arrivare a un determinato traguardo, più o meno raffinate sotto il profilo `TeXnico`, e in genere è molto più semplice ottenere il risultato voluto che spiegare *come* lo si è ottenuto. Ancor più difficile è spiegare il *perché*, e chi scrive non ha di certo avuto una così grande pretesa.

Riferimenti bibliografici

- BRINGHURST, R. (2009). *Gli elementi dello stile tipografico*. Sylvestre Bonnard.
- CEVOLANI, G. (2006). «Norme tipografiche per l'italiano in `LaTeX`». *ArsTeXnica*, 1, pp. 28–42.
- ECO, U. (2005). *Come si fa una tesi di laurea*. Bompiani.
- HUFFLEN, J.-M. (2007). «Guidelines for bibliographical citations in `LaTeX`». *ArsTeXnica*, 4, pp. 103–110.
- LEHMAN, P. (2010). «The `biblatex` package». URL <http://www.ctan.org/tex-archive/macros/latex/exptl/biblatex/>. Versione 0.9a.
- LESINA, R. (2009). *Il Nuovo Manuale di Stile*. Zanichelli.
- MORI, L. F. (2008). «Gestire la bibliografia con `LaTeX`». *ArsTeXnica*, 6, pp. 37–51.
- PANTIERI, L. (2009). «L'arte di gestire la bibliografia con `biblatex`». URL http://www.lorenzopantieri.net/LaTeX_files/Bibliografia.pdf.
- VALBUSA, I. (2010). «Il pacchetto `biblatex-philosophy`». Disponibile su CTAN all'indirizzo: <http://www.ctan.org/tex-archive/macros/latex/exptl/biblatex-contrib/biblatex-philosophy/>.

▷ Ivan Valbusa
Dipartimento di Filosofia, Università
degli Studi di Verona
ivan dot valbusa at univr dot
it