



UNIVERSITÀ DEGLI STUDI DI SIENA
DIPARTIMENTO DI FILOLOGIA E CRITICA DELLE LETTERATURE ANTICHE E MODERNE
UNIVERSITÀ DI PISA, UNIVERSITÀ PER STRANIERI DI SIENA,
CONSIGLIO NAZIONALE DELLE RICERCHE (OVI)

**DOTTORATO DI RICERCA INTERNAZIONALE IN
“FILOLOGIA E CRITICA”
DOTTORATO PEGASO – REGIONE TOSCANA**

CICLO XXXV

Curriculum: “FILOLOGIA ROMANZA”
Settore scientifico-disciplinare: L-FIL-LET/09

UN MODELLO CONCETTUALE PER LE EDIZIONI CRITICHE DIGITALI

TESI PRESENTATA DA: Chiara MARTIGNANO

DOCENTE TUTOR: Prof. Maria Sofia LANNUTTI (Università di Firenze)

Tesi discussa all'Università di Siena, il 13 novembre 2023

Commissione:

Prof. Maria Sofia LANNUTTI (Università di Firenze)

Prof. Lino LEONARDI (Scuola Normale Superiore di Pisa)

Prof. Roberto ROSSELLI DEL TURCO (Università di Torino)

Dott. Emiliano DEGL'INNOCENTI (CNR-OVI)

ANNO ACCADEMICO 2022/2023

Abstract

Nel campo della filologia digitale la maggior parte delle edizioni scientifiche digitali (ESD) viene prodotta con una metodologia denominata “sorgente-*output*”, che consiste nella codifica del testo, perlopiù in formato XML secondo lo standard TEI, e nella pubblicazione sul web attraverso interfacce grafiche sofisticate e ricche di funzionalità. Nonostante questa metodologia si sia consolidata negli ultimi trent’anni e che, nel mentre, siano stati prodotti diversi strumenti informatici per la pubblicazione (o visualizzazione) di ESD generici, ovvero ideati per essere applicati a più edizioni diverse, la maggior parte di esse continua a essere pubblicata attraverso strumenti informatici sviluppati *ad hoc*. Questa tendenza poco sostenibile dipende dal fatto che gli strumenti generici sono spesso percepiti dagli studiosi come poco usabili, difficili da personalizzare e inadatti ai propri obiettivi scientifici. Negli ultimi anni la comunità scientifica si è interrogata su come realizzare strumenti di visualizzazione generici flessibili ed efficaci. Nella letteratura emergono sostanzialmente due risposte: da un lato il bisogno di modelli editoriali condivisi che rappresentino i *desiderata* dei filologi, dall’altro il ricorso alle pratiche dell’ingegneria del software per produrre strumenti informatici più robusti, durevoli e facili da mantenere nel tempo.

In questo elaborato si presenta una strategia di modellizzazione che, tenendo in conto le esigenze sia dei filologi sia degli sviluppatori informatici, consenta di elaborare dei modelli editoriali che guidino concretamente lo sviluppo di strumenti informatici per la visualizzazione di ESD. La strategia prevede che i modelli editoriali, teoricamente inquadrati rispetto all’approccio filologico in cui sono emersi e formalizzati come ontologie, fungano da modelli concettuali o modelli del dominio, per descrivere il campo applicativo di uno strumento informatico. Grazie ai principi del *domain-driven design*, i modelli concettuali prodotti possono essere applicati per progettare l’organizzazione logica e il funzionamento di uno strumento di visualizzazione, in quanto individuano i componenti alla base di una particolare tipologia di edizione e i loro rapporti reciproci.

Nell’elaborato, oltre alla strategia di modellizzazione in sé, viene presentato un modello concettuale di esempio, che descrive le edizioni critiche realizzate secondo l’approccio filologico neo-lachmanniano. Il modello di esempio è stato strutturato in modo tale da poter essere usato come punto di partenza per elaborare altri modelli editoriali, sviluppando così una rete di modelli concettuali.

L'elaborato termina con delle indicazioni di massima su come progettare uno strumento di visualizzazione a partire dal modello concettuale di esempio e con un resoconto dei prossimi passi necessari per consolidare la strategia di modellizzazione e completare la ricerca.

In the field of digital philology, most digital scholarly editions (DSE) are produced using a methodology called “source-output”, which involves encoding the text, mostly in XML format according to the TEI standard, and publishing it on the web through advanced graphical interfaces rich in functionality. This methodology has been established over the past thirty years and several computer tools have been developed for the publication (or visualization) of generic DSEs, designed to be applied to multiple different editions. However, the majority of DSEs continue to be published using custom-developed computer tools. This unsustainable trend is due to the fact that generic tools are often perceived by scholars as less usable, difficult to customize, and unsuitable for their scientific objectives. In recent years, the scientific community has been exploring how to create flexible and effective generic visualization tools. Two main responses have emerged in the literature: the need for shared editorial models that represent philologists' *desiderata*, and the adoption of software engineering practices to produce more robust, durable, and easily maintainable tools.

This thesis presents a modeling strategy that takes into account the needs of both philologists and software developers, allowing the creation of editorial models that concretely guide the development of tools for the visualization of DSEs. The strategy involves using editorial models, theoretically framed within the philological approach in which they have emerged and formalized as ontologies, to serve as conceptual models or *domain models* that describe the application domain of a visualization tool. By applying the principles of *domain-driven design*, the produced conceptual models can be used to design the logical organization and functioning of a visualization tool by identifying the components underlying a particular type of edition and their relationships.

In addition to the modeling strategy itself, this thesis presents an exemplary conceptual model that describes critical editions created according to the neo-Lachmannian philological approach. The example model has been structured in a way that it can be used as a starting point for developing other editorial models, thus developing a network of conceptual models. The thesis concludes with general guidelines on how to design a visualization tool based on the exemplary conceptual model

and provides an overview of the next necessary steps to consolidate the modeling strategy and complete the research.

Indice

Introduzione	17
1 Strumenti informatici per la filologia digitale: stato dell'arte e domande di ricerca	21
1.1 <i>Definizione di “strumento informatico per la filologia digitale”</i>	22
1.2 <i>Cenni storici sul rapporto tra la filologia e lo sviluppo di strumenti informatici</i>	28
1.2.1 Quattro fasi	29
1.2.2 La quinta fase?.....	32
1.2.3 Cenni sui principali cambiamenti nel campo del software e dell'hardware	33
1.2.4 Considerazioni conclusive	35
1.3 <i>Gli strumenti informatici per la filologia digitale oggi</i>	37
1.3.1 Il modello “sorgente-output”	37
1.3.2 Principali categorie di strumenti per le edizioni critiche digitali.....	40
1.3.3 Gli strumenti di produzione.....	44
1.3.4 Gli strumenti di visualizzazione.....	51
1.3.5 Gli strumenti di produzione e visualizzazione (o strumenti misti).....	55
1.3.6 Piattaforme per la pubblicazione e infrastrutture istituzionali	56
1.4 <i>Analisi del panorama attuale</i>	57
1.4.1 Le tre ‘I’	62
1.5 <i>Le domande di ricerca</i>	69
2 La strategia di modellizzazione	75
2.1 <i>La base teorica per stabilire i requisiti di uno strumento di visualizzazione generico</i>	76
2.2 <i>Una strategia di modellizzazione invece di un unico modello</i>	79
2.3 <i>Una rete di modelli concettuali</i>	80
2.4 <i>I principi del Domain-Driven Design</i>	85
2.5 <i>La formalizzazione dei modelli concettuali</i>	87
2.5.1 Perché un'ontologia?	89
2.5.2 Il linguaggio di modellizzazione	91
2.5.3 I limiti di un'ontologia.....	92
2.5.4 Cenni sul linguaggio alla base di un'ontologia OWL 2.....	93

2.5.5	Ontologie attinenti il campo della filologia e della critica testuale.....	95
3	Il modello concettuale di esempio.....	105
3.1	<i>Il dominio e lo scopo del modello.....</i>	106
3.1.1	Una scelta controcorrente?.....	107
3.2	<i>Modalità di sviluppo dell'ontologia.....</i>	109
3.3	<i>Teoria di riferimento.....</i>	111
3.3.1	Edizioni di esempio.....	112
3.4	<i>Utilizzo delle ontologie esistenti.....</i>	114
3.5	<i>I concetti basilari.....</i>	116
3.5.1	Definizione di edizione critica.....	116
3.5.2	L'opera, la tradizione e il testimone.....	117
3.5.3	Formalizzazione del concetto di edizione critica.....	121
3.5.4	L'allestimento dell'edizione critica e l'editore.....	127
3.5.5	La pubblicazione dell'edizione critica e il lettore.....	130
3.5.6	Esempi di applicazione dei concetti basilari.....	133
3.6	<i>I componenti dell'edizione critica.....</i>	135
3.6.1	Definizione di "componente".....	136
3.6.2	La generalizzazione dei principali componenti di un'edizione critica.....	137
3.6.3	La descrizione della <i>restitutio textus</i>	141
3.6.4	La lezione.....	143
3.6.5	L'intervento critico: la descrizione della <i>restitutio textus</i> a livello microscopico.....	152
3.6.6	La descrizione delle lezioni in apparato.....	162
3.6.7	L'apparato critico.....	168
3.6.8	L'introduzione.....	191
4	Dal modello concettuale di esempio alla progettazione di uno strumento di visualizzazione.....	199
4.1	<i>Uno sviluppo object-oriented di strumenti di visualizzazione complessi.....</i>	200
4.2	<i>UML.....</i>	202
4.2.1	Cenni teorici su UML e la programmazione a oggetti.....	203
4.3	<i>La "traduzione" dell'ontologia in class diagram.....</i>	209

4.3.1	Scholarly Edition Component e i rapporti di parentela.....	211
4.3.2	Il rapporto “parte-intero”	214
4.3.3	Particolarità dell’apparato.....	219
4.3.4	La traduzione delle proprietà in attributo, funzionalità o qualità grafica	220
4.4	<i>La progettazione dello strumento informatico</i>	222
4.4.1	I compiti principali di uno strumento di visualizzazione.....	223
4.4.2	Il paradigma MVC	227
4.4.3	La realizzazione di implementazioni diverse	230
4.4.4	La gestione dell’interazione con i componenti grafici	233
4.4.5	La configurazione dell’interfaccia grafica	243
4.4.6	La suddivisione in moduli	245
4.4.7	Qualche spunto per il <i>layout</i>	246
Conclusioni		251
Bibliografia		255
Appendice A Strumenti informatici per la filologia digitale		271
Appendice B Definizioni inserite nell’ontologia		335
Appendice C Critical Edition Ontology		373

Indice delle figure

Figura 1.1 Schermata del software CTE.....	25
Figura 1.2 Grafico che mostra il numero di edizioni scientifiche digitali che sono basate su documenti in XML/TEI divise per anno in cui la loro realizzazione è stata avviata.....	38
Figura 1.3 Schema del funzionamento degli strumenti di produzione.....	41
Figura 1.4 Schema che rappresenta le difficoltà nella conversione tra i formati di output di strumenti informatici diversi.....	42
Figura 1.5 Schema del funzionamento degli strumenti misti.....	43
Figura 1.6 Schema del funzionamento degli strumenti di visualizzazione.....	44
Figura 1.7 Grafico che rappresenta il numero di edizioni scientifiche digitali realizzate ogni anno mediante l'utilizzo di uno strumento generale o generalizzato per la filologia digitale.....	60
Figura 2.1 Schema del rapporto tra le classi principali di LRMOO e CIDOC-CRM (LRMoo v. 0.9, p. 10).....	99
Figura 2.2 Schema delle ontologie sviluppate nell'infrastruttura NIE-INE (Cools e Padlina 2021).....	102
Figura 2.3 Architettura delle SPAR Ontologies.....	103
Figura 2.4 Il concetto di annotazione secondo il Web Annotation Data Model.....	104
Figura 3.1 Schema delle classi Work, Witness e Textual Tradition dell'ontologia di esempio.....	119
Figura 3.2 Schema delle classi che rappresentano la tradizione testuale di un'opera.....	121
Figura 3.3 Schema che rappresenta la classe se:Edition e le sue sottoclassi dirette.....	125
Figura 3.4 Schema della classe Critical Edition e delle sue superclassi.....	126
Figura 3.5 Schema del rapporto tra le classi Scholarly Edition e Work.....	126
Figura 3.6 Le principali classi e proprietà di LRMOO e i loro collegamenti alle classi di CIDOC CRM.....	129
Figura 3.7 Schema che rappresenta il rapporto tra le classi Editor, Editors Group e Scholarly Edition.....	130
Figura 3.8 Schema del rapporto tra la classe Scholarly Edition e Reader.....	133
Figura 3.9 Formalizzazione di un esempio di edizione a stampa attraverso i concetti basilari.....	134
Figura 3.10 Formalizzazione di un esempio di edizione digitale attraverso i concetti basilari.....	135
Figura 3.11 Schema dei principali componenti di un'edizione scientifica.....	138
Figura 3.12 Il rapporto tra Edited Text e Accompanying Material.....	141
Figura 3.13 Schema del rapporto tra Witness ed Edited Text.....	141
Figura 3.14 Schema del rapporto tra il testo critico e i testimoni.....	143

Figura 3.15 Grafo delle classi previste dalla CAO per descrivere il concetto di lezione (CAO).	145
Figura 3.16 Esempio di applicazione della CAO per descrivere due diverse lezioni.	146
Figura 3.17 Esempio di utilizzo della CAO per attribuire una lezione a un editore esterno..	147
Figura 3.18 Schema del rapporto tra Witness e Reading.....	149
Figura 3.19 Esempio di manoscritto danneggiato che presenta dei buchi.....	150
Figura 3.20 Schema delle classi Gap e Unclear Passage e del loro rapporto con Reading e Witness.	152
Figura 3.21 Incipit del Canzoniere di Petrarca nel codice Vat. lat. 3195.....	153
Figura 3.22 Esempio di come la regolarizzazione grafica di una lezione può essere formalizzata nel modello concettuale.....	154
Figura 3.23 Verso 4 del Canto I dell'Inferno nel codice Trivulziano 1080.	155
Figura 3.24 Verso 4 del Canto I dell'Inferno nel codice Urbinate latino 366.	155
Figura 3.25 Esempio di come formalizzare un intervento critico con il modello concettuale.	155
Figura 3.26 Esempio di come due diversi interventi critici sulla medesima lezione possono essere formalizzati con il modello concettuale.	156
Figura 3.27 Esempio di selectio formalizzato con le classi del modello concettuale.....	158
Figura 3.28 Schema delle classi per descrivere un intervento critico.....	159
Figura 3.29 Esempio dell'utilizzo della classe Critical Intervention Criterion.	159
Figura 3.30 Schermata della seconda edizione critica digitale della Commedia curata da Prue Shaw.	161
Figura 3.31 Esempio di come formalizzare una lacuna e la sua resa grafica nel testo critico..	162
Figura 3.32 Esempio di come si formalizza un'omissione sanata per congettura.	162
Figura 3.33 Schema di Reading In Apparatus e delle classi che ne rappresentano il tipo e la causa.	164
Figura 3.34 Esempio di istanza di Reading In Apparatus associata a un'istanza di Reading In Apparatus Type.	164
Figura 3.35 Schema del rapporto tra una lezione in apparato e la mano scribale cui è attribuita.	167
Figura 3.36 Schema delle proprietà riflessive di Reading In Apparatus.	168
Figura 3.37 Esempio di come formalizzare il rapporto tra due lezioni in apparato correlate.	168
Figura 3.38 Schema della prima soluzione per rappresentare l'apparato critico.....	170
Figura 3.39 Schema della seconda soluzione per rappresentare l'apparato critico.	171
Figura 3.40 Schema delle diverse tipologie di apparato e delle rispettive voci.	172
Figura 3.41 Schermata dell'interfaccia del software EVT 2.....	173
Figura 3.42 Schema del rapporto tra le classi Edited Text e Apparatus.	174

Figura 3.43 Schema del rapporto tra l’annotazione, l’unità di variazione e il luogo di variazione nell’ontologia CAO (CAO).....	175
Figura 3.44 Schema delle classi utilizzate per rappresentate le voci d’apparato e le rispettive sottoclassi.....	177
Figura 3.45 Schema che mostra come utilizzare la classe oa:Text Quote Selector (Web Annotation Vocabulary).....	179
Figura 3.46 Schema che rappresenta le diverse tipologie di selettori di un luogo del testo edito e il rapporto con il contenuto di una voce d’apparato.....	181
Figura 3.47 Schema delle classi e delle proprietà che rappresentano il commento filologico.....	182
Figura 3.48 Schema delle classi e delle proprietà che descrivono il concetto di fascia dell’apparato critico.....	183
Figura 3.49 Esempio di come formalizzare tre diverse fasce dell’apparato critico.....	184
Figura 3.50 Schema del rapporto tra Reading In Apparatus e Base Reading In Apparatus.....	185
Figura 3.51 Schema delle proprietà che associano una lezione in apparato alla lezione accolta a testo mostrata nell’apparato critico.....	186
Figura 3.52 Schema del rapporto tra la lezione in apparato e il rimando alla sigla del testimone.....	187
Figura 3.53 Schema del rapporto tra le classi cao:Base Reading, cao:Reading e cao:Variation Unit.....	188
Figura 3.54 Schema delle classi e delle proprietà disponibili per rappresentare un gruppo di lezioni.....	190
Figura 3.55 Esempio di come formalizzare il rapporto tra Reading e Reading In Apparatus.....	190
Figura 3.56 Schema della classe Critical Apparatus Entry Content Note.....	191
Figura 3.57 Schema delle classi per descrivere il concetto di introduzione e delle sue parti...	192
Figura 3.58 Schema delle classi e delle proprietà disponibili per formalizzare l’elenco dei testimoni.....	194
Figura 3.59 Esempio di cladogramma tratto dalla seconda edizione digitale della Commedia curata da Prue Shaw.....	195
Figura 3.60 Schema dei rapporti tra le classi Witnesses List, Stemma Codicum e Introduction.....	196
Figura 4.1 Class diagram delle classi “Alunno” e “Professore”.....	2044
Figura 4.2 Class diagram della classe “Alunno” e del suo oggetto “Mario Rossi”.....	2055
Figura 4.3 Class diagram della classe “Alunno” con i relativi metodi e dell’oggetto “Mario Rossi”.....	2055
Figura 4.4 Class diagram completo delle classi “Alunno” e “Professore”.....	2066
Figura 4.5 Activity diagram dell’operazione “metti voto”.....	2077

Figura 4.6 Tassonomia dei diagrammi strutturali e comportamentali (UML v. 2.5.1, p. 685).	2088
Figura 4.7 Schema che rappresenta la parte dell'ontologia di esempio che descrive il rapporto tra l'edizione scientifica e il lettore.	210
Figura 4.8 Diagramma UML generato con OWLGrEd della parte dell'ontologia di esempio che descrive il rapporto tra l'edizione scientifica e il lettore.....	21010
Figura 4.9 Class diagram dei principali componenti alla base di un'edizione scientifica digitale.	2144
Figura 4.10 Class diagram che rappresenta il rapporto di composizione tra il testo edito e un passo del testo edito così come tra le loro rispettive sottoclassi.....	2166
Figura 4.11 Class diagram che rappresenta i rapporti tra l'introduzione, le sue parti, la lista dei testimoni e lo stemma codicum.....	2177
Figura 4.12 Class diagram che rappresenta il rapporto di aggregazione tra un apparato e una voce d'apparato così come tra le rispettive sottoclassi.	2188
Figura 4.13 Class diagram che descrive i rapporti di composizioni tra gli elementi che possono strutturare l'apparato critico e una sua voce.	2188
Figura 4.14 Class diagram che indica i rapporti tra una voce d'apparato e il luogo nel testo cui è riferita.....	2199
Figura 4.15 Class diagram che illustra i rapporti tra le lezioni presentate all'interno di una voce dell'apparato critico e le relative strutture.....	2199
Figura 4.16 Class diagram della classe Reading In Apparatus e dei relativi attributi.	220
Figura 4.17 Class diagram della classe Siglum Reference.	22121
Figura 4.18 Diagramma UML dell'interfaccia Scholarly Edition Component e del metodo retrieve.	2244
Figura 4.19 Diagramma UML dell'interfaccia Scholarly Edition Component con i metodi retrieve e format.	2277
Figura 4.20 Schema del pattern MVC.....	2288
Figura 4.21 Esempio di due "view" differenti.....	2299
Figura 4.22 Schema che presenta gli elementi alla base del design pattern "Bridge" (Gamma et al. 1994).	2322
Figura 4.23 Schermata dell'interfaccia grafica dello strumento di visualizzazione Versioning Machine.....	2344
Figura 4.24 Class diagram di un controller per la classe Scholarly Edition Component.	2355
Figura 4.25 Schermata dell'interfaccia grafica della Faust Edition.....	2377
Figura 4.26 Schermata dell'apparato critico e del testo critico all'interno della Faust Edition.	2377
Figura 4.27 Schermata di un verso del testo critico e delle rispettive trascrizioni dei testimoni all'interno della Faust Edition.	2388

Figura 4.28 Esempio di selettore separato dal passo del testo critico.	2388
Figura 4.29 Class diagram dei controller per l'apparato, la voce d'apparato, il passo del testo edito e il selettore del passo.	24040
Figura 4.30 Class diagram delle classi Reading In Apparatus e Siglum Reference.....	24141
Figura 4.31 Funzionalità di filtro delle varianti offerta nel software EVT 2.....	2422
Figura 4.32 Class diagram della funzionalità di filtro delle varianti.	2433

Introduzione

La ricerca presentata in questo elaborato si inserisce nell'ambito della filologia digitale e più precisamente della realizzazione di edizioni scientifiche digitali (ESD). Questo termine, calco dell'inglese *digital scholarly edition*, indica un'edizione fruibile in ambiente digitale e realizzata da uno studioso, attraverso strumenti digitali, applicando delle metodologie rigorose e rendendo verificabile il proprio lavoro.

Una ESD può offrire molteplici vantaggi agli studiosi, tra cui:

- raggiungere un pubblico di lettori più ampio, pubblicando sul web in *open access*;
- integrare più facilmente all'interno delle edizioni critiche un maggior numero di contenuti (riproduzioni fotografiche, trascrizioni, ecc.) prodotti in proprio, oppure riutilizzare o creare riferimenti ad altri materiali già presenti in rete (per esempio le riproduzioni dei manoscritti servendosi del framework IIIF);
- intervenire più agevolmente sulla propria edizione in un secondo momento per aggiungere materiale o rivederne l'impostazione alla luce di nuove acquisizioni;
- lavorare alle edizioni collaborativamente;
- mettere a disposizione della comunità scientifica i propri dati in un formato *machine-readable*, e quindi ulteriormente elaborabile anche con metodi computazionali.

Negli ultimi trent'anni sono state realizzate svariate ESD e, ad oggi, la metodologia maggiormente applicata per realizzare una ESD è detta "sorgente-*output*" (Mancinelli e Pierazzo 2020), che prevede l'adozione dello standard TEI e la pubblicazione del file (o più file) sorgente in XML/TEI mediante uno strumento informatico dedicato.

"The typical approach to the digital scholarly editing generally encompasses two phases: 1) textual encoding and 2) web publishing. [...] the aim of the electronic publication phase is to provide end-users (mainly scholars) with high-level functionalities (e.g. advanced visualization and searching) through Web graphical user interfaces." (Del Grosso et al. 2017)

Nonostante siano attualmente disponibili diversi strumenti di pubblicazione (o visualizzazione) generici, cioè ideati per adattarsi a edizioni diverse, la maggior parte delle ESD viene pubblicata nel web attraverso strumenti di visualizzazione realizzati *ad hoc*, prediligendo una modalità di *editing*

“haute couture”, invece, che “prêt-à-porter” (Pierazzo 2019). Si tratta molto spesso di sistemi informatici sofisticati che offrono delle funzionalità e delle modalità di consultazione del testo innovative. Di contro, sono sistemi costosi e complessi sia da realizzare sia da mantenere. Per questo motivo, accade di frequente che una ESD diventi inaccessibile una volta terminati i fondi attraverso i quali la sua realizzazione era stata finanziata. Gli strumenti generici di visualizzazione generici faticano a imporsi in quanto, in alcuni casi, sono incompleti o inadatti a conseguire specifici obiettivi degli studiosi. Gli strumenti generici, infatti, “agiscono come collane editoriali, vale a dire che offrono una formula comune con minimi margini di flessibilità” (Mancinelli e Pierazzo 2020). La difficoltà di adeguare uno strumento generico alle esigenze scientifiche di una particolare ESD e le limitazioni nel poterlo personalizzare, portano, quindi, gli studiosi ad avere una certa riluttanza a servirsi di strumenti generici e, in generale, a nutrire una certa sfiducia nell’efficacia di questi strumenti.

Da una decina di anni la comunità di studiosi si interroga su come mai gli strumenti generici non riescano soddisfare le esigenze tecniche e scientifiche delle ESD. Una delle principali cause individuate è che mancano dei principi o modelli che indichino concretamente come gli strumenti informatici per la filologia digitale debbano essere sviluppati. È un problema che non riguarda solo gli strumenti di visualizzazione (o pubblicazione), ma anche i *tool* ideati per condurre altre attività tipiche della critica testuale come la trascrizione delle fonti primarie, la lemmatizzazione, ecc. Per quanto riguarda gli strumenti di visualizzazione generici, il motivo per cui faticano a essere utilizzati per ESD diverse è che molto spesso sono stati progettati sulla base di uno specifico progetto di edizione, invece che su un modello editoriale condiviso e stabilito dalla comunità scientifica:

“What most tools have in common is the fact [that] their development seems to have been generated from data models (how a file is encoded), from specific projects, and from specific requests made by early users, more than from an effort to assess the editorial models needed by the scholarly community” (Pierazzo 2019).

La soluzione a questo problema è, dunque, che la comunità scientifica produca collettivamente dei modelli che tengano conto delle peculiarità e delle potenzialità dell’ambiente digitale:

“In order to be able to offer a stable infrastructure for digital editions it is therefore necessary to reflect on the features that the scholarly community will consider essential to a particular type of text or scholarly problem and to agree on some essential models which take into account the new affordances offered by the digital” (Pierazzo 2019).

Questi modelli, però, non devono rivolgersi esclusivamente agli studiosi, ma devono essere utilizzabili prima di tutto dagli sviluppatori informatici. L'obiettivo della mia ricerca è quello di elaborare una strategia di modellizzazione, ovvero capire concretamente come questi modelli possano essere prodotti al duplice fine di guidare lo sviluppo dei software di visualizzazione e di rappresentare i *desiderata* della comunità dei filologi. Per questo motivo, il presente elaborato si rivolge sia agli sviluppatori informatici che lavorano nel campo della filologia digitale, sia agli studiosi che realizzano o che desiderano realizzare edizioni scientifiche digitali.

Il presente lavoro di ricerca si fonda su un'analisi degli strumenti generici per la filologia digitale, per concentrarsi poi sui software di visualizzazione. Nel primo capitolo, dopo alcune premesse teoriche che comprendono la definizione di "strumento informatico per la filologia digitale" e una breve rassegna della storia del rapporto tra filologia e informatica, si propone una possibile classificazione degli strumenti per la filologia digitale. Per ogni categoria proposta si offrono alcuni esempi di strumento ritenuti maggiormente rappresentativi. Il primo capitolo procede con un'analisi delle tendenze nello sviluppo di software per la filologia digitale e degli aspetti problematici che rendono questi strumenti poco usabili, ponendo il *focus* sugli strumenti di visualizzazione. Gli strumenti analizzati sono presentati nel dettaglio nell'appendice A. Il primo capitolo si chiude con le domande che hanno guidato la presente ricerca e che hanno portato all'elaborazione di una strategia di modellizzazione.

Nel secondo capitolo viene definita la strategia di modellizzazione, mediante la quale si intende realizzare dei "modelli editoriali" per guidare lo sviluppo di strumenti informatici sofisticati e innovativi per la visualizzazione di ESD. La strategia prevede che i modelli editoriali siano sviluppati come modelli concettuali o "di dominio" e che, a partire da questi modelli, sia possibile progettare uno strumento di visualizzazione applicando i principi del *domain-driven design*. Il secondo capitolo procede con l'analisi dei linguaggi di modellizzazione che meglio si prestano alla formalizzazione di un modello editoriale, illustrando come un'ontologia sia la scelta più opportuna. Il capitolo si chiude con una rassegna delle ontologie esistenti che pertengono al campo della filologia e della critica testuale, da utilizzare come riferimento per sviluppare una rete di modelli editoriali.

Il terzo capitolo è dedicato al modello concettuale di esempio che descrive un'edizione critica di approccio neo-lachmanniano. L'ontologia di esempio è suddivisa in tre parti principali, di cui la

prima che introduce i concetti alla base di un'edizione critica, come per esempio “testimone”, “opera”, ecc. La seconda parte descrive il rapporto tra il testo critico e i testimoni, offrendo un insieme di classi e proprietà per rendere trasparente il lavoro ricostruttivo dell'editore e i singoli interventi effettuati sul testo. I concetti descritti in questa parte servono come base per rappresentare in modo dettagliato i contenuti dell'apparato critico e il testo critico. La terza e ultima parte descrive i diversi componenti di un'edizione critica, dall'apparato critico, all'introduzione, alle note di commento filologico. Quest'ultima parte è il punto di partenza per la progettazione dello strumento informatico, mentre le altre due parti sono necessarie per inquadrare correttamente i concetti della terza. La descrizione del modello concettuale di esempio è introdotta da una riflessione sul perché è stata scelta questa particolare tipologia di edizione e dalla presentazione di come il lavoro di sviluppo dell'ontologia è stato svolto. Le definizioni delle classi e delle proprietà native dell'ontologia di esempio sono presentate nell'appendice B, mentre l'ontologia nella sua interezza è consultabile nell'appendice C.

Il quarto e ultimo capitolo illustra come un modello concettuale possa essere *in primis* tradotto nel linguaggio di modellizzazione UML, linguaggio largamente adottato per la progettazione di software, e poi applicato per la progettazione di uno strumento informatico di visualizzazione, utilizzando l'ontologia di esempio. L'elaborato si chiude con un resoconto dei risultati ottenuti e la descrizione dei prossimi passi necessari per consolidare la strategia di modellizzazione e per i possibili sviluppi futuri.

Capitolo 1

Strumenti informatici per la filologia digitale: stato dell'arte e domande di ricerca

La ricerca che ho condotto e che presento in questo elaborato nasce da un'analisi degli strumenti informatici per la filologia digitale attualmente disponibili. L'analisi si concentra in particolare sugli “strumenti di visualizzazione”, ovvero i prodotti software concepiti per visualizzare e pubblicare un'edizione scientifica digitale,¹ realizzata secondo il paradigma “sorgente-output” (Mancinelli e Pierazzo. 2020). Ciò che emerge dall'analisi è che il panorama attuale è piuttosto frammentato. Negli ultimi anni sono stati creati diversi strumenti di visualizzazione generici, ovvero ideati per essere utilizzati in più progetti di filologia digitale. La loro adozione, tuttavia, rimane limitata, in quanto la comunità scientifica li percepisce come poco usabili e i singoli progetti tendono a sviluppare (o far sviluppare) i propri sistemi di visualizzazione *ad hoc*. Questa tendenza è problematica in termini sia di sostenibilità dello sviluppo software sia di solidità e stabilità delle edizioni scientifiche digitali.

L'esito dell'analisi mi ha spinto a interrogarmi sulla ragione della limitatezza degli strumenti di visualizzazione, individuando problematiche comuni anche alle altre tipologie di strumenti per la filologia digitale. Tali problematiche dipendono, da un lato, dalle modalità e dai contesti in cui questi strumenti vengono prodotti e, dall'altro, dalla difficoltà della comunità scientifica a individuare dei requisiti comuni per gli strumenti informatici.

Nella letteratura scientifica altri studiosi hanno sottolineato la necessità di elaborare dei principi e dei modelli condivisi che guidino chiaramente lo sviluppo di strumenti informatici per la filologia digitale. L'obiettivo della mia ricerca è quello di capire come dei modelli condivisi possano esprimere

¹ Come anticipato nell'introduzione, in questo elaborato il termine “edizione scientifica digitale” è utilizzato per indicare in generale un'edizione fruibile in ambiente digitale, curata da uno studioso applicando delle metodologie rigorose e rendendo verificabile il proprio lavoro.

i *desiderata* tecnici e scientifici per visualizzare un'edizione scientifica digitale e di studiare concretamente come tali modelli possano essere realizzati, affinché risultino utili agli sviluppatori informatici e direttamente applicabili nell'implementazione di tali strumenti.

Nella prima parte del capitolo si propongono delle definizioni di base e dei cenni storici sul rapporto tra la filologia e l'informatica per inquadrare l'oggetto dell'analisi, cioè gli strumenti informatici per la filologia digitale. Nella seconda parte vengono illustrate le diverse categorie di strumenti individuate con dei riferimenti a esempi concreti. Infine, si conduce un'analisi degli strumenti per la filologia digitale e vengono presentate le domande che hanno portato al presente studio.

1.1 Definizione di “strumento informatico per la filologia digitale”

Per inquadrare che cosa si intende con “strumento informatico per la filologia digitale”, l'oggetto dell'analisi condotta nel presente capitolo, conviene dividere la definizione a metà: da un lato “strumento informatico” e dall'altro “per la filologia digitale”. Partiamo dall'espressione “strumento informatico”, con la quale si fa riferimento prevalentemente ai concetti di software applicativo o solamente applicativo che, nel *Systems and Software Engineering Vocabulary*, standard ISO² elaborato dall'Institute of Electrical and Electronics Engineers (IEEE) nel 2017, sono così definiti:

“**software** 1. computer programs,³ procedures and possibly associated documentation and data pertaining to the operation of a computer system”.

“**application software** 1. software designed to help users perform particular tasks or handle particular type of problems, as distinct from software that controls the computer itself; 2. software or a program that is specific to the solution of an application problem; 3. software designed to fulfill specific needs of a user.”

Semplificando si può dire che, mentre con software si intende un programma o un insieme di programmi necessari per il funzionamento del computer stesso (la parte “soft”, cioè facilmente modificabile, del computer contrapposta all'hardware), un software applicativo è un programma o un

² Cfr. la voce “Software” dell'Enciclopedia Treccani (<https://www.treccani.it/enciclopedia/software>). Cfr. la voce “Applicazione” dell'Enciclopedia Treccani (<https://www.treccani.it/enciclopedia/applicazione>).

³ Un programma (*computer program*) è una sequenza di istruzioni e di definizioni di dati impartite a un elaboratore elettronico, affinché esegua delle funzioni di calcolo o di controllo. Per ulteriori chiarimenti si rimanda alla voce “Programma” nell'Enciclopedia della Matematica Treccani (https://www.treccani.it/enciclopedia/programma_%28Enciclopedia-della-Matematica%29/).

insieme di programmi che assolve a dei problemi specifici di un determinato dominio applicativo,⁴ quindi permette all'utente finale di svolgere dei compiti precisi. Per esempio, Word è un software applicativo per la videoscrittura, che permette all'utente di scrivere e redigere documenti al computer.

La scelta dell'etichetta “strumento informatico” invece di “applicativo” è dovuta al fatto che oggi, in seguito alla diffusione del web e al consolidamento di nuovi tipi di architetture informatiche (in particolare i servizi), molti prodotti software non vengono più distribuiti come applicativi che devono essere scaricati e installati su un computer. Molti sono accessibili dal web e utilizzabili direttamente all'interno del proprio browser. Altri, invece, sono pensati per essere utilizzati da altri strumenti informatici e sono accessibili mediante API. L'etichetta “strumento informatico” comprende tutte queste tipologie di prodotti software, mettendo al contempo l'enfasi sul fatto che si tratta di prodotti al servizio dell'utente e adibiti a un particolare compito.

Passiamo ora alla seconda parte della definizione: “per la filologia digitale”. La preposizione “per” indica semplicemente che si tratta di uno strumento applicato nel campo della filologia digitale. La definizione del concetto di “filologia digitale” richiede di essere inquadrata in una riflessione più ampia sulla contrapposizione tra “humanities computing” e “digital humanities” (Buzzetti 2017), che però fuoriesce dagli scopi del presente elaborato. Si riporta però la contrapposizione tra “filologia digitalizzata” e “filologia digitale” (Mancinelli e Pierazzo 2020, Sahle 2016), dove la prima è intesa come l'utilizzo di strumenti digitali per realizzare edizioni scientifiche secondo le metodologie tradizionali della filologia, mentre la seconda indica la realizzazione di edizioni scientifiche digitali applicando metodologie nate in ambiente digitale. In questa rassegna, dato che la prospettiva è posta sugli strumenti, invece che sulle metodologie in sé, con “filologia digitale” si intende sia per la “filologia digitale” sia per quella “digitalizzata”.

I filologi oggi dispongono di un elevato numero di strumenti informatici, di cui solo alcuni sono stati ideati appositamente per essere applicati nel campo della filologia e della critica testuale, mentre

⁴ La definizione di dominio fornita nel *Systems and Software Engineering Vocabulary* è la seguente: “**domain** 1. distinct scope, within which common characteristics are exhibited, common rules observed, and over which a distribution transparency is preserved; 2. problem space; 3. area of knowledge or activity characterized by a set of concepts and terminology understood by practitioners in that area”. In generale, si può definire “dominio” di un applicativo l'ambito reale in cui viene applicato, per esempio l'ambito commerciale, della salute pubblica, delle telecomunicazioni, ecc.

la maggior parte di essi è uno strumento generalista (*general purpose*).⁵ Si pensi a Word di Microsoft o altri *word processor*⁶ analoghi, che sono diventati insieme al *personal computer* uno strumento di lavoro imprescindibile per molte figure professionali diverse, tra cui anche i filologi.

Un possibile esempio è costituito dal portale TAPoR (Text Analysis Portal for Research),⁷ collezione di strumenti per umanisti digitali liberamente accessibile online e sviluppata presso l'Università di Alberta. Inizialmente focalizzato sugli strumenti per l'analisi dei testi (come suggerisce il nome stesso), TAPoR è stato ampliato in un secondo momento, fino a comprendere strumenti decisamente generalisti quali, per esempio, Wordpress,⁸ il celebre *content management system* per la realizzazione di siti web, e Slack,⁹ un'applicazione di messaggistica istantanea studiata per chi lavora in un gruppo di ricerca.

La domanda sorge spontanea: in base a quali criteri si possono distinguere gli strumenti informatici generalisti, rivolti cioè a un pubblico generico, da quelli “per la filologia digitale”? Il distinguo tra le due tipologie in alcuni strumenti è sfumato, ma è possibile servirsi di due criteri. Un criterio è che si tratti di strumenti concepiti appositamente per assolvere a un compito riconosciuto dalla comunità scientifica come parte delle pratiche della filologia digitale. Questo tipo di strumenti è spesso realizzato da studiosi di discipline umanistiche che possiedono anche competenze di programmazione software, oppure da informatici sotto il loro diretto controllo. Come esempio di strumento individuabile secondo questo primo criterio e creato da questo tipo di studioso anfibio, si riporta l'applicativo Classical Text Editor (CTE)*, realizzato dal classicista Stefan Hagel nel 1997 e tuttora utilizzabile.¹⁰

⁵ In ambito informatico, con l'espressione *general purpose* si intende un applicativo versatile e adattabile a diversi impieghi. L'espressione si contrappone a *special purpose*, che al contrario indica applicativi realizzati per un compito specifico.

⁶ Con *word processor* si intende un tipo di applicativo software specializzato nella videoscrittura.

⁷ Il portale è consultabile all'indirizzo: <https://tapor.ca/home>.

⁸ Sito web di Wordpress: <https://wordpress.com/it/>.

⁹ Sito web di Slack: <https://slack.com/intl/it-it>.

¹⁰ Per i software menzionati nel presente capitolo che sono elencati anche nell'Appendice A non si fornisce in nota l'indirizzo del sito web, in quanto è già disponibile nella suddetta appendice.

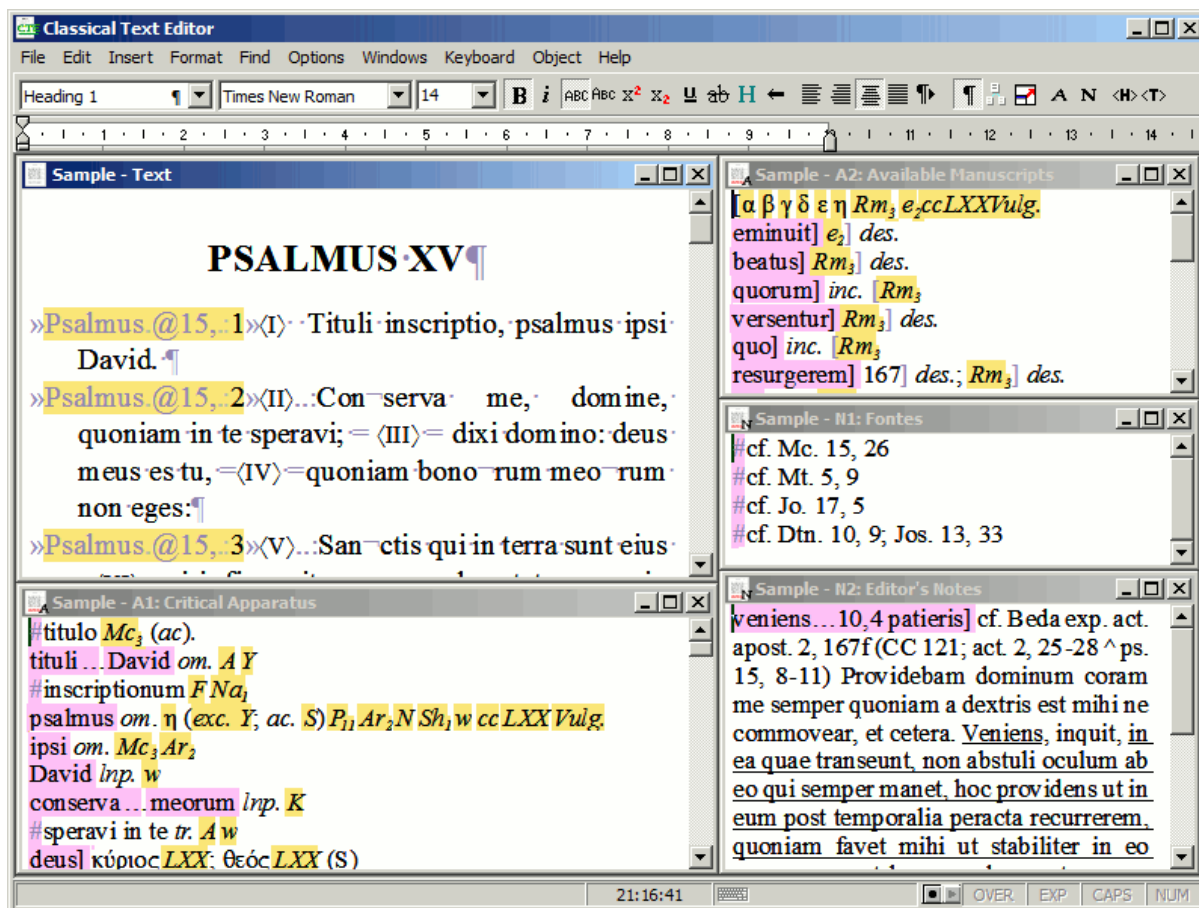


Figura 1.1 Schermata del software CTE.

Come si osserva nella figura 1.1, CTE offre diverse funzionalità comuni a tutti i *word processor*, come per esempio la formattazione del testo in grassetto. Essendo stato pensato, però, per realizzare edizioni critiche, consente anche di: creare apparati critici molto articolati e complessi; creare le sigle per i testimoni, singoli o per gruppi; mettere a confronto il testo con l'immagine del manoscritto; esportare il risultato in formato per la stampa o per la pubblicazione elettronica (CD-ROM, Web), sia in HTML sia secondo lo standard TEI.

L'altro criterio per individuare gli strumenti "per la filologia digitale" è che, pur non essendo stati ideati appositamente per essere applicati nel campo della critica del testo, si siano rivelati essere particolarmente efficaci per assolvere ad alcuni compiti e di conseguenza siano stati largamente adottati dalla comunità scientifica. Si riporta come esempio Oxygen*, *editor* generalista per documenti in XML, nonché uno degli strumenti più utilizzati tra gli studiosi che realizzano edizioni digitali secondo lo standard TEI.

I due criteri sopra enunciati richiedono che a monte siano messe a fuoco quali sono le attività della critica testuale. Per esempio, il software Oxygen può essere considerato uno “strumento informatico per la filologia digitale” se la marcatura di un testo in XML è considerata un’attività della pratica ecdotica in ambiente digitale. Stabilire quali siano esattamente queste attività è, però, problematico, in quanto le attività svolte da un filologo per realizzare un’edizione scientifica dipendono dalla metodologia adottata e dall’ambiente in cui il filologo lavora.

Per determinare le diverse attività di ricerca che possono essere svolte mediante uno strumento informatico nel campo delle *digital humanities*, e quindi anche della filologia digitale, si può fare riferimento alla Taxonomy of Digital Research Activities in the Humanities, abbreviato TaDiRAH.¹¹ Questa tassonomia, usata tra l’altro nel già menzionato portale TAPoR per la classificazione degli strumenti, fornisce la terminologia per descrivere le attività insieme alle metodologie a esse associate, ed è stata progettata proprio per aiutare a raccogliere e classificare informazioni sugli strumenti, i metodi e i progetti delle DH. Le principali attività previste da TaDiRAH sono: “analyzing” (analisi), “capturing” (acquisizione), “creating” (creazione), “disseminating” (disseminazione), “enriching” (arricchimento), “interpreting” (interpretazione) e “storing” (conservazione). Le attività sono organizzate gerarchicamente e ciascuna delle attività principali è suddivisa in sotto-attività, a loro volta suddivise in ulteriori sotto-attività, per esempio l’attività “enriching” comprende la sotto-attività “editing”, che a sua volta comprende le attività “adding”, “correcting”, “merging” ecc.

Riprendendo i due applicativi presentati come esempi, CTE e Oxygen, possiamo dire che, secondo la tassonomia TaDiRAH, il primo consente di effettuare l’attività etichettabile come “editing”¹² o più precisamente di “formatting”,¹³ mentre il secondo sia l’attività di “editing” che di “tagging”.¹⁴

¹¹ Sito web ufficiale di TaDiRAH: <https://tadirah.info/>. Sito web contenente la tassonomia completa: <https://vocabs.dariah.eu/tadirah/en/>.

¹² Cfr. con la definizione di “editing” all’interno di TaDiRAH: “editing refers to adding, correcting, condensating, organizing or in other form modifying information of a specific object with the intention to complete and finalize it by making structural, layout-related, semantic, or other scholarly explicit inline or stand-off markup.” (<https://vocabs.dariah.eu/tadirah/editing>).

¹³ Definizione di “formatting” all’interno di TaDiRAH: <https://vocabs.dariah.eu/tadirah/formatting>.

¹⁴ Definizione di “tagging” all’interno di TaDiRAH: <https://vocabs.dariah.eu/tadirah/tagging>.

Le attività definite nella tassonomia TaDiRAH potrebbero risultare poco chiare a una parte dei lettori cui questo elaborato si rivolge, ovvero filologi che non conoscono o hanno poca dimestichezza con le metodologie proprie della filologia digitale, ma che desiderano servirsi di strumenti informatici per la filologia digitale. Per categorizzare questi strumenti pare opportuno, quindi, fare riferimento alle diverse fasi di lavoro per la realizzazione di un'edizione critica che tradizionalmente vengono insegnate nei corsi di filologia nell'ambito accademico italiano. Queste fasi sono state sintetizzate da Leonardi (2021) in un articolo in cui presenta una proposta per “un'infrastruttura di filologia digitale”, a partire da fasi decisamente consolidate della critica testuale, tra cui la *collatio*, la *recensio* e la *constitutio textus*, delineate già da Paul Maas nel volume *Textkritik* (1927). Nell'articolo, Leonardi sottolinea come le diverse fasi possano essere condotte sia con metodologie “tradizionali” sia applicando nuove tecnologie (per esempio software HTR¹⁵ o programmi di collazione automatica) e, nel dettaglio, sono:

- il censimento dei testimoni;
- la trascrizione (lemmatizzata o meno) dei testimoni;
- la collazione, che può essere condotta sul testo integralmente o solo su alcune parti selezionate in quanto particolarmente rappresentative delle peculiarità della tradizione (*loci critici*);
- la classificazione dei testimoni, sulla base dello studio della tradizione, che può portare eventualmente alla costituzione di uno *stemma codicum*;
- infine, l'allestimento vero e proprio dell'edizione.

Le fasi sopra elencate, per quanto fortemente radicate nel metodo (neo-)lachmanniano o stemmatico, prese singolarmente permettono di individuare attività di ricerca, o meglio, degli obiettivi comuni a diverse metodologie filologiche, digitali e non. In altre parole, le attività di ricerca attuate con l'ausilio di uno strumento informatico, per quanto possano differire tra loro, possono essere comunque inquadrare negli obiettivi del lavoro ecdotico tradizionale. Obiettivi che, a un livello più astratto, rientrano perfettamente nelle attività principali definite dalla tassonomia TaDiRAH: ana-

¹⁵ Acronimo di *Handwritten Text Recognition* (“riconoscimento di testo manoscritto”) usato per indicare i software che permettono al computer di leggere e interpretare correttamente del testo scritto a mano, ricavandolo a partire da immagini o fotografie di documenti.

lisi, acquisizione, creazione, disseminazione, arricchimento, interpretazione e archiviazione. Le effettive attività di ricerca variano nel tempo, in base ai formati dei dati e alle tecnologie che via via si impongono. Tuttavia, è realistico credere che gli obiettivi della pratica filologica rimarranno concettualmente gli stessi, anche se di volta in volta saranno proposte nuove soluzioni, implementate con metodologie nuove. Si pensi banalmente alla diffusione del web semantico e dei *linked open data*. Anche in ambito filologico sono state condotte delle sperimentazioni che hanno portato alla pubblicazione di edizioni digitali in cui i dati sono disponibili in uno o più formati standard del web semantico.¹⁶ La pubblicazione dei dati in questi formati particolari avrà sicuramente richiesto l'applicazione di strumenti informatici congrui, tuttavia, a livello concettuale, l'attività svolta rientra nella fase della pubblicazione e disseminazione del testo edito.

Ricapitolando, con “strumento informatico per la filologia digitale” si intende un applicativo o un programma ideato per essere applicato nel campo della filologia digitale o largamente adottato dalla comunità scientifica, che, assistendo o sostituendo il filologo, permette di effettuare una o più attività finalizzate alla realizzazione di un'edizione scientifica digitale.

1.2 Cenni storici sul rapporto tra la filologia e lo sviluppo di strumenti informatici

L'obiettivo di questa sezione è quello di fornire una prospettiva storica per inquadrare il contesto in cui sono nati gli strumenti utilizzati oggi. Gli strumenti attualmente disponibili saranno presentati in modo più dettagliato ed esteso nella successiva sezione.

Tracciare una storia del rapporto tra strumenti informatici e le pratiche della critica del testo non è compito banale, perché richiede di tenere conto dei cambiamenti che negli anni sono occorsi in tanti campi diversi intrecciati tra di loro. Da un lato c'è lo sviluppo del software, dell'hardware e delle telecomunicazioni, dall'altro lato ci sono la filologia, la critica testuale e le loro applicazioni mediante le tecnologie digitali.

¹⁶ Un esempio di edizione di questo tipo è quella delle lettere di Vespasiano da Bisticci, curata da Francesca Tomasi e realizzata presso l'Università di Bologna (<http://projects.dharc.unibo.it/vespasiano/>).

Nel volume *Che cos'è un'edizione scientifica digitale?* (2020) Mancinelli e Pierazzo individuano quattro fasi principali che hanno contraddistinto la storia della filologia e delle edizioni digitali (p. 32), di cui le prime tre sono state mutate dall'analisi di Tito Orlandi in *Informatica testuale: teoria e prassi* (2010, pp. 82-90):

1. la sperimentazione;
2. il consolidamento;
3. l'inizio del "www";
4. l'interoperabilità.

Le fasi sopra elencate non sono definite da spartiacque nitidi, ma sono state delineate tenendo conto degli avvenimenti più significativi e delle principali tendenze che hanno contraddistinto un determinato momento storico. In questa sezione sono ripresi gli strumenti e le tendenze descritte da Mancinelli e Pierazzo (2020) ritenuti più interessanti rispetto ai fini del presente lavoro di ricerca, mantenendo la suddivisione in quattro fasi da loro proposta.¹⁷ Per chiarire gli aspetti tecnici che hanno contraddistinto le diverse fasi, in una sottosezione dedicata sono forniti dei riferimenti ai cambiamenti più significativi avvenuti nel campo del software, dell'hardware e delle telecomunicazioni.

1.2.1 Quattro fasi

La prima fase si avvia negli anni Quaranta con le prime sperimentazioni dell'uso degli elaboratori elettronici in campo umanistico. Molto noto è l'*Index Thomisticus* di padre Roberto Busa, che grazie a una collaborazione con IBM Italia nel 1949 avviò la lemmatizzazione mediante computer e schede perforate dell'opera di Tommaso d'Aquino, segnando l'inizio della linguistica computazionale. Passando alla filologia computazionale, in quegli anni gli elaboratori elettronici furono utilizzati prevalentemente per automatizzare due operazioni considerate facilmente meccanizzabili: *recensio* e *collatio*. Tra i più noti esperimenti condotti in questo periodo, si ricordano il lavoro di Charles Hinman e del monaco benedettino Jacques Froger. Hinman studiò le varianti di stato delle edizioni a

¹⁷ Le prime tre fasi presentate sia in Mancinelli e Pierazzo 2020 sia in Orlandi 2010 sono state tratte dal saggio *History of Humanities Computing* di Susan Hockey (2004). Nel presente elaborato si è ritenuto opportuno fare riferimento prevalentemente a Mancinelli e Pierazzo 2020 sia perché si tratta della pubblicazione più recente, che individua appunto anche una quarta fase, sia perché il *focus* è posto sul campo della filologia digitale, invece che sull'informatica umanistica *tout court*.

stampa delle opere di Shakespeare mediante un sistema di collazione ottica. A partire dagli anni Cinquanta continuò i propri studi realizzando delle collazioni automatiche mediante il computer. Jacques Froger, che nel 1968 pubblicò il libro *La Critique des textes et son automatisaton*, mise a punto il metodo di collazione automatica elaborato da Henri Quentin, basato su un approccio “quantitativo”. Quest’ultimo, dovendo realizzare l’edizione critica della Vulgata della Bibbia, contraddistinta da una tradizione testuale di notevoli dimensioni, scelse di non distinguere qualitativamente tra lezioni “buone” e lezioni “erronee”, ma di affidarsi a un approccio puramente quantitativo, considerando che anche l’archetipo o l’originale possono contenere lezioni “erronee”. Questa prima fase si chiude simbolicamente con il congresso dei filologi computazionali che si tenne a Parigi nel 1979, decretando una prima crisi dell’applicazione di metodi formali all’ecdotica. In quell’occasione, infatti, furono constatati i limiti dei metodi automatizzati e la complessità dei presupposti teorici di cui i sistemi informatici avrebbero dovuto tener conto per essere efficaci.¹⁸

La seconda fase comprende l’arco temporale tra la fine degli anni Settanta e la fine degli anni Ottanta. Da un punto di vista tecnologico, è determinata dalla diffusione del *personal computer* e di internet. Se durante la prima fase la ricerca era incentrata su analisi di tipo computazionale, durante la seconda la ricerca ruota intorno a due questioni: l’integrazione delle nuove tecnologie nei procedimenti di realizzazione delle edizioni critiche e la rappresentazione del testo in formato elettronico.¹⁹ Nel 1978 viene pubblicato TUSTEP,²⁰ un software per la creazione di edizioni critiche per la stampa. Già a partire dagli anni Settanta sono nati i primi archivi testuali elettronici, tra cui l’Oxford Text Archive e in Italia il corpus realizzato nell’ambito del progetto di A Valle per le *Concordanze della Lingua Poetica Italiana delle Origini* (CLPIO) e il progetto CLIPON di Savoca per le concordanze della lingua poetica italiana fra Otto e Novecento. Con la diffusione degli archivi elettronici, emerge la necessità di strategie comuni per l’elaborazione e la conservazione dei testi in formato elettronico. Proprio nel 1987 nasce la Text Encoding Initiative (TEI), riferimento cruciale per la codifica dei testi,

¹⁸ Per approfondire la storia dei sistemi di collazione automatica con anche una panoramica sugli strumenti utilizzati oggi si rimanda a Nury e Spadini 2020 e al capitolo *Highs and lows of computer-assisted stemmatics* in Trovato 2017. Spostando il *focus* dagli strumenti alla teoria filologica, come pubblicazioni che fanno il punto sul metodo del Lachmann o in generale stemmatica, all’interno del quale si inserisce il concetto di collazione, si raccomandano Trovato 2017 e Roelli (a cura di) 2020.

¹⁹ Sul testo in formato elettronico si rimanda a Shillingsburg 2006 e Orlandi 2010.

²⁰ Il software TUSTEP (Tübingen System von Textverarbeitungs-Programmen) riscosse grande successo ed è tuttora utilizzato.

la cui importanza per la pratica delle edizioni digitali è cresciuta negli anni. Nata da uno sforzo collettivo da parte di studiosi dell'Europa, dell'Asia e del Nord America, la prima versione ufficiale delle linee guida TEI (P3) fu pubblicata nel 1994,²¹ e inizialmente era basata sul linguaggio di marcatura SGML.

La terza fase si sviluppa tra l'inizio degli anni '90 e i primi anni 2000 ed è stata segnata da: la diffusione del World Wide Web; "l'affermarsi e la continua evoluzione dei linguaggi di marcatura e di programmazione"; "lo sviluppo dell'ipertesto" (Mancinelli e Pierazzo 2020). I cambiamenti tecnologici di questa fase hanno comportato due grandi cambiamenti teorici nel campo della filologia digitale. Il primo grande cambiamento riguarda il processo editoriale: se prima l'utilizzo del computer aveva modificato le metodologie di produzione di un'edizione a stampa, ora il computer modifica il prodotto e l'edizione diventa elettronica. Il secondo cambiamento è che l'edizione in formato elettronico cambia la propria natura, diventando ipertestuale e multimediale. Come osservano Buzzetti e McGann (2006), in un'edizione digitale, grazie alla possibilità di inserire delle immagini, si integrano i due diversi modelli editoriali tradizionali dell'edizione facsimile e dell'edizione critica. Il software più rilevante prodotto in questa fase è Collate, lo strumento per la collazione automatica mediante l'applicazione di metodi filogenetici, sviluppato da Peter Robinson tra il 1986 e il 1989. Nel 2002 viene pubblicata la versione P4 in XML delle linee guida TEI e, in generale, si può affermare che in questa terza fase la codifica dei testi ha giocato un ruolo fondamentale. Rispetto alla suddivisione temporale proposta da Mancinelli e Pierazzo, occorre specificare che questa terza fase potrebbe essere ulteriormente divisa in due, distinguendo tra gli anni '90 e i primi anni 2000. Nella prima parte, infatti, le edizioni scientifiche digitali erano distribuite su supporto ottico, quali i CD-ROM, mentre nella seconda vengono pubblicate sempre in forma di ipertesto ma nel web.

La quarta e ultima fase è iniziata con l'avvento del Web 2.0²² e del web semantico.²³ Nell'ambito della ricerca filologica le nuove piattaforme e i *social network* sono stati utilizzati in alcuni casi per imbastire nuove forme di lavoro collaborativo, dando origine al modello della *social edition* (Siemens et al. 2012). Per quanto riguarda il web semantico, i prodotti più rilevanti per la loro applicazione in

²¹ Per approfondire la storia della Text Encoding Initiative, si rimanda alla pagina dedicata all'interno del sito ufficiale: <https://tei-c.org/about/history/>.

²² https://www.treccani.it/enciclopedia/web-2-0_%28Lessico-del-XXI-Secolo%29/.

²³ [https://www.treccani.it/enciclopedia/web-semantico_\(Lessico-del-XXI-Secolo\)/](https://www.treccani.it/enciclopedia/web-semantico_(Lessico-del-XXI-Secolo)/).

ambito umanistico sono i modelli CIDOC CRM (*Conceptual Reference Model*) e FRBR (*Functional Requirements for Bibliographic Records*) sviluppati sotto forma di ontologia. In generale, la ricerca scientifica in questa ultima fase si è interrogata sull'interoperabilità dei dati. Rispetto a questo particolare orizzonte di ricerca lo strumento più interessante e utilizzato anche nell'ambito della filologia digitale è IIIF (*International Image Interoperability Framework*),²⁴ ideato nel 2011 e promosso da una comunità internazionale di ricercatori, biblioteche, musei e altri enti che lavorano con le immagini per migliorare l'interscambio e l'interoperabilità di immagini sul web. Nel 2016 è nato il concetto di dati FAIR: *findable, accessible, interoperable, reusable* (Wilkinson et al. 2016). Infine, in questa fase si registrano anche le prime critiche e prese di distanza dalla TEI.

1.2.2 La quinta fase?

In chiusura della rassegna storica non si può non fare riferimento a una “quinta fase” che sembra che si stia delineando in questi ultimi anni e che vede coinvolta l'intelligenza artificiale. Lo scorso 30 novembre l'azienda OpenAI ha messo gratuitamente a disposizione il modello ChatGPT, la cui performance ha stupito milioni di utenti, dando vita a un ricco dibattito sia a livello di opinione pubblica sia in ambito accademico sullo stato attuale della ricerca nel campo dell'intelligenza artificiale. Diverse persone si stanno interrogando sui possibili risvolti futuri che la diffusione dell'AI porterà in diversi aspetti della nostra società. Sull'onda dell'attuale interesse per l'intelligenza artificiale, pare opportuno fornire qualche cenno sulla sua applicazione anche in campo filologico. Alcuni ricercatori dell'Università del Kentucky sono riusciti nel 2016 a srotolare virtualmente un rotolo di pergamena carbonizzata e a leggerne il contenuto, nel 2019 a leggere dei caratteri presenti su un frammento di papiro grazie all'applicazione di metodi di *machine learning* (cfr. Brusuelas 2021, p. 45). Nel 2022, invece, alcuni ricercatori dell'Università dell'Eastern Illinois pubblicano i risultati ottenuti applicando metodi di *machine learning* per la ricostruzione di alcune parti del testo greco del Nuovo Testamento (Cfr. Jones et al. 2022, p. 5). In futuro si può immaginare un ulteriore consolidamento dell'utilizzo dell'intelligenza artificiale come strumento che permette al filologo di leggere testimoni altrimenti illeggibili e di stabilire su basi statistiche quali potrebbero essere le lezioni origi-

²⁴ <https://iiif.io/>.

nali più probabili. Di conseguenza alla comunità scientifica spetterà interrogarsi sulla validità dell'applicazione di queste tecniche nella critica testuale. Si può ipotizzare che nei prossimi anni diversi studiosi di filologia digitale dirotteranno i propri interessi di ricerca su questo nuovo campo. Occorrerà valutare in un secondo momento se questo interesse costituirà una vera e propria fase della storia della filologia digitale.

1.2.3 Cenni sui principali cambiamenti nel campo del software e dell'hardware

Nell'arco temporale trattato nella precedente sezione, quindi tra gli anni Quaranta del Novecento e oggi, tanti cambiamenti sono avvenuti nel campo dell'informatica. Per quanto riguarda l'hardware, che si è evoluto all'insegna di una progressiva portabilità, performatività e pervasività, ci si può limitare a due principali cambiamenti. Il computer si è trasformato da macchina utilizzata da un numero ristretto di persone, perlopiù all'interno di aziende e centri di ricerca, a strumento di lavoro per chiunque. Questo cambiamento è stato possibile grazie sia alla riduzione delle dimensioni e dei costi dei computer stessi, sia all'introduzione delle interfacce grafiche e dei sistemi operativi, che hanno permesso di lavorare con il computer in modo più facile e intuitivo. L'altro grande cambiamento è stato l'emergere di nuovi dispositivi portatili e di dimensioni inferiori, caratterizzati comunque da un'elevata potenza di calcolo. Dopo i primi palmari, *smartphone* e *tablet* sono oggi oggetti di uso quotidiano e, per quanto riguarda gli scopi della presente ricerca, sono a tutti gli effetti supporti ampiamente utilizzati anche per la lettura e lo studio.²⁵

Passando al software, i primi programmi erano scritti in codice binario, erano dunque costituiti da lunghe sequenze numeriche che i programmatori fornivano in input ai computer *mainframe* attraverso schede perforate oppure mediante gli interruttori presenti sull'interfaccia dei computer. Successivamente vennero creati i linguaggi di programmazione a partire dal linguaggio assembler (o *assembly*), sviluppato negli anni Cinquanta. Il primo linguaggio procedurale è stato FORTRAN, creato nel 1954 dalla IBM, mentre Lisp è il primo linguaggio di tipo funzionale, creato sempre verso

²⁵ Un altro tipo di supporto elettronico per la lettura che si è diffuso negli ultimi decenni è l'e-reader.

la fine degli anni Cinquanta. I primi linguaggi orientati agli oggetti, ovvero Smalltalk e Simula, vengono sviluppati tra la fine degli anni Sessanta e l'inizio degli anni Settanta, ma il paradigma della programmazione a oggetti si diffonde e si consolida negli anni Ottanta con il linguaggio C++.

Con l'avvento di internet e poi del *world wide web*, sorgono due nuove esigenze: da un lato lo sviluppo di tecnologie per il web, dall'altro la definizione di nuove architetture per mettere sistemi diversi in comunicazione tra di loro attraverso il web. Riguardo al primo punto il W3C, il consorzio fondato nel 1994 da Tim Berners-Lee, ha giocato un ruolo fondamentale, sviluppando e mantenendo le tecnologie standard del web, tra cui HTML, CSS e XML. Alla seconda esigenza, invece, ha fatto fronte l'emergere di nuovi modelli di architetture per realizzare sistemi distribuiti, tra cui CORBA, sviluppato dall'Object Management Group,²⁶ e COM di Microsoft. In questa fase continuano a proliferare nuovi linguaggi di programmazione. I due linguaggi maggiormente affermati per lo sviluppo di applicativi web sono Java e JavaScript.

I primi anni 2000 sono segnati dalla diffusione di XML, la cui prima versione risale al 1998, e di una nuova metodologia per lo sviluppo di software chiamata *agile*. Questa metodologia si contrappone a quella tradizionale detta *waterfall* (a cascata)²⁷ ritenuta troppo rigida e poco efficace per un rapido sviluppo di applicativi. In questi anni nasce anche il web semantico, un'infrastruttura di metadati che aiutano le macchine a interpretare su un piano semantico e a ricercare le risorse pubblicate in rete. Il W3C sviluppa una serie di standard dedicati tra cui RDF, N3 e OWL.

Intorno al 2010 nascono dei nuovi paradigmi per lo sviluppo di software, per cui gli applicativi tendono sempre più a essere strutturati come collezioni di servizi distribuiti mediante il web. Al

²⁶ L'Object Management Group (OMG) è un consorzio nato nel 1989 con l'obiettivo di creare un sistema di gestione di un'architettura distribuita, il menzionato CORBA. Successivamente il consorzio si è dedicato alla definizione di altri standard per il settore informatico, tra cui il più noto è UML. Sito web del consorzio OMG: <https://www.objectmanagementgroup.org/>.

²⁷ Con "waterfall model" si intende un modello del procedimento di sviluppo di software, all'interno del quale le diverse fasi, solitamente la fase di concettualizzazione, la fase di specifica dei requisiti, la fase di progettazione, la fase di implementazione, la fase di test e la fase di installazione e controllo finale, vengono eseguite in quest'ordine, eventualmente con qualche sovrapposizione ma senza reiterare (se non in minima parte) le diverse fasi. (Cfr. *Systems and Software Engineering Vocabulary*).

tempo stesso i programmatori si domandano come produrre software più robusto e facile da mantenere. Nascono così degli approcci alla programmazione software che hanno riscosso molto successo, da un lato il *domain-driven design*, che mira a realizzare software più robusti modellando la loro logica interna sulle caratteristiche dell'ambito (dominio) all'interno del quale vengono utilizzati, e dall'altro lato il *test driven development*, che prevede che un programma venga sviluppato partendo dai test, cioè da quelle porzioni di codice che verificano il corretto funzionamento del resto del codice.

L'enorme quantità di dati che viene riversata in rete, anche in seguito alla diffusione massiva dei *social network*, porta molte aziende informatiche e sviluppatori a interrogarsi su come sfruttarla a scopi sia commerciali che di ricerca. Negli ultimi anni l'analisi di dati e in particolare l'intelligenza artificiale è diventato il campo della ricerca informatica in cui a livello tecnologico le aziende stanno maggiormente investendo.

Così come l'hardware è alla portata di tutti, anche lo sviluppo di software oggi è altrettanto accessibile e facilmente realizzabile. Questa maggiore facilità è dovuta anche alla possibilità di ospitare i propri applicativi in *cloud* su server esterni e di poter accedere a servizi esterni tramite API o poter riutilizzare porzioni di codice libero e aperto, abbattendo notevolmente i costi dello sviluppo.

1.2.4 Considerazioni conclusive

Facendo riferimento alle fasi descritte nella precedente sezione, si può affermare che anche i software per la filologia digitale si siano alternati nel tempo in modo simile, in base agli obiettivi scientifici degli studiosi e alle nuove tecnologie che via via si sono imposte. Nella prima fase di sperimentazione, gli strumenti prodotti sono stati sviluppati appositamente per assolvere a dei quesiti filologici. Si trattava di strumenti utilizzati da un numero circoscritto di studiosi, ma che hanno permesso alla comunità scientifica di riflettere sul rapporto tra tecnologie informatiche e critica del testo, saggiando per la prima volta le potenzialità, i limiti e le peculiarità della filologia computazionale.

Durante la seconda e la terza fase, il computer è diventato gradualmente uno strumento di lavoro sempre più diffuso e adottato.

“Dagli anni ottanta, con la diffusione del personal computer, lo strumento informatico più semplice impiegato nella preparazione di un'edizione critica, così da costituire il «livello zero» dell'ecdotica

elettronica, è il word processor; tuttavia il testo elettronico così prodotto resta spesso un «formato di passaggio» o un «byproduct» dal momento che l'obiettivo dell'editore continua a essere un testo a stampa." (Rassegna delle risorse digitali a cura di Cinzia Pusceddu in Fiormonte 2003)

Inizialmente i software prodotti in questo periodo, quali TUSTEP, affiancavano e/o sostituivano lo studioso nel flusso di lavoro per la produzione di un'edizione critica a stampa. Successivamente anche l'edizione è diventata elettronica, fruita prima tramite floppy-disk e CD-ROM e infine come pagina web ipertestuale. Nella terza fase, dunque, la pratica ecdotica non è più vincolata alla pubblicazione a stampa, ma conosce ed esplora nuove forme, grazie alla separazione tra sorgente e output. Al tempo stesso, la codifica dei testi in formato elettronico fa sì che gli studiosi sperimentino nuove modalità sia di analisi del testo che di produzione delle edizioni critiche.

La quarta fase, ad avviso di chi scrive, è contraddistinta da due tendenze apparentemente contraddittorie tra loro. Da un lato il *personal computer* e il web sono diventati degli strumenti di lavoro imprescindibili per il filologo, che oggi dispone in rete di molti materiali, tra cui le riproduzioni fotografiche dei manoscritti, cataloghi, archivi, edizioni, risorse lessicografiche, ecc. Dall'altro lato, dopo l'iniziale curiosità suscitata dalle prime edizioni elettroniche e una fase di sperimentazione, l'edizione elettronica o digitale non ha ancora raggiunto lo stesso grado di autorevolezza e affidabilità tradizionalmente riconosciute all'edizione a stampa. Allo stesso modo le metodologie filologiche e i modelli editoriali nati nel web non sembrano aver superato il confine dell'ambiente digitale, fallendo nel compito di diventare strumenti di riferimento per la filologia *tout court*.²⁸ Dunque gli studiosi di filologia digitale e di *digital humanities* si stanno interrogando su come rimediare alle difficoltà tecniche e scientifiche poste dall'utilizzo delle tecnologie digitali, in modo tale da contribuire anche a conferire maggiore stabilità e autorevolezza alla disciplina.

²⁸ Rispetto a questo punto è interessante osservare come nella maggior parte dei manuali di filologia rivolti agli studenti universitari siano presenti solo dei cenni alla ricerca svolta nel campo della filologia digitale, relegati solitamente nelle note o in brevi sezioni dedicate. Il lavoro dei filologi digitali rimane in qualche modo separato e quindi subordinato rispetto a quello dei colleghi che destinano le proprie edizioni alle stampe.

1.3 Gli strumenti informatici per la filologia digitale oggi

1.3.1 Il modello “sorgente-*output*”

Negli ultimi venticinque anni, grazie alla progressiva diffusione e consolidamento di TEI come standard *de facto* per le edizioni scientifiche digitali, la maggior parte di quelle prodotte sono inquadrabili nel modello concettuale e tecnologico “sorgente-*output*”:

“Secondo questo approccio, il testo annotato (o codificato) viene archiviato in uno o più file che nel loro complesso prendono il nome di sorgente (al maschile); nella maggior parte dei casi i file sorgente sono codificati nel linguaggio XML secondo le linee guida prodotte dalla TEI, che rappresentano lo standard di fatto nel campo delle edizioni digitali, oltre che un importante riferimento intellettuale” (Mancinelli e Pierazzo 2020).

Di conseguenza, anche gli strumenti informatici prodotti per la pratica ecdotica sono stati sviluppati prendendo a riferimento il modello “sorgente-*output*”:

“Una serie di programmi informatici prendono il sorgente come punto di partenza (o *input*) e producono diversi oggetti digitali in uscita (detti, quindi, *output*): per esempio, partendo dallo stesso sorgente XML possono produrre una o più versioni HTML da pubblicare sul Web, un testo stampabile in PDF, un eBook da leggere su un tablet, degli indici interattivi, ecc.” (*idib.*)

Per poter comprendere meglio la natura di questi strumenti, occorre precisare tre aspetti del loro funzionamento, rispetto a come è stato descritto sopra. Punto primo: molto spesso “il testo annotato (o codificato)” non è l’unico materiale di partenza per allestire un’edizione digitale. L’edizione può comprendere immagini, contenuti multimediali e altri dati di interesse scientifico strutturati in formati diversi, reperiti da basi di dati, *repository* e altre architetture di *data management*. Si pensi alla diffusione dello standard per la condivisione di immagini IIIF (*International Image Interoperability Framework*) e al suo utilizzo anche all’interno di edizioni critiche digitali.

Il secondo punto riguarda quanto XML/TEI sia effettivamente adottato come formato per la realizzazione del file sorgente. Sulle 320 edizioni digitali catalogate nel *Catalogue of Digital Editions* di Franzini, l’utilizzo di XML/TEI all’interno dell’edizione è registrato sia nel campo “infrastruttura”, dove è menzionato solo in 35 sui 178 record per cui sono disponibili i dati sull’infrastruttura (poco meno del 20%), sia in un campo a parte “*tei_transcription*”, dove, invece, l’utilizzo di TEI è

registrato in 206²⁹ record su 320, che corrisponde al 64% circa del totale. Incrociando questo secondo dato con l'anno di inizio di realizzazione dell'edizione digitale, si ottiene il seguente grafico.

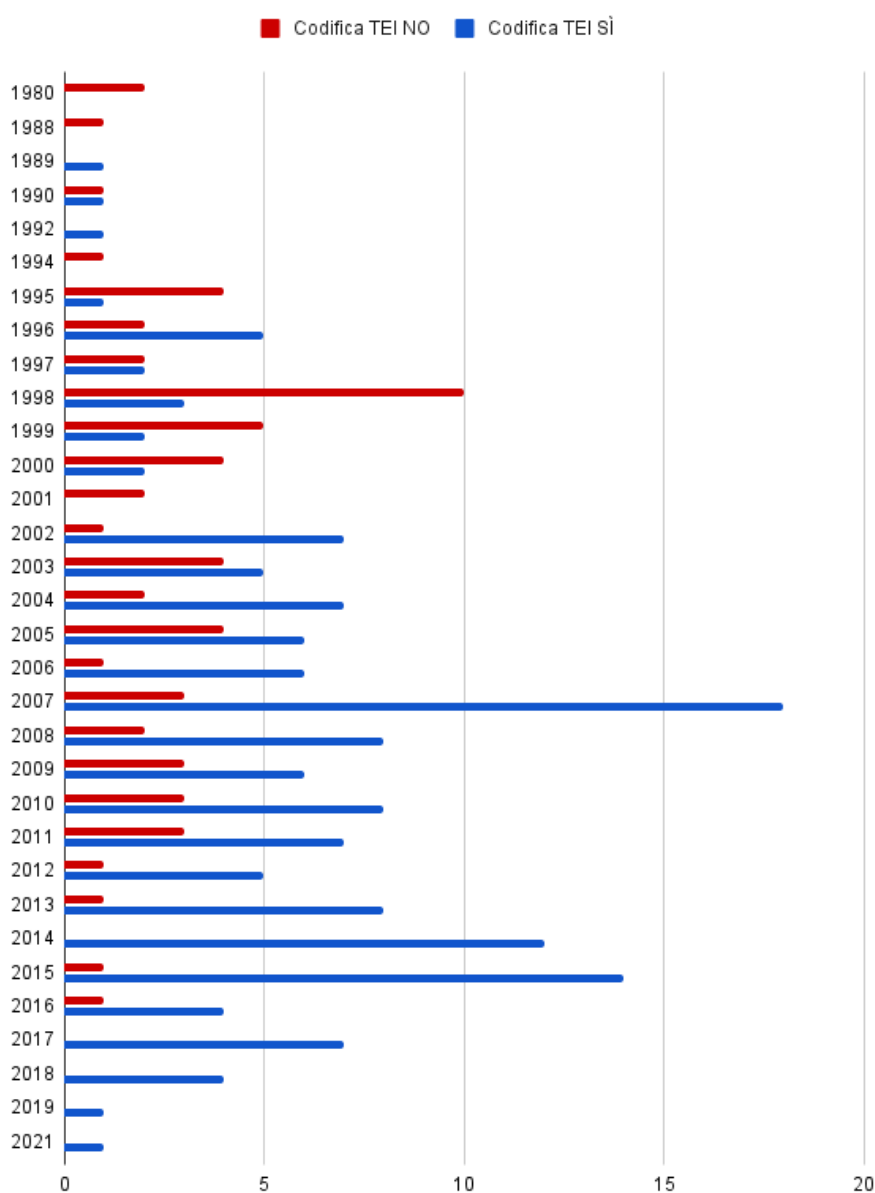


Figura 1.2 Grafico che mostra il numero di edizioni scientifiche digitali che sono basate su documenti in XML/TEI divise per anno in cui la loro realizzazione è stata avviata.

Ovviamente, nei primi anni presentati nel grafico, che corrispondono al periodo antecedente e subito successivo la nascita della TEI, i numeri sono molto ridotti. A partire dal 2002 si registra una crescita, che raggiunge dei picchi nel 2007 e di nuovo tra il 2014 e il 2015. Mentre negli anni più

²⁹ Il campo "tei_transcription" prevede tre possibili valori: 1 (trascrizione in TEI presente), 0 (trascrizione in TEI assente) e 0.5 che però non è chiaro a che cosa corrisponda. Ai fini del presente calcolo statistico, i 24 record contrassegnati con valore 0.5 sono stati uniti alle 182 edizioni che si sono servite di XML/TEI.

recenti si nota un notevole calo. L'andamento del grafico molto probabilmente è influenzato da *bias* dovuti alle tempistiche e modalità in cui il catalogo stesso è stato compilato. Ad ogni modo, si può affermare che la maggior parte delle edizioni digitali utilizza e ha utilizzato XML/TEI, sebbene non si tratti di una netta maggioranza. Per questo motivo nella maggior parte degli strumenti informatici per la filologia digitale XML/TEI è supportato come formato di input e/o di output.

Terzo punto: dato che il modello concettuale “sorgente-*output*” si basa su una separazione netta tra i dati (il contenuto) e la loro presentazione (o visualizzazione), sorge spontaneo interrogarsi su quale sia il fulcro di un'edizione scientifica digitale, se i dati che contiene o la loro presentazione.³⁰ Nel presente lavoro di tesi il concetto di edizione è considerato come l'unione di dati e presentazione. Senza addentrarsi troppo nel dibattito, si può concordare che i dati siano il fulcro, la sostanza dell'edizione scientifica digitale. Se l'edizione, però, non è pubblicata, non può essere considerata un'edizione, perché non realizza il suo fine ultimo. Inoltre, solo mediante la pubblicazione l'edizione diventa davvero uno strumento di ricerca scientifica, in quanto diviene verificabile, citabile e riutilizzabile da altri studiosi. Il concetto di pubblicazione è legato a doppio filo con il concetto di fruibilità. Un'edizione marcata in XML/TEI se pubblicata sotto forma di documenti di testo è sì accessibile al pubblico, ma risulta fruibile da un numero più circoscritto di lettori, dato che richiede quanto meno delle conoscenze base di XML e di TEI per poter essere interpretata correttamente. Se la medesima edizione viene trasformata in una pagina web e formattata graficamente emulando le convenzioni tipografiche delle edizioni a stampa, risulta fruibile da un pubblico più ampio. In altre parole, la presentazione ha lo scopo di eliminare o perlomeno attenuare le possibili barriere tra i dati e il pubblico di lettori. Nel presente lavoro di tesi, dunque, sono stati analizzati non solo applicativi che permettono di produrre i dati alla base di un'edizione scientifica digitale, ma anche quelli che consentono di visualizzarla, in quanto la presentazione è considerata come una parte fondamentale del concetto di edizione.

³⁰ Per approfondire la questione si rimanda al volume *Digital Scholarly Editions as Interfaces* (Bleier et al. 2018) e al contributo di Barabucci, Spadini e Turska 2017, in cui gli autori riflettono se: “It is possible to think of a critical edition as a collection of pieces of pure data? Or is a representation layer fundamental to the concept of ‘edition?’” (p. 39).

1.3.2 Principali categorie di strumenti per le edizioni critiche digitali

Nella presente sezione vengono descritti gli strumenti informatici per la filologia digitale più noti e utilizzati che sono stati prodotti negli ultimi anni. Il focus di questa rassegna è stato posto sugli strumenti che consentono di produrre un *output* destinato alla pubblicazione e fruizione nel web. In generale, il prodotto finale di questo tipo di strumenti è una “risorsa web”, a prescindere che si tratti semplicemente di un documento in formato HTML, di un sito web, di una raccolta di oggetti digitali (archivio o biblioteca digitale) o di una collezione di *linked open data*.

Prendendo a riferimento il modello “sorgente-*output*” e quindi la distinzione tra dati e presentazione, questo tipo di strumenti possono essere suddivisi in tre principali categorie:

1. **gli strumenti di produzione**, con i quali è possibile produrre i file sorgente o, più in generale, i dati alla base dell’edizione;
2. **gli strumenti di visualizzazione**, che consentono di generare l’output finale o di pubblicare l’edizione come risorsa web;
3. **gli strumenti sia di produzione che di visualizzazione (o misti)**, che permettono di svolgere entrambe le operazioni per realizzare un’edizione scientifica digitale nella sua interezza.

Come anticipato (vd. 1.1), per rendere la catalogazione più chiara a un pubblico generico di filologi, gli strumenti informatici per la filologia digitale possono essere classificati in base alle fasi tradizionali del lavoro ecdotico. Delle fasi già elencate nella prima sezione del capitolo, si è ritenuto opportuno proporre come categorie a parte per gli strumenti informatici di produzione la trascrizione dei testimoni (o testimone) e la collazione. A queste due fasi conviene aggiungere altri due tipi di attività che spesso vengono effettuate contestualmente alla realizzazione di un’edizione critica: da un lato l’analisi del testo, intesa anche come lemmatizzazione o analisi delle concordanze o altre operazioni analoghe di analisi approfondita del testo edito; dall’altro lato l’annotazione, quindi l’operazione di apporre commenti ai testi e agli altri materiali su cui l’edizione è basata.

Per realizzare un’edizione scientifica digitale possono essere impiegati più strumenti di produzione diversi, ciascuno che assolve a un compito specifico. Un’edizione può essere realizzata, per

esempio, partendo dalla trascrizione dei testimoni mediante un applicativo di HTR. Successivamente le trascrizioni ottenute potrebbero essere collazionate automaticamente da un applicativo apposito. Infine, i file sorgenti risultanti, in formato XML/TEI o altro, potrebbero essere trasformati in risorse web pronte per la pubblicazione. Ogni strumento di produzione genera in output dei file sorgente, che contengono dei dati. Questi file possono essere integrati così come sono all'interno dell'edizione scientifica digitale oppure possono essere elaborati ulteriormente o convertiti in altri formati per poi rientrare nell'edizione scientifica digitale. Si offre uno schema che illustra come l'edizione sia realizzata attraverso diverse fasi di lavoro, che possono essere effettuate applicando degli strumenti di produzione, risultando poi in una serie di file sorgente che costituiscono la base dell'edizione digitale.

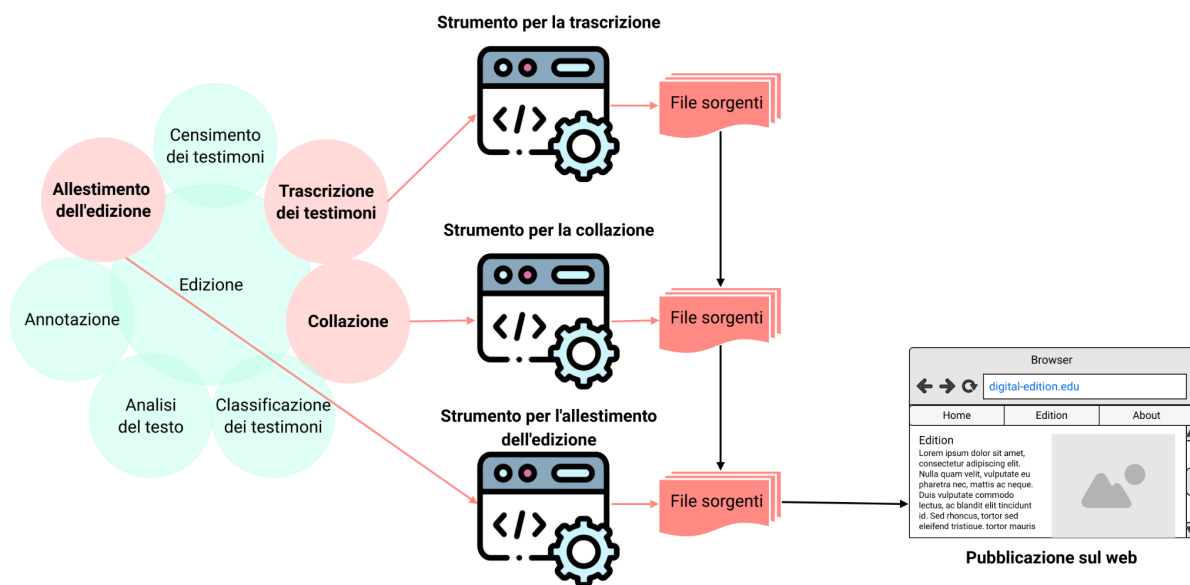


Figura 1.3 Schema del funzionamento degli strumenti di produzione.

Un problema che si riscontra di frequente, quando si lavora con molteplici strumenti di produzione, è l'incompatibilità tra i file prodotti da ciascuno strumento. Talvolta questo problema può essere risolto grazie a conversioni di formato intermedie. Altre volte, nonostante il formato sia lo stesso, ci possono essere delle incompatibilità che dipendono dal modo in cui sono stati strutturati i dati. Per esempio, due file XML/TEI possono essere resi non immediatamente interoperabili per via dell'utilizzo di schemi di codifica diversi. Con schema di codifica o (modello di codifica) si intende un documento che definisce la struttura e la semantica di un documento XML, in altre parole quali sono i *tag* (marcatori) e attributi contemplati e come possono essere organizzati tra di loro. Le linee

guida TEI sono un esempio di schema di codifica, in quanto forniscono un vocabolario di possibili elementi e di possibili combinazioni che possono essere utilizzate. Ciascun utente, entro certi limiti, può strutturare il proprio documento XML come ritiene più opportuno, producendo comunque un documento compatibile con le indicazioni della TEI

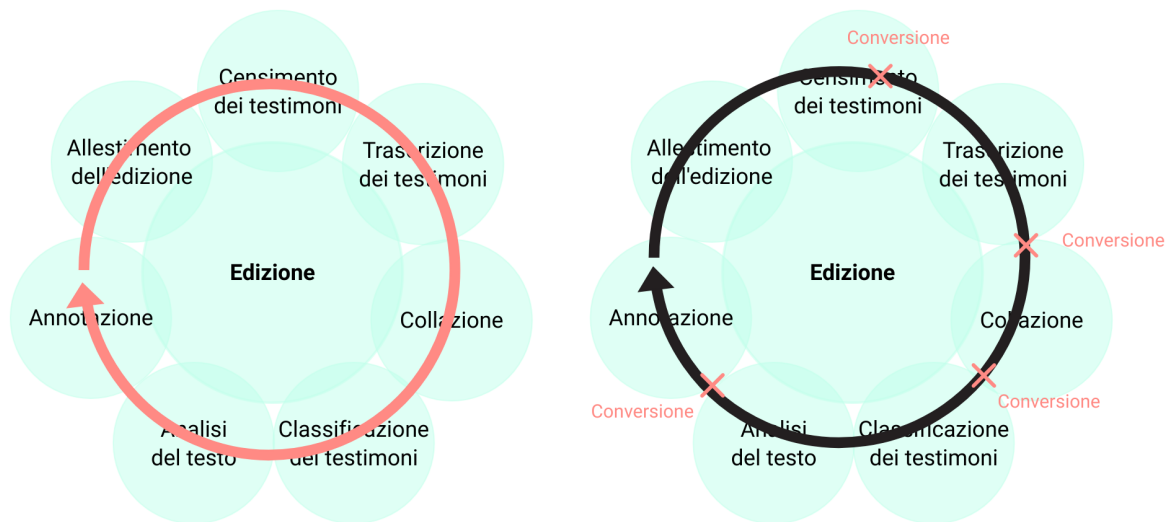


Figura 1.4 Schema che rappresenta le difficoltà nella conversione tra i formati di output di strumenti informatici diversi.

Per ovviare a questo problema, molti applicativi per la filologia digitale sono stati progettati come soluzioni *all in one*, che accompagnino il filologo dall'inizio alla fine del lavoro editoriale. Sono cioè degli strumenti misti che possono essere utilizzati sia per produrre sia per visualizzare un'edizione digitale.

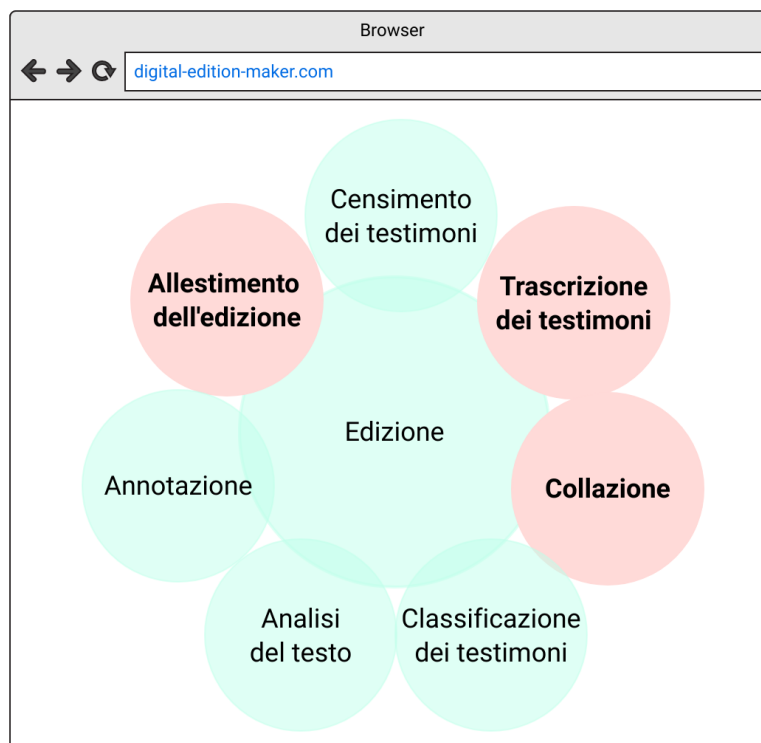


Figura 1.5 Schema del funzionamento degli strumenti misti.

Il già menzionato TUSTEP rientra proprio in questa categoria. Concentrandoci, però, su quanto è stato sviluppato negli ultimi anni, si tratta tendenzialmente di piattaforme online che permettono di caricare i propri materiali di partenza (riproduzioni fotografiche dei testimoni, trascrizioni, ecc.), per poi elaborare ulteriormente i dati e pubblicarli sotto forma di edizione in rete. Questa tendenza è legata al fatto che il web ha permesso di implementare nuove modalità di lavoro decisamente più collaborative anche nel campo della filologia.

Per le edizioni digitali il confine tra fase di produzione e fase di visualizzazione non è facilmente riconoscibile, mentre nelle edizioni a stampa solitamente è ben tracciato anche da un passaggio di consegne, dall'editore/filologo al tipografo, e dal passaggio da materiali privati a volume pubblicato. Per gli scopi della presente analisi si può dire che uno strumento di sola visualizzazione non aggiunge nuovi dati di interesse filologico rispetto a quelli ricevuti in input, mentre uno strumento di produzione genera nuove informazioni. Si pensi a un applicativo per la collazione automatica, che in input riceve due o più testi, mentre in output genera di solito un unico file, in cui i due o più testi sono

allineati, mettendo in evidenza i punti di accordo e di disaccordo tra di essi. Gli allineamenti individuati dallo strumento sono delle nuove informazioni di interesse filologico che non erano presenti in input.

Uno strumento di sola produzione, invece, non consente di visualizzare né di pubblicare un'edizione scientifica digitale nella sua interezza. Al più permette di visualizzare (e anche di condividere e pubblicare) i dati generati, che però costituiscono solo una parte dei contenuti dell'edizione digitale.

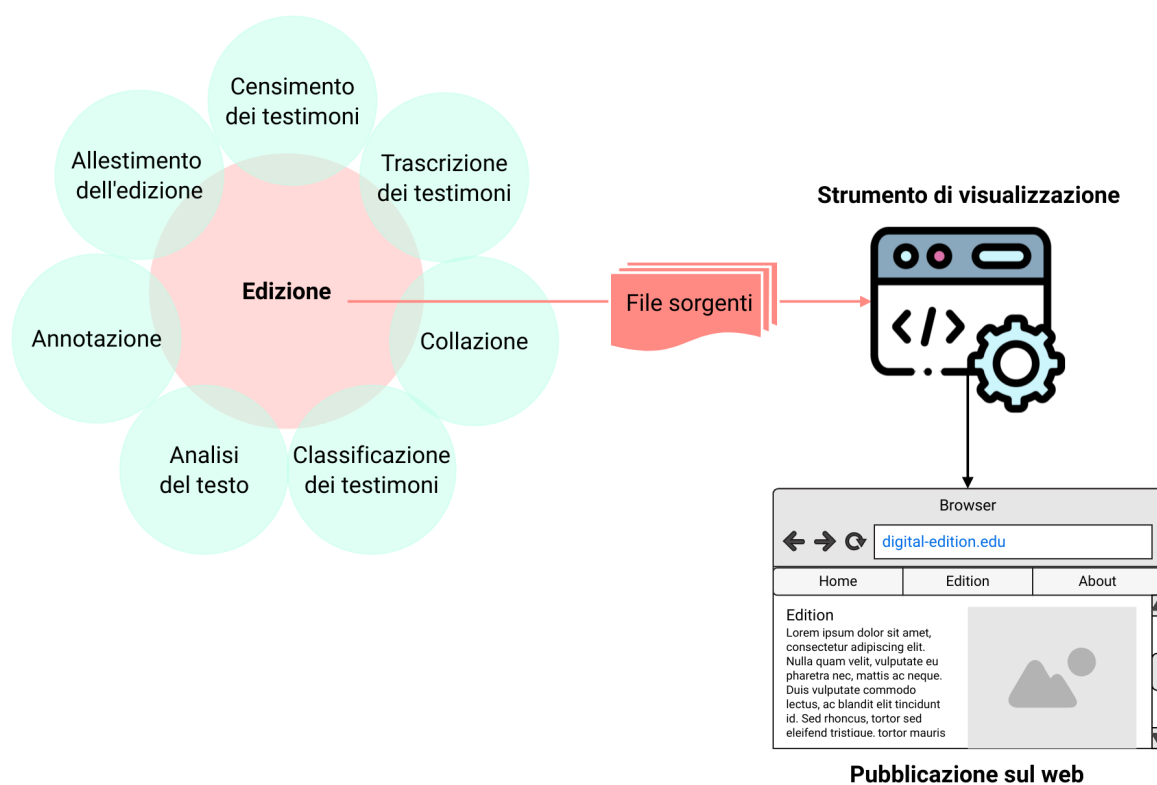


Figura 1.6 Schema del funzionamento degli strumenti di visualizzazione.

1.3.3 Gli strumenti di produzione

Per gli strumenti di produzione è possibile individuare un “grado zero”, rispetto al modello “sorgente-output” e alla diffusione della TEI: gli *editor* per scrivere documenti in formato XML. In questa categoria rientrano sia gli *editor* specifici per XML, come il già menzionato Oxygen, che attualmente è tra i più utilizzati, così come tutti gli altri *editor* generici, per esempio l'applicativo Visual Studio Code³¹ prodotto da Microsoft.

³¹ Sito web di Visual Studio: <https://code.visualstudio.com/>.

Negli anni sono stati prodotti *editor* di documenti XML/TEI più *user-friendly*, che grazie a un'interfaccia simile a quella dei *word processor* non richiedono all'utente di marcare. La maggior parte delle volte si tratta di applicativi web, che consentono di effettuare non solo l'editing, ma anche altre operazioni, per esempio la trascrizione dei testimoni, la collazione, l'analisi del testo e anche la creazione di *linked open data* in formato RDF, come nel caso di LEAF-Writer*. Per suddividere in ulteriori categorie gli strumenti di produzione che assolvono a più compiti, si terrà di conto della loro funzionalità primaria oppure della funzionalità per cui sono stati maggiormente adottati dalla comunità scientifica. Ad ogni modo, per ogni strumento verrà precisato se si tratta di uno strumento che permette di svolgere più attività.

1.3.3.1 Trascrizione

L'avvento del web e delle nuove tecnologie ha permesso agli studiosi di semplificare notevolmente il lavoro di trascrizione, da un lato usufruendo di ambienti di lavoro digitali, dall'altro potendo adottare due modalità innovative di trascrizione, distinte ma complementari: la trascrizione automatica effettuata dalle macchine e la trascrizione collaborativa.

La trascrizione automatica inizialmente era condotta attraverso processi detti di *optical character recognition* (OCR), che convertono dei testi a stampa in immagini digitali e in testi elettronici. Si tratta della tecnologia che ha permesso al Progetto Gutenberg e soprattutto a Google Books di arricchire le proprie biblioteche digitali di quantità enormi di testi. L'utilizzo di questi strumenti anche in ambito scientifico è stato accolto con un certo scetticismo iniziale da parte della comunità degli studiosi, in quanto i primi software di OCR erano noti per i loro alti margini di errore e per la possibilità di essere usati quasi esclusivamente con testi a stampa. La maggiore regolarità dei caratteri stampati nella forma e nella loro disposizione sulla pagina facilita, infatti, il riconoscimento da parte dei software.

In termini di resa e accuratezza i moderni software di OCR, tra cui ABBYY FineReader* e Transym*, non hanno nulla a che vedere con i loro predecessori. Il funzionamento di fondo di questi procedimenti è stato ampiamente rivisto e migliorato. Software quali OCRopus* e Tesseract* sono basati su reti neurali che permettono di riconoscere le intere righe invece dei singoli caratteri. Inoltre,

si sono diffusi strumenti di *handwritten text recognition* (HTR) che sono in grado di leggere in automatico testi manoscritti, con margini di errore piuttosto limitati. Margini che possono essere ulteriormente diminuiti grazie ai principi del *machine learning*, quindi “allenando” la macchina su corpora testuali oppure fornendo alla macchina vocabolari per poter interpretare correttamente le parole.³²

Il progresso tecnologico ha richiesto qualche anno per arrivare a produrre degli strumenti di trascrizione automatica adeguati. Nel mentre, diversi studiosi si sono interrogati su come sfruttare le più recenti tecnologie, in particolare i *social media*, nel campo delle *digital humanities*, formulando un nuovo modello di edizione battezzato, appunto, “social edition” (Siemens et al. 2012). Tra le diverse attività dell’*editing* filologico, la trascrizione si presta molto a essere condotta collaborativamente.³³ Nel processo di trascrizione dei testimoni all’interno di un progetto di edizione possono essere coinvolte solo persone che possiedono determinate competenze tecniche, quindi studiosi e accademici, oppure volontari di ogni sorta, servendosi di strategie di vero e proprio *crowdsourcing*, per incoraggiare il maggior numero possibile di volontari a partecipare.³⁴ Solitamente le trascrizioni ad opera di volontari vengono revisionate da un gruppo di editori esperti, in modo tale da garantire l’affidabilità e qualità del prodotto scientifico. Una delle prime e più celebri iniziative di *crowdsourcing* per la trascrizione online è *Transcribe Bentham*,³⁵ lanciata nel 2010 nel contesto del Bentham Project dello University College di Londra.

Per descrivere gli applicativi di trascrizione, partiamo da quelli che non si servono di processi automatici di OCR o HTR, ma che si contraddistinguono in quanto facilitano il lavoro di mediante

³² Per una prospettiva storica più dettagliata sull’applicazione dei software di OCR e HTR nel campo delle *digital humanities* si rimanda a Muehlberger et al. 2019.

³³ Cfr. “A definitively ‘social’ edition employs new and emerging tools for interaction around such activities as transcription, bookmarking and bibliography-building, flagging and tagging, commenting and annotating, linking to contextual material (especially for names and integration of bibliographic information), glossary and other analytical functions, and all other pertinent activities that sit at the evolving intersection of social media and the electronic scholarly edition.” (Siemens et al. 2012, p. 453)

³⁴ Si riporta, più come curiosità che come informazione di interesse scientifico, che talvolta i “trascrittori” online sono inconsapevoli di trascrivere. Quando si trascrive la porzione di testo presentata in un reCAPTCHA si contribuisce alla trascrizione di un testo che nella sua forma a stampa presenta un aspetto grafico troppo arzigogolato per essere interpretato correttamente da un software di OCR.

³⁵ Pagina web dell’iniziativa *Transcribe Bentham*: <https://blogs.ucl.ac.uk/transcribe-bentham/>.

un'interfaccia *user-friendly* e che in alcuni casi possono essere usati collaborativamente. Tendenzialmente si tratta di strumenti che permettono di caricare le immagini dal proprio dispositivo oppure attraverso IIF. L'utente visualizza l'immagine e accanto dispone di un'interfaccia di *editing*, con la quale può trascrivere agevolmente quanto legge nell'immagine. Questi strumenti molto spesso offrono anche delle funzionalità per collegare porzioni di testo più o meno estese alle rispettive aree nell'immagine. Terminato il lavoro di trascrizione, il testo elaborato e gli altri file prodotti (per esempio i file contenenti i riferimenti spaziali delle aree individuate dall'utente nelle immagini) possono essere esportati in diversi formati. Un esempio di questo tipo di applicativo è il recente FairCopy*.

Al giorno d'oggi diversi strumenti per la trascrizione di documenti offrono sia funzionalità di riconoscimento automatico da parte della macchina sia di lavoro collaborativo mediante il web. Tra questo tipo di strumenti uno dei più popolari è la piattaforma online Transkribus*, utilizzata anche dall'iniziativa *Transcribe Bentham* sopra citata.

Si menziona infine la piattaforma web FromThePage*, che si distingue dagli altri strumenti per la trascrizione finora citati, in quanto offre a pagamento un servizio di trascrizione compiuta esclusivamente da volontari, senza l'ausilio di macchine. Si rivolge perlopiù a biblioteche, archivi e altri enti che vogliono digitalizzare il proprio patrimonio librario o documentale e al momento dispone di più di 5000 volontari.

1.3.3.2 Collazione

Tra gli attuali software per la collazione automatica il più celebre è sicuramente CollateX*, nato come una completa riscrittura di Collate, il software sviluppato da Peter Robinson. Quest'ultimo strumento è stato largamente apprezzato e utilizzato dalla comunità scientifica per realizzare edizioni critiche, tuttavia si è resa necessaria una completa riscrittura, in quanto, da un punto di vista architettonico, Collate risultava complesso da distribuire e da integrare in altri sistemi informatici. CollateX consente di leggere e confrontare due o più versioni di un testo. Ciascuna versione viene divisa in più parti, o token, che possono essere comparati e allineati per individuare similarità e differenze

tra le versioni. Il risultato di questo allineamento può essere utilizzato per realizzare un apparato critico o per l'analisi stemmatica della genesi di un testo. Inoltre, l'applicativo permette di visualizzare graficamente gli allineamenti, generando un output in formato GraphML e GraphViz DOT.³⁶

Un altro software per il confronto e la collazione automatica che è stato molto utilizzato è lo strumento *open source* Juxta*, che però da diversi anni non è più sviluppato né mantenuto. Gli sviluppatori di Juxta insieme agli sviluppatori di CollateX hanno basato il proprio lavoro sul cosiddetto “modello Gothenburg”,³⁷ che prevede cinque fasi distinte, in cui è suddivisibile il flusso di lavoro di una collazione automatica:

1. **tokenizzazione**, cioè suddivisione del testo in unità, che possono corrispondere a sillabe, parole o elementi strutturali più ampi, quali periodi, strofe e paragrafi;
2. **normalizzazione** o regolarizzazione, che permette di stabilire delle equivalenze tra token che differiscono solo su un piano grafico-fonetico;
3. **allineamento** dei token che concordano, inserendo degli spazi vuoti così che le sequenze di token tratti dai diversi documenti siano tutte incolonnate;
4. **analisi/riscontro**, fase durante la quale l'utente revisiona ed eventualmente corregge manualmente l'allineamento prodotto dal software;
5. **visualizzazione** del risultato finale.

CollateX deve essere scaricato e installato su *desktop*, mentre LERA, un altro strumento per la collazione automatica, può essere utilizzato come servizio web, contattando prima gli sviluppatori e chiedendo loro di creare una copia dell'applicativo sui loro *server*, che è poi accessibile attraverso un indirizzo URL specifico.

Uno strumento di collazione automatica atipico, che al momento sembra essere ancora in fase sperimentale, è HyperCollate*, concepito per confrontare diverse versioni di un testo non sotto forma di stringhe testuali, ma come grafi. La collazione da grafo a grafo consente di tenere conto della

³⁶ Per approfondire la questione della validità scientifica ed efficacia della rappresentazione attraverso un grafo del risultato di una collazione si rimanda a Schmidt e Colomb 2009.

³⁷ Il modello prende il nome dalla città in cui nel 2009 si incontrarono gli sviluppatori di CollateX e di Juxta per individuare i diversi compiti cui un software di collazione automatica deve assolvere, al fine di avere un quadro teorico comune che facilitasse lo sviluppo collaborativo degli strumenti e dei loro componenti.

struttura gerarchica dei documenti e della presenza di revisioni nel testo (aggiunte, sostituzioni, ecc.), senza appiattirle in un'unica stringa di testo che coincide unicamente con il contenuto testuale.

Il tema della visualizzazione dei risultati della collazione in ambiente digitale è molto interessante. Si può osservare che diversi strumenti per la collazione automatica attualmente disponibili offrono sia modalità di visualizzazione innovative, sia modalità di visualizzazione che sono state mutate dalle edizioni a stampa, dalle tavole di allineamento all'edizione sinottica.³⁸ Tra gli strumenti per la visualizzazione grafica degli esiti di una collazione automatica si cita TRAViz*, una libreria JavaScript che genera grafi di varianti testuali aventi la particolarità di essere interattivi. In particolare, l'utente può modificare, mediante una funzionalità di *drag and drop*, la struttura del grafo, separando o unendo vertici di ramificazione diverse, in modo tale correggere o rimuovere possibili errori prodotti dall'algoritmo di allineamento. Solitamente questo livello di manipolazione non viene implementato negli strumenti di visualizzazione di una collazione automatica per non invalidare il valore scientifico dei risultati generati:

“the possibility of adjusting the visualisation and thus the representation of variation foregrounds the idea that the output of a tool is open to interpretation” (Bleeker et al. 2019a, p. 151).

1.3.3.3 Annotazione

Il termine annotazione necessita di essere disambiguato, in quanto può indicare sia l'annotazione semantica, intesa come codifica o marcatura, per esempio in XML/TEI, sia l'annotazione da parte del filologo, quindi il commento filologico o più in generale critico. In questa sezione e ai fini della classificazione degli strumenti, il termine annotazione è inteso come commento critico.

Nel campo della critica testuale, l'annotazione può essere effettuata tendenzialmente per due scopi diversi: può essere finalizzata a uno studio individuale o collaborativo dei testi e dei materiali alla base dell'edizione, condotto nel mentre che si realizza l'edizione; oppure può essere la modalità mediante la quale lo studioso aggiunge ai materiali alla base dell'edizione (tendenzialmente le trascrizioni e le riproduzioni fotografiche dei manoscritti) una sorta di livello “critico”, annotando delle

³⁸ Per un approfondimento sulle diverse modalità di visualizzazione degli esiti della collazione utilizzate nel web e sui requisiti per una visualizzazione efficace, si rimanda a Bleeker et al. 2019a, anticipando in nota le principali modalità individuate dagli autori dell'articolo: “We can distinguish roughly five types of visualisation: alignment tables, parallel segmentation, synoptic viewers, variant graphs, and phylogenetic trees or ‘stemmata’” (p. 146).

informazioni che guidano il lettore nella comprensione del testo e della sua tradizione. Un esempio di strumento che consente di registrare annotazioni nelle due modalità è Digital Mappa*.

Molti strumenti permettono entrambi gli usi, quindi sia l'annotazione come fase di lavoro interno sia la condivisione e pubblicazione delle proprie annotazioni. Tra questi ci sono Pundit* e Annotation Studio*. In altri casi l'annotazione costituisce una fase specifica del lavoro editoriale in quanto assolve a un particolare obiettivo scientifico, per cui vengono creati degli strumenti dedicati. Recogito*, per esempio, è una piattaforma online per l'annotazione collaborativa delle informazioni geografiche all'interno di testi, immagini e database, che possono essere pubblicati e condivisi sotto forma di *open data* mediante la piattaforma. Infine, per l'annotazione di pagine web il W3C ha sviluppato il *Web Annotation Data Model*, che descrive il concetto di annotazione come “a rooted, directed graph that represents a relationship between resources” (Web Annotation DM). Lo scopo del modello è quello di fornire una descrizione standard che permetta di condividere annotazioni tra sistemi diversi, per condividerle o per preservarle nella migrazione tra dispositivi e piattaforme diverse.

1.3.3.4 Analisi del testo

Una volta realizzata l'edizione, il filologo può condurre un'ulteriore analisi del testo edito sotto diversi punti di vista, effettuando per esempio una lemmatizzazione o un'analisi morfo-sintattica, i cui esiti diventano parte integrante dello studio del testo. Negli ultimi anni, con la diffusione in ambito commerciale delle applicazioni di *natural language processing* (o linguistica computazionale), è aumentato il numero di software disponibili per realizzare automaticamente questo tipo di analisi, sia commerciali sia gratuiti. Questi ultimi sono spesso prodotti da centri di ricerca universitari o da altri enti culturali, come per esempio Stanza, la suite di strumenti informatici di NLP in Python, realizzata dall'Università di Stanford, Gatto* (Gestione degli Archivi Testuali del Tesoro delle Origini), software lessicografico sviluppato dall'Opera del Vocabolario Italiano per allestire il corpus TLIO (Tesoro della Lingua Italiana delle Origini) e TreeTagger*, strumento per la lemmatizzazione e per il

*part-of-speech tagging*³⁹ sviluppato presso l'Istituto per la Linguistica Computazionale dell'Università di Stoccarda.

Altri strumenti che rientrano in questa categoria sono concepiti non tanto per l'editore, quanto per l'utente finale di un'edizione scientifica digitale, ovvero il lettore. Si tratta di strumenti che, generando delle particolari visualizzazioni grafiche, consentono di leggere un testo “da lontano” (Morretti 2005). Uno degli strumenti informatici più famosi per condurre questo tipo di analisi del testo intesa come *distant reading* è Voyant Tools*, applicativo web pensato sia per un pubblico di ricercatori e studenti di *digital humanities* sia per un pubblico generico.

1.3.4 Gli strumenti di visualizzazione

La netta maggioranza degli strumenti di visualizzazione per le edizioni critiche è stata concepita per edizioni critiche in formato XML/TEI, in modo tale da potersi adattare a edizioni diverse prodotte in questo formato.

Tra gli strumenti di visualizzazione è possibile individuare un “grado zero”, ovvero i già menzionati fogli di stile, che consentono di convertire un documento XML/TEI in un formato per la pubblicazione dell'edizione a stampa o nel web. Un foglio di stile altro non è che un programma scritto in XSLT, che è, infatti, un linguaggio di programmazione di tipo dichiarativo. Nella tassonomia TaDiRAH l'attività che svolgono questi programmi è quella della conversione⁴⁰ da un formato a un altro, ma nel contesto della pratica ecdotica, si tratta di un'operazione che permette di pubblicare e visualizzare l'edizione.

Il consorzio TEI ha prodotto negli anni una ricca collezione di fogli di stile liberamente utilizzabili.⁴¹ Per facilitare ulteriormente la comunità dei propri utenti, ha realizzato anche OxGarage,⁴² un applicativo web per convertire documenti TEI in numerosi altri formati. Dal 2021 è disponibile una

³⁹ Con *part-of-speech tagging* (letteralmente marcatura delle parti del discorso) si intende una marcatura grammaticale condotta al fine di assegnare a ogni parola in un testo una parte del discorso, per esempio nomi, verbo, aggettivo, avverbio, ecc.

⁴⁰ Cfr. definizione di “converting” all'interno di TaDiRAH: <https://vocabs.dariah.eu/tadirah/converting>.

⁴¹ I fogli di stile elaborati dal consorzio TEI sono disponibili nel repository GitHub: <https://github.com/TEIC/Style-sheets>.

⁴² <https://oxgarage.tei-c.org/>.

nuova versione di OxGarage, chiamata TEIGarage*, che permette anche di accedere al servizio di conversione tramite API.

Appena sopra a questi strumenti di “grado zero” ci sono strumenti come TEIBoilerplate*, realizzato nel 2012 presso l’Università dell’Indiana, e CETEIcean*, una libreria JavaScript *open source* realizzata nel 2017. Questi strumenti consentono di visualizzare documenti in formato XML/TEI direttamente all’interno di un browser e di personalizzarne l’aspetto grafico. Mentre TEIBoilerplate mantiene intatto il documento XML/TEI, inserendolo in una pagina HTML, CETEIcean trasforma gli elementi del documento TEI in altrettanti *custom elements*⁴³ HTML, ai quali è possibile associare delle funzioni in JavaScript così da sviluppare delle funzionalità. Sia TEIBoilerplate sia CETEIcean sono stati concepiti per realizzare edizioni critiche digitali che da un punto di vista tecnologico siano fondate sugli standard del web (HTML, CSS e JavaScript) e che al contempo si presentino come delle pagine web statiche, così da risultare più durevoli e facili da mantenere nel tempo.

La necessità diffusasi tra gli studiosi che lavorano in XML/TEI di avere maggiore controllo sulla modalità di presentazione dei propri documenti e di poterli pubblicare sul web, ha portato la comunità TEI a sviluppare delle soluzioni. In primo luogo, alle linee guida TEI è stato aggiunto un insieme di elementi e attributi per fornire direttamente all’interno di un documento le istruzioni per la sua formattazione grafica in output, detto il *TEI Processing Model*.⁴⁴ Una volta che l’editore ha indicato come la propria edizione in XML/TEI deve essere resa graficamente, può servirsi come strumento di visualizzazione di TEIPublisher*, applicazione realizzata nel 2015 che consente di generare un sito web oppure di convertire i documenti in altri formati, quali PDF e ePUB.⁴⁵ In secondo luogo, il consorzio TEI insieme a un gruppo di università statunitensi ha realizzato TAPAS*,⁴⁶ uno strumento per l’archiviazione e condivisione online di documenti marcati in XML/TEI.

⁴³ <https://html.spec.whatwg.org/multipage/custom-elements.html#custom-element>.

⁴⁴ <https://www.tei-c.org/release/doc/tei-p5-doc/en/html/TD.html#TDPMPM>.

⁴⁵ Il formato ePUB (che sta per *electronic publication*) è uno standard aperto per e-book creato dall’International Digital Publishing Forum (IDPF). Negli anni è diventato il formato di riferimento per la pubblicazione di e-book e ha guadagnato popolarità in quanto è un formato basato su XML e libero, cioè non è un formato proprietario di aziende che producono e-reader e/o e-book. Per maggiori informazioni si rimanda alla seguente pagina nel sito del W3C: <https://www.w3.org/AudioVideo/ebook/>.

⁴⁶ Non è chiaro quando è stato avviato il progetto. Ad ogni modo nel 2013 era sicuramente attivo.

Gli strumenti per la visualizzazione di secondo e ultimo grado sono stati creati non solo per consentire agli editori di personalizzare le modalità di presentazione e fruizione della propria edizione scientifica digitale, ma anche per sperimentare come sfruttare le potenzialità del web al fine di valorizzare un'edizione scientifica digitale. A differenza degli strumenti di primo grado, dall'approccio più minimalista, questi strumenti sono dinamici e sofisticati e prevedono interfacce flessibili, funzionalità avanzate e diverse modalità di visualizzazione dei contenuti. Si tratta di strumenti più complessi e costosi da realizzare, che spesso vengono creati nell'ambito di un particolare progetto di ricerca, per poi venire messi a disposizione della comunità scientifica in un secondo momento. Un esempio è il software *open source* e *client only*⁴⁷ EVT (Edition Visualization Technology)*. Nato come strumento di visualizzazione *ad hoc* per l'edizione digitale del Vercelli Book,⁴⁸ EVT è diventato adattabile a edizioni diverse grazie alla collaborazione con altri progetti scientifici, in particolare il Codice Pelavicino Digitale,⁴⁹ che se ne sono serviti e che hanno contribuito allo sviluppo. Un altro strumento di "secondo grado" molto noto è Versioning Machine*,⁵⁰ framework e interfaccia web che permette di visualizzare e confrontare molteplici versioni di un medesimo testo (per esempio i diversi testimoni di un'opera) codificati in XML/TEI.

1.3.4.1 Strumenti per la visualizzazione di immagini

Moltissime edizioni critiche digitali offrono oltre al testo critico e le trascrizioni dei testimoni le riproduzioni fotografiche dei testimoni. Diversi strumenti di visualizzazione di documenti XML/TEI supportano anche le immagini, come per esempio EVT e TEIPublisher. Tuttavia data la rilevanza e la diffusione di IIF, conviene spendere qualche riga per presentarne il funzionamento da un punto di vista tecnico e offrire degli esempi di applicativi nati appositamente per la visualizzazione di immagini formattate in base a questo standard. Inoltre, i visualizzatori (*viewer*) compatibili con IIF sono spesso facilmente integrabili in altri strumenti e per questo vengono largamente utilizzati.

⁴⁷ L'espressione *client only* (letteralmente "solo cliente") individua gli applicativi che per funzionare non necessitano di rapportarsi con un *server*. Una volta che l'applicativo viene caricato nel *browser* dell'utente (cioè dal cliente), tutte le sue funzionalità non richiedono la comunicazione con un server per poter essere eseguite.

⁴⁸ Sito web dell'edizione digitale del Vercelli Book: <http://vbd.humnet.unipi.it/beta2/>.

⁴⁹ Sito web dell'edizione digitale del Codice Pelavicino: <https://pelavicino.labcd.unipi.it/>.

⁵⁰ Versioning Machine è stato ideato nel 2000 da Susan Schreibman e ricevendo negli anni diversi finanziamenti da parte di enti culturali e università statunitensi. Attualmente il software è alla sua quinta versione, pubblicata a gennaio del 2016.

La comunità IIF ha sviluppato quattro diverse API per rispondere ai diversi bisogni degli utenti. Tra queste le due API più interessanti rispetto agli obiettivi della presente rassegna sono ‘Image API’ e ‘Presentation API’. La prima consente di costruire degli URI parametrici, mediante i quali recuperare una determinata area all’interno dell’immagine e di definire per ciascuna area come può essere manipolata visivamente. La seconda API permette di condividere immagini accompagnate da metadati descrittivi e da tutte le informazioni da presentare all’utente, contenute in un documento JSON detto “presentation manifest”. Le altre due API ausiliarie sono ‘Authentication API’, mediante la quale è possibile regolare l’accesso alle immagini, richiedendo agli utenti di autenticarsi, e ‘Content Search API’, che permette di effettuare delle ricerche all’interno delle informazioni legate all’immagine.

Attualmente sono disponibili numerosi visualizzatori che supportano lo standard IIF. Tra i visualizzatori più affermati vi sono: OpenSeadragon* (integrato, tra l’altro, in EVT) e Leaflet-IIF,⁵¹ che consentono di visualizzare l’immagine ad alti livelli di zoom, ma hanno come limite quello di presentare una sola immagine alla volta e senza i relativi metadati; Mirador* e UniversalViewer*, a differenza dei primi due, mostrano i metadati per la risorsa nella sua interezza e consentono altre funzionalità, quali il download e l’aggiunta di annotazioni al file *manifest* delle immagini.

Un problema comune alla presentazione dei testimoni attraverso le loro riproduzioni fotografiche è che appiattiscono in 2D oggetti tridimensionali, non permettendo agli studiosi di avere delle informazioni precise su come i codici sono stati fascicolati, salvo recuperando tali informazioni dai cataloghi. Per ovviare a questo problema sono stati sviluppati degli strumenti informatici che permettono di generare delle visualizzazioni più “tridimensionali” dei testimoni nei loro supporti, per integrare poi tali visualizzazioni all’interno delle edizioni critiche, come per esempio VisColl*⁵² e Manicule*.

⁵¹ Sito web di Leaflet-IIF: <https://github.com/mejackreed/Leaflet-IIF>.

⁵² Il nome dell’applicativo sta per *Collation Visualization*, dove ‘collation’ rimanda non al confronto tra i testimoni ma alla ‘collazione fisica’, intesa come il modo in cui i fogli di un manoscritto sono disposti e legati insieme.

1.3.4.2 Pubblicazione in altri formati

Per completare la rassegna degli strumenti di visualizzazione, occorre fornire qualche riferimento anche su quelli che consentono di pubblicare e visualizzare i dati alla base di un'edizione in formati particolari. Negli ultimi anni, per esempio, ci sono state delle sperimentazioni nel campo del web semantico, che hanno portato alla pubblicazione dei dati prodotti contestualmente alla realizzazione di un'edizione digitale come *linked open data*. Il senso di questo tipo di pubblicazioni è quello del riuso dei dati stessi. Sono, quindi, pubblicazioni destinate a un pubblico molto preciso, che non vuole “leggere” l'edizione, ma servirsi dei dati editi da altri studiosi per i propri scopi di ricerca. LIFT (Linked data from TEI)*, per esempio, è uno strumento che permette di trasformare un'edizione in XML/TEI in un *knowledge graph*,⁵³ pronto per la pubblicazione sul web come *linked open data*.

1.3.5 Gli strumenti di produzione e visualizzazione (o strumenti misti)

Gli strumenti di produzione e di sola visualizzazione sono spesso realizzati da un singolo sviluppatore o da un gruppo ristretto di sviluppatori, che lavorano nel medesimo centro di ricerca o dipartimento. Gli strumenti misti, invece, tendono a essere sviluppati in contesti istituzionali più ampi e articolati, sia perché sono degli strumenti complessi e costosi da sviluppare, sia perché molto spesso nascono come risposta personalizzata alle esigenze di un progetto di ricerca di grandi dimensioni.

Il primo strumento misto che si è imposto nella storia della filologia digitale e che tuttora rappresenta un riferimento nella comunità scientifica è il già menzionato TUSTEP*. Ideato nel 1966 presso l'Università di Tübingen da Wilhelm Ott e Kuno Schälkle, continua a essere aggiornato e sviluppato. TUSTEP consente di effettuare diverse operazioni, tra cui: trascrivere, collazionare, costituire apparati critici, generare indici e concordanze. Da un punto di vista tecnico, è costituito da diversi moduli, ciascuno ideato per eseguire un'operazione specifica, che l'utente può combinare a piacere in modo da creare flussi di lavoro personalizzati che rispondano alle proprie esigenze. Per utilizzare il programma è necessario lanciare dei comandi (in lingua tedesca o inglese) attraverso un'interfaccia a riga di comando non molto *user-friendly*.

⁵³ Un *knowledge graph*, noto anche come ‘rete semantica’, rappresenta una rete di enti del mondo reale - oggetti, eventi, persone, concetti, ecc. - e illustra i loro rapporti reciproci. Si tratta, in altre parole, di un altro modo di organizzare la conoscenza, strutturando i dati partendo dalle loro relazioni.

Gli strumenti misti sviluppati in anni più recenti sono la maggior parte delle volte degli ambienti online per il lavoro collaborativo. Alcuni esempi sono: eLaborate* che permette di caricare le scansioni dei documenti, trascrivere e annotare i testi e pubblicare i risultati online; TextGrid* un applicativo per l'*editing* collaborativo o individuale e la pubblicazione di risorse basate su TEI; TextualCommunities*,⁵⁴ un ambiente online per la trascrizione collaborativa e la collazione automatica di testi, che permette anche di usare i materiali editi come base per costruire un sito web mediante un'API.

1.3.6 Piattaforme per la pubblicazione e infrastrutture istituzionali

Per lo studioso che desidera pubblicare online la propria edizione, un'alternativa all'utilizzo di strumenti dedicati consiste nel rivolgersi a delle piattaforme per la pubblicazione. Queste piattaforme possono essere parte di riviste scientifiche, come nel caso delle Micro Editions della già citata Scholarly Editing, oppure di una casa editrice. Il vantaggio nell'utilizzare queste piattaforme è che molto spesso i loro responsabili offrono a titolo gratuito supporto agli editori, guidandoli nell'allestimento dei materiali in formato elettronico oppure occupandosene in prima persona.

Un esempio di piattaforma per la pubblicazione di edizioni scientifiche elettroniche è Inkless Editions,⁵⁵ creata da Peter Robinson nel 2000 con il nome di Scholarly Digital Editions (SDE). Specializzata nella pubblicazione di testi e manoscritti medievali, di recente ha curato la seconda versione della *Commedia* di Dante edita da Prue Shaw.⁵⁶

Altri esempi sono: Rotunda, la piattaforma di pubblicazione dell'Università della Virginia; Editionenportal Thüringen;⁵⁷ le edizioni elettroniche della rivista online Romantic Circles,⁵⁸ specializzata negli studi sulla cultura e letteratura del periodo romantico.

In chiusura della rassegna si vuol porre l'attenzione di chi legge sulla rilevanza delle infrastrutture universitarie e istituzionali nel campo della filologia digitale e dello sviluppo di software dedicati.

⁵⁴ Textual Communities è stato sviluppato presso l'Università del Saskatchewan con il supporto di fondi nazionali canadesi. Tra i suoi autori vi è Peter Robinson. Il sito del software, infatti, ospita anche tra le "communities" di esempio una parte delle Canterbury Tales edita da Robinson.

⁵⁵ <http://www.inklesseditions.com/>.

⁵⁶ <https://dantecommedia.it/>.

⁵⁷ <http://www.editionenportal.de/>.

⁵⁸ <https://romantic-circles.org/editions>.

Diversi strumenti, infatti, sono nati in seno alla realizzazione di progetti di *digital humanities* di ampio respiro, intrapresi da infrastrutture che comprendono centri di ricerca, università, biblioteche, archivi e simili. Laddove il singolo dipartimento o la singola biblioteca non dispone di fondi a sufficienza, l'unione con altri e di conseguenza l'unione dei fondi, consente di sviluppare progetti più ambiziosi ed elaborati. Alcuni esempi di infrastrutture che hanno avuto un impatto significativo sullo sviluppo di strumenti e di edizioni digitali sono:

- la società e-editiones,⁵⁹ basata in Svizzera e di cui fa parte anche l'azienda eXist Solutions, focalizzata sullo sviluppo di software *open source* per le edizioni digitali;
- la statunitense NINES (Networked Infrastructure for Nineteenth-Century Electronic Scholarship),⁶⁰ dismessa da diversi anni, ma nota per essere stata il contesto istituzionale all'interno del quale è stato sviluppato Juxta;
- NIE – INE (Nationale Infrastruktur für Editionen / Infrastructure nationale pour les éditions),⁶¹ basata in Svizzera, avviata nel 2016 e chiusa nel 2019.

1.4 Analisi del panorama attuale

Sulla base dello stato dell'arte tracciato nella precedente sezione, presentiamo qui le principali tendenze e gli aspetti più problematici individuati.

Tra le tendenze emerge, in primo luogo, l'utilizzo di TEI come standard di riferimento anche per la produzione di software per la pratica ecdotica. Di 57 strumenti analizzati 24 supportano XML/TEI come formato di input e/o di output. Questa tendenza può essere spiegata piuttosto facilmente facendo riferimento a due principali ragioni: da un lato, TEI è basata su XML, tecnologia standard del W3C, utilizzata nello sviluppo di siti e risorse web, dall'altro il consorzio TEI ha elaborato una dettagliata documentazione tecnica e una serie di strumenti, dai fogli di stile agli schemi in diversi formati, che semplificano il lavoro di sviluppo degli strumenti informatici.

⁵⁹ <https://www.e-editiones.org/>.

⁶⁰ <https://nines.org/about/what-is-nines/>.

⁶¹ Il sito web ufficiale dell'infrastruttura non è più disponibile.

La seconda tendenza rilevata è il progressivo superamento del modello “sorgente-*output*” per la realizzazione di edizioni critiche digitali, sostituito da un modello più generico e che in parte ingloba quello precedente, che proponiamo di chiamare semplicemente “dati-visualizzazione”. Entrambi i modelli si fondano su una separazione tra i contenuti e la loro presentazione, ma la differenza risiede nelle architetture dei software e, di conseguenza, nelle modalità in cui permettono di realizzare edizioni critiche digitali. Il modello “sorgente-*output*” prevede una divisione tra la fase di produzione dei dati alla base dell’edizione e la fase di visualizzazione dell’edizione. Queste due fasi sono spesso condotte da due o più applicativi diversi ma, come si è visto, possono essere eseguite all’interno di un unico applicativo. Quando si lavora in un unico applicativo, non è più necessario generare un file sorgente per poter passare alla fase successiva. A ciò si aggiunge che non è detto che i dati prodotti dall’utente siano salvati nel server come file, ma potrebbero essere salvati in un database come altre strutture di dati, per esempio come oggetti in formato JSON. Inoltre, se si utilizza uno strumento con un’interfaccia WYSIWYG,⁶² il modo in cui sono strutturati i dati prodotti è invisibile all’utente, rimane nascosto nel server e nella logica dell’applicativo. Il concetto di “file sorgente”, dunque, diventa superfluo in questa tipologia di strumenti. Ciò non esclude che gli applicativi permettano di esportare i dati in documenti di diverso formato, quali XML/TEI e PDF. Tuttavia, ai fini dell’edizione, è sufficiente che i dati vengano visualizzati nell’interfaccia. Per questo motivo si propone il nome “dati-visualizzazione” per il nuovo modello.

L’ultima tendenza riguarda le modalità in cui gli strumenti vengono ideati e sviluppati. Nella maggior parte dei casi si tratta di strumenti che nascono all’interno di un progetto di ricerca, condotto da un singolo studioso oppure in *équipe*, per rispondere alle esigenze di un particolare progetto di edizione. Una volta riscontrata l’efficacia dello strumento, lo studioso o il gruppo di studiosi decide di metterlo a disposizione della comunità scientifica, come software a pagamento, gratuito o *open source*. Essendo strumenti nati in un contesto accademico, solitamente vengono sviluppati direttamente da studiosi che conoscono a fondo il problema che lo strumento deve risolvere oppure sotto la loro supervisione. Inoltre, lo sviluppo e il mantenimento di questi strumenti dipende perlopiù da fondi di natura precaria, perché a tempo determinato e perché vincolati al raggiungimento di

⁶² Con WYSIWYG (*What You See Is What You Get*) si intende un’interfaccia che permette di visualizzare su schermo il risultato finito.

specifici obiettivi scientifici. Nella minor parte dei casi, invece, sono strumenti che nascono per rivolgersi a un pubblico ampio e variegato e adattarsi a diversi progetti di edizione, realizzati in ambito accademico oppure da privati a scopo commerciale. Per semplificare, le due tipologie di strumenti si possono categorizzare come strumenti generalizzati e strumenti generici. Un esempio di strumento generalizzato e che oramai può essere considerato generico, in quanto adottato e adattato da diversi progetti, è EVT (vd. 1.3.4).

Nell'analisi sono stati omessi tutti gli applicativi che sono stati creati *ad hoc* per un particolare progetto di edizione, senza venire poi generalizzati. Riprendendo le categorie proposte in Pierazzo 2019, si può affermare che nella presente analisi sono stati considerati solo gli strumenti per realizzare “edizioni prêt-à-porter”, tralasciando quelli per le edizioni “haute couture”. Tuttavia, il numero di progetti di edizione che si basano su strumenti costruiti *ad hoc* sembra superare nettamente il numero delle edizioni che si servono di strumenti pronti all'uso.

A riguardo abbiamo condotto un'analisi statistica, utilizzando i dati contenuti nel *Catalogue of Digital Editions*. Su 320 edizioni totali, il catalogo offre per 178 record,⁶³ all'interno del campo denominato “infrastructure”, un elenco dei linguaggi di programmazione, delle tecnologie e degli strumenti utilizzati per lo sviluppo dell'edizione digitale. Si può individuare in 68 record il riferimento a uno strumento generico o generalizzato per la pratica ecdotica (circa il 35% dei record per cui i dati sull'infrastruttura sono disponibili), adottando una definizione molto larga, che tenga in conto anche database e framework per la realizzazione di siti web basati su XML come eXist-db,⁶⁴ xMod⁶⁵ e Kiln,⁶⁶ e altri strumenti decisamente generici, quali il già citato Oxygen.

Suddividendo per anno il numero di edizioni prodotte con uno strumento generico o generalizzato per la pratica ecdotica ed escludendo le edizioni per cui non si dispone della data di inizio della loro realizzazione, non si nota una tendenza in crescita. Al contrario, si osserva un andamento bene o male costante, con un picco raggiunto tra il 2014 e il 2015.

⁶³ Per 15 record non è stato possibile reperire l'informazione, mentre in 127 record l'informazione non è ancora stata inserita nel catalogo.

⁶⁴ <http://exist-db.org/exist/apps/homepage/index.html>.

⁶⁵ <https://www.tei-c.org/Vault/MembersMeetings/2008/xmod/index.html>.

⁶⁶ <https://github.com/kcl-ddh/kiln>.

Numero di edizioni per anno prodotte con strumenti per la pratica ecdotica

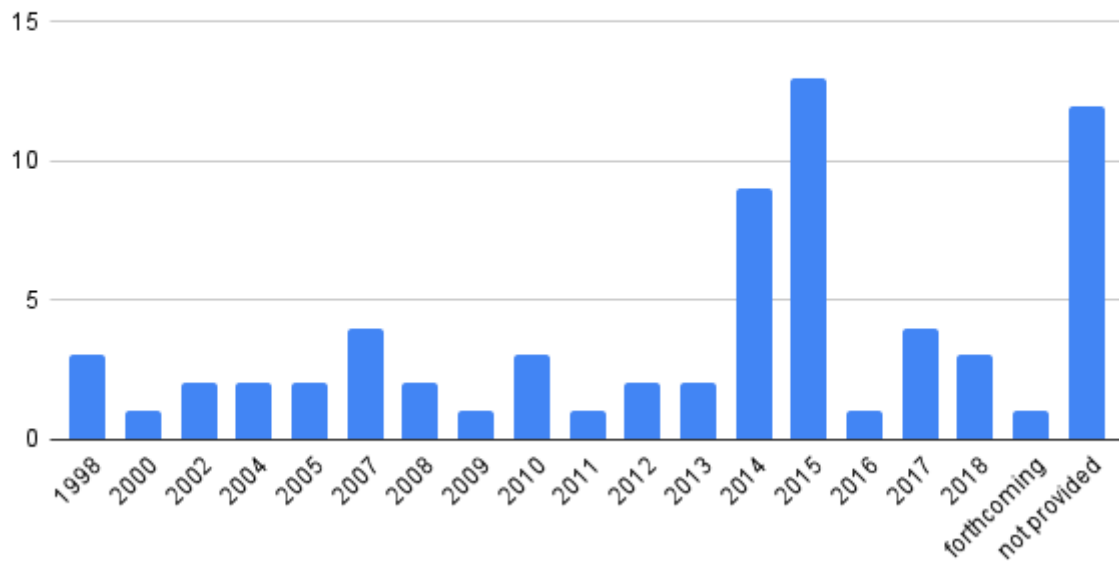


Figura 1.7 Grafico che rappresenta il numero di edizioni scientifiche digitali realizzate ogni anno mediante l'utilizzo di uno strumento generale o generalizzato per la filologia digitale.

In questo grafico, così come in quello sull'adozione di XML/TEI presentato nella sezione 1.3, è molto probabile che ci siano dei *bias* dovuti alle tempistiche e modalità in cui il catalogo stesso è stato compilato. Ad ogni modo, da questa semplice analisi emerge piuttosto chiaramente che chi realizza un'edizione digitale tende a sviluppare da sé i propri strumenti piuttosto che affidarsi a quelli esistenti. Sorge spontaneo domandarsi come mai.

In un articolo del 2005 dal titolo *Current issues in making digital editions of medieval texts—or, do electronic scholarly editions have a future?* Peter Robinson constatava la presenza di una crisi nel campo delle edizioni critiche digitali, dopo i primi dieci anni di sperimentazioni, attribuendo la principale causa alla mancanza di *tools* usabili.⁶⁷ A 18 anni di distanza, il numero di strumenti è decisamente cresciuto: ricordiamo che il portale TAPoR ne raccoglie più di 1600, mentre solo all'interno di questo lavoro di tesi ne abbiamo analizzati 57. Si può presupporre, dunque, che non si tratti di un problema di mancanza di strumenti, perlomeno non da un punto di vista quantitativo. Ci sono molti

⁶⁷ "If digital editions are so manifestly superior, then why indeed are we in the state of affairs described above? Why are so many scholars, and so many scholarly projects, still making print editions? I suggest that the answer to this lies almost wholly in the second aspect to this problem: the availability of usable tools." (Robinson 2005, p. 5)

più strumenti informatici a disposizione, ma non sono ancora ritenuti “usabili”. In un punto dell’articolo Robinson sembra alludere proprio a questo problema:

“But even if he or she can afford a proprietary system (and that is a large if), our scholar is still going to need a high level of skill to set up and configure the software, and then customize the interface for his or her data. And even then, he or she may still find **that the software cannot do everything required of it**” (Robinson 2005, p. 7, mia enfasi).

Nell’articolo del 2019 dal titolo *What future for digital scholarly editions? From Haute Couture to Prêt-à-Porter*, già citato sopra, Elena Pierazzo mette a confronto le edizioni realizzate con sistemi *ad hoc* e quelle prodotte con strumenti generici/generalizzati, elencando per entrambe i principali vantaggi e svantaggi, che potrebbero indurre uno studioso a preferire gli uni o gli altri. Riportiamo qui in forma schematica le principali differenze individuate da Pierazzo:

1. **i costi di implementazione e manutenzione:** le edizioni “haute couture” sono più costose sia da sviluppare sia da mantenere nel tempo, mentre gli strumenti pronti all’uso consentono di contenere le spese per lo sviluppo e la manutenzione;
2. **il grado di flessibilità e personalizzazione:** gli strumenti *ad hoc* sono realizzati “su misura”, quindi soddisfano le esigenze tecniche e scientifiche del progetto di edizione, i sistemi generici/generalizzati, invece, tendenzialmente offrono un grado inferiore di personalizzazione e un numero più limitato di funzionalità;⁶⁸
3. **il livello di competenze tecniche necessarie:** per sopperire al ridotto grado di personalizzazione di un sistema generico/generalizzato è possibile intervenire sul codice sorgente del software, se disponibile. In questo modo si dispone del pieno controllo sulle funzionalità e sulle modalità di visualizzazione. Tuttavia, per poter intervenire sul codice sono necessarie delle competenze tecniche avanzate, che difficilmente un filologo possiede. Quando si realizza un sistema *ad hoc* molto spesso il lavoro di sviluppo viene affidato a dei tecnici, senza che allo studioso siano richieste particolari competenze informatiche;
4. **fondi e obiettivi scientifici:** spesso i fondi elargiti per realizzare un’edizione digitale vengono assegnati a progetti di edizione che mirano a realizzare strumenti informatici innovativi

⁶⁸ La limitata possibilità di personalizzare gli strumenti generici/generalizzati dipende dal fatto che: “agiscono come collane editoriali, vale a dire che offrono una formula comune con minimi margini di flessibilità” (Mancinelli e Pierazzo 2020).

e che quindi sviluppano dei nuovi strumenti *ad hoc*. Di contro, i giovani ricercatori potrebbero essere maggiormente interessati a pubblicare i risultati dei propri studi come edizioni digitali, servendosi di strumenti generici/generalizzati (per via anche dei costi ridotti). Tuttavia, potrebbero rinunciare all'idea, in quanto non dispongono delle competenze tecniche necessarie o dei fondi per poter avere un supporto informatico.

Le differenze tra edizioni “haute couture” e “prêt-à-porter” descritte da Pierazzo aiutano a mettere a fuoco delle problematiche ricorrenti dell'*editing* filologico in digitale. Concentrandosi sugli strumenti generici/generalizzati, occorre chiarire come mai gli strumenti disponibili continuano a essere percepiti come non usabili da buona parte della comunità scientifica.

Prima di illustrare nel dettaglio le ragioni individuate, è opportuno specificare a quale “comunità scientifica” si fa riferimento. Ad avviso di chi scrive il distinguo tra filologi “digitali” e filologi “non digitali” non è fruttuoso per la ricerca, né sostenibile per il futuro. Si tratta di una distinzione utile per specificare le aspettative riguardo alle competenze e le modalità di lavoro di entrambe le categorie. Tuttavia, lo sviluppo di strumenti per la pratica ecdotica deve essere rivolto a entrambe le categorie, considerando tutti i filologi come potenziali utenti, a prescindere dalle loro competenze informatiche. Per questo motivo, nel presente lavoro di tesi con “comunità scientifica” si considerano in generale i filologi.

1.4.1 Le tre ‘I’

I principali problemi che rendono uno strumento informatico per la pratica ecdotica non usabile, emersi dalla presente analisi sono tre e possono essere definiti come le tre ‘I’: instabilità, inaccessibilità e inadeguatezza.

1.4.1.1 Instabilità

L’instabilità riguarda sia le tecnologie digitali in generale sia, in particolare, le modalità in cui gli strumenti informatici per la filologia digitale vengono sviluppati.

Partendo dal generale, si può constatare come negli anni si siano alternate alcune tendenze nella realizzazione di siti web, che si sono diffuse massicciamente e molto rapidamente, ma che poi sono cadute. Si pensi per esempio a Flash di Adobe, utilizzato per lo sviluppo di siti e giochi online, che è

stato progressivamente sostituito dagli standard aperti, quali HTML e WebGL. L'utilizzo di standard aperti, in particolare quelli sviluppati e mantenuti dal W3C, permette di sviluppare strumenti più stabili nel tempo, in quanto garantiscono una maggiore longevità e una più facile manutenibilità. Un altro aspetto che tende a variare molto rapidamente sul web è la grafica. Questi cambiamenti dipendono sia da fattori estetici e commerciali sia dalla diffusione sul mercato di nuove tecnologie e di nuovi dispositivi. La diffusione dello *smartphone*, per esempio, ha richiesto che nel campo della progettazione grafica per il web si adottassero dei nuovi accorgimenti, arrivando anche all'approccio *mobile first*, per cui si progetta l'aspetto grafico di un sito o di un'app partendo dalla sua resa in un dispositivo mobile e poi su *desktop*. L'attenzione all'aspetto grafico di un sito web o di un applicativo non è di secondaria importanza. Al giorno d'oggi, data l'elevata competizione tra i numerosi servizi offerti sul web, siamo abituati a prodotti molto curati anche nell'aspetto grafico. Uno strumento poco curato nella grafica o dall'*appeal* vecchio potrebbe sembrare meno affidabile di altri. Inoltre la gradevolezza estetica di uno strumento influisce anche sulla sua usabilità, come riassume bene il titolo dell'articolo di Tractinsky, Katz e Ikar (2000): *What is beautiful is usable*. Uno strumento per la pratica ecdotica, dunque, per essere usabile deve essere mantenuto e aggiornato rispetto alle ultime tendenze tecnologiche e, idealmente, anche del *web design*.

Passando all'instabilità nel particolare, si può rilevare come gli strumenti per la pratica ecdotica siano sviluppati in modo fortemente instabile. Prima di tutto si tratta di un'instabilità di natura economica. Come già illustrato, nel contesto di università e altri enti di ricerca lo sviluppo viene spesso portato avanti grazie a fondi di natura precaria. Inoltre l'applicativo può appartenere a un ente (centro di ricerca, dipartimento, ecc.) oppure a un determinato sviluppatore o gruppo di sviluppatori. Nel primo caso, le persone che materialmente sviluppano l'applicativo tendono ad alternarsi (sempre a causa dei contratti e dei finanziamenti precari), per cui menti diverse mettono mano sul codice, portando a delle discontinuità sull'organizzazione logica dell'applicativo e sulle modalità di scrittura del software.⁶⁹ Nel secondo caso, invece, la mente o le menti sono le stesse, ma il lavoro di sviluppo è

⁶⁹ Anche nella scrittura di codice ci sono 'stili' diversi, che dipendono dalla formazione e dal gusto personale dello sviluppatore. Bisogna considerare anche le difficoltà che si riscontrano quando si lavora su un codice scritto da altri. Il primo periodo molto spesso è necessaria una fase di lettura e interpretazione del codice già presente, prima di poter iniziare a lavorare.

comunque condotto con discontinuità. Molto spesso gli sviluppatori, se sono ricercatori o professori, non hanno modo di dedicarsi allo sviluppo in maniera continuativa, perché sono presi dall'insegnamento o da altre ricerche. Altro fenomeno frequente è che cambiando l'affiliazione dello sviluppatore, cambino anche la collocazione e i requisiti tecnici dell'applicativo, che viene ospitato su server differenti o distribuito secondo nuove modalità.

Gli applicativi commerciali tendono a essere più stabili. Tuttavia, la loro stabilità dipende dal loro successo e dalla capacità di mantenersi aggiornati, da un punto di vista tecnologico e grafico, e 'appetibili' rispetto agli interessi di ricerca nel mondo accademico.

1.4.1.2 Inaccessibilità

La seconda 'T' corrisponde a 'inaccessibilità'. L'inaccessibilità di uno strumento dipende spesso dalla sua instabilità tecnologica, per esempio, se l'applicativo non è più compatibile con i moderni browser o con i sistemi operativi più diffusi oppure se l'interfaccia dell'applicativo risulta inadatta rispetto agli attuali dispositivi e alle loro caratteristiche tecniche (quali la risoluzione dello schermo). Il concetto di inaccessibilità può essere anche inteso come incomprendibilità. Uno strumento risulta inaccessibile quando la sua interfaccia grafica non comunica efficacemente all'utente il modo in cui può essere usato.

Altri aspetti che rendono un applicativo inaccessibile sono la mancanza di documentazione e l'assenza di un supporto tecnico. Se lo sviluppo dello strumento è instabile, difficilmente metterà a disposizione dei propri utenti un servizio di supporto tecnico. Gli applicativi a pagamento offrono sempre sia la documentazione che il supporto tecnico, ma possono risultare inaccessibili per i costi. Diversi applicativi offrono dei periodi di prova oppure dei piani gratuiti, che consentono un uso adatto per progetti di edizione di piccola-media entità. Per esempio Transkribus offre gratuitamente la trascrizione di 3000 pagine stampate e di 500 pagine scritte a mano. Nel caso dei software *open source*, invece, l'inaccessibilità è data dalla mancanza di documentazione tecnica e dal grado di leggibilità del codice sorgente.⁷⁰

⁷⁰ Ci sono diversi principi e buone pratiche che possono essere seguiti per rendere il codice più leggibile. Se ne citano alcuni come esempio: scrivere sinteticamente, suddividere il codice in più file diversi, ciascuno contenente poche righe di codice; commentare il codice; utilizzare nel codice gli stessi termini utilizzati nell'interfaccia grafica.

1.4.1.3 Inadeguatezza

Il terzo problema degli strumenti per la pratica ecdotica è l'inadeguatezza, la terza e ultima 'I', intesa come una mancata corrispondenza tra le caratteristiche dello strumento e le necessità del filologo. Riprendendo le parole di Robinson (2005) l'inadeguatezza si manifesta quando "the software cannot do everything required of it" e può dipendere da aspetti diversi, tra cui:

- il formato dei dati che l'applicativo prevede in input o in output;
- il modo in cui i dati vengono elaborati dall'applicativo;
- le funzionalità offerte dall'applicativo;
- le modalità di presentazione dei dati, se prevista.

Questi requisiti tecnici, funzionali, grafici e di organizzazione logica del software solitamente vengono stabiliti in fase di progettazione sulla base delle indicazioni fornite dal cliente che ha commissionato la realizzazione dello strumento informatico. Nel caso di uno strumento informatico per la filologia digitale generico chi determina i requisiti? Su quali basi?

Facendo riferimento alla distinzione tra strumenti generalizzati e strumenti generici, già illustrata nella precedente sezione, si può dire che quelli generalizzati solitamente sono ancorati a un modello dei dati e a un insieme di specifiche che in origine rifletteva le esigenze scientifiche di un particolare progetto di edizione digitale, e che solo in un secondo momento sono stati rivisti per essere adattati ad altre edizioni:

"What most tools have in common is the fact [that] their development seems to have been generated from data models (how a file is encoded), from specific projects, and from specific requests made by early users, more than from an effort to assess the editorial models needed by the scholarly community." (Pierazzo 2019, p. 9, mia enfasi)

Per questo motivo uno strumento generalizzato tende a presentare delle rigidità. Nel caso degli strumenti generici, invece, i requisiti vengono stabiliti dallo sviluppatore (o gruppo di sviluppatori). Se lo sviluppatore si occupa anche di filologia digitale, si presuppone che conosca a fondo i problemi

e le caratteristiche del dominio,⁷¹ cioè del campo in cui lo strumento deve essere applicato. Viceversa, se lo sviluppatore non possiede queste competenze scientifiche, se dunque non è un “esperto del dominio”,⁷² deve riuscire a individuare le caratteristiche del dominio e le esigenze dei potenziali utenti, possibilmente confrontandosi con degli esperti. Tuttavia, gli strumenti generici presentano spesso il difetto di non riuscire a soddisfare a pieno le esigenze degli utenti o di risultare carenti su aspetti che alcuni studiosi ritengono di fondamentale importanza.

Il problema di fondo risiede nella conoscenza e nella descrizione del dominio, in cui gli strumenti informatici devono essere applicati. Per capire meglio che cosa si intende, si presenta come esempio la progettazione di una piattaforma per l'*e-commerce*, la vendita online di prodotti. Una medesima piattaforma può essere facilmente usata da tante attività ed esercizi commerciali diversi, che vendono prodotti differenti. Molte delle funzionalità e dei requisiti che la piattaforma deve possedere, infatti, sono gli stessi a prescindere dall'attività commerciale, come il caricamento dei prodotti sulla piattaforma, la verifica delle quantità di prodotti disponibili in magazzino, l'acquisto da parte del cliente, ecc. Inoltre, la vendita di prodotti online è regolamentata dalla legge, per cui chi progetta la piattaforma, non solo riesce a individuare e a comprendere più facilmente quali sono i requisiti, ma dispone anche di indicazioni comuni, stabilite da altri, su quali sono le regole da rispettare.

Il campo della filologia digitale è un campo in divenire, connotato da una forte tendenza alla sperimentazione. Di conseguenza, la progettazione di uno strumento informatico per la filologia digitale risulta difficoltosa per due motivi: 1) si tratta di un campo molto complesso e poco noto, che difficilmente uno sviluppatore informatico conosce a meno che non abbia condotto degli studi specifici; 2) si può dire che tendenzialmente manchino dei riferimenti chiari, stabiliti dalla comunità scientifica e condivisi su quali debbano essere i requisiti che guidino lo sviluppo di strumenti informatici generici per la filologia digitale.

⁷¹ Per spiegare più approfonditamente il concetto di dominio (in inglese *domain*), così come è utilizzato nel campo dell'ingegneria del software, si riporta la definizione del *Systems and Software Engineering Vocabulary*: “area of knowledge or activity characterized by a set of concepts and terminology understood by practitioners in that area”.

⁷² Nell'ingegneria del software con “esperto del dominio” (in inglese *domain expert*) si intende: “individual who is intimately familiar with the domain and can provide detailed information to the domain engineers” *Systems and Software Engineering Vocabulary*).

Nella letteratura scientifica, questa lacuna è stata già individuata diverse volte negli ultimi anni da più studiosi.

“Despite increasing theoretical awareness, **the tools of representation and analysis produced so far have not satisfactorily addressed the problem of the specificity and complexity of the cultural and literary studies domain.** In fact, the intellectual investment in the definition of new models and languages for the formal representation and processing of complex cultural objects has been rather low. **Most commonly we have inherited and applied models and languages developed in computer science for different domains and necessities.**” (Ciotti 2014, mia enfasi)

“**Scholarly editing has been a leader in the use of computing in the humanities and yet many of its tools are borrowed or adapted from applications that were designed for other purposes.** This is good to the extent that adaptations built on the shoulders of previous efforts have saved time and provided desired results, but **it is bad when off--the--shelf solutions result in compromises, not exactly what the scholarly editor wanted,** even though they might help a project meet a funding deadline. The call for tools designed and developed for specific purposes in scholarly editing has been already made by many. [...] However, while varieties of interesting projects sporting new tools and designs are necessary to explore the options and possibilities that can be developed, **we lack a comprehensive and thoughtful set of principles to define the characteristics that our tools, content, displays, and environments do or should incorporate.**” (Shillingsburg 2013, mia enfasi)

“Although the digital turn of textual scholarship is nowadays a reality and many advancements have been made in encoding and visualizing textual resources, **flexible and shared models in the construction of tools for scholarly editing are still missing. This lack typically leads to the development of ad hoc – i.e. not reusable software.**” (Del Grosso et al. 2017, mia enfasi)

In realtà, negli ultimi anni sono stati prodotti alcuni modelli allo scopo di guidare lo sviluppo degli strumenti informatici per la filologia digitale.

Un “modello” più generico, che però non si è imposto nella comunità scientifica (sulla base di quanto chi scrive è riuscita a rilevare nella letteratura), è stato elaborato nel contesto del progetto Humanities Research Infrastructure and Tools (HRIT), del Textual Studies and Digital Humanities della Loyola University di Chicago, coordinato da Peter Shillingsburg. Il progetto HRIT ha elaborato un’infrastruttura di strumenti informatici per la realizzazione di archivi testuali elettronici e di edizioni digitali. Nel sito del progetto viene offerto un vero e proprio manuale per gli sviluppatori,⁷³ che illustra come utilizzare e sviluppare ulteriormente i *tool* che fanno parte dell’infrastruttura, ma, al contempo, illustra i principi che hanno guidato la realizzazione dell’infrastruttura stessa. Il progetto HRIT, infatti, non ha elaborato un vero e proprio “modello” ma un insieme di principi

⁷³ <https://sites.google.com/a/ctsdh.luc.edu/hrit-intranet/documentation/handbook>.

generali “for the construction of tools and environments for developing, maintaining, and publishing scholarly textual archives / editions / commentary and pedagogical presentations” (Shillingsburg 2013, p. 2). I principi sono quattro, ma solo il terzo riguarda prettamente lo sviluppo software, invitando a utilizzare un approccio modulare nella progettazione e nella struttura dei componenti informatici, mentre gli altri tre riguardano il modo in cui i dati alla base di un’edizione digitale o in generale di un archivio devono essere strutturati.⁷⁴

Altri modelli sono stati ideati per una specifica tipologia di strumento di produzione. Alla sezione 1.3.3.2, per esempio, abbiamo citato il “modello Gothenburg”, elaborato dagli sviluppatori di CollateX e Juxta, per indicare quali sono le operazioni che deve effettuare un software per la collazione automatica. Nel campo della trascrizione automatica, invece, è stato sviluppato SegmOnto (Gabay et al. 2021), un vocabolario⁷⁵ controllato che descrive i contenuti delle pagine di un libro a stampa o manoscritto, allo scopo di omogeneizzare i dati richiesti in input dagli strumenti software per la *layout analysis* (cioè l’analisi di come i contenuti sono disposti in una pagina). L’obiettivo ultimo per il quale SegmOnto è stato sviluppato è affinché diventi uno standard di riferimento per progettare il flusso di lavoro per l’estrazione automatica di testo dalle scansioni di pagine.

I due problemi sopra enunciati (p. 58) possono essere arginati, dunque, grazie a uno sforzo collettivo di modellizzazione dei requisiti dei software condotto da chi conosce a fondo il problema.

Infine, il problema dell’inadeguatezza è strettamente legato a un altro problema diffuso tra gli strumenti per la pratica ecdotica, già osservato nella parte dedicata all’inaccessibilità, ovvero la frequente mancanza della documentazione tecnica. Se il modello dei dati, i requisiti, e le funzionalità

⁷⁴ Cfr. “**1st Principle** For literary works created in pre-digital eras, manuscripts and print primary materials are physical. The digital surrogates representing the primary materials are secondary-- - a first level of interpretation. The critical analysis of the materials is tertiary --another level of interpretation. [...] **2nd Principle** Digital imaging is the closest one can come electronically to the physical primary original. Digital transcriptions, by contrast, are new typesettings, regardless of their mode of creation. [...] **3rd Principle** Modular (component) design and structure are best for digital tools, tasks, and content types. [...] **4th Principle** Distinguishing textual properties from analytical and explanatory markup is essential to the durability of archival surrogates” (Shillingsburg 2013, p. 3).

⁷⁵ In informatica il termine ‘vocabolario’ (in inglese *vocabulary*) fa riferimento a uno degli elementi costitutivi del web semantico e indica un insieme di definizioni di concetti e di relazioni tra di essi, definito allo scopo di rappresentare un’area di interesse. I vocabolari possono essere usati per determinare i termini che possono essere usati in un applicativo (cfr. <https://www.w3.org/standards/semanticweb/ontology>). Nel caso di SegmOnto, il vocabolario definisce i tipi di dati che un *layout analyser* può individuare in una pagina.

dello strumento sono espliciti e descritti in modo facilmente comprensibile anche da parte di chi non possiede competenze informatiche, per gli utenti risulta più facile valutare se lo strumento è adatto o meno ai propri scopi scientifici. Inoltre, la presenza di una descrizione formale dell'applicazione, facilita la manutenzione e il riuso del codice sorgente da parte di terzi.

1.5 Le domande di ricerca

Il panorama degli strumenti per la filologia digitale, tracciato nel presente capitolo, risulta essere frammentato: si osserva da un lato un continuo proliferare di strumenti nati *ad hoc* per particolari progetti di edizione; dall'altro lato il rapido alternarsi degli strumenti generici realizzati negli ultimi trent'anni. Gli strumenti per la pratica ecdotica oggi disponibili faticano ad affermarsi all'interno della comunità scientifica, in quanto vengono ritenuti poco o per niente usabili per via della loro instabilità, inaccessibilità e inadeguatezza.

Analizzando il problema dal punto di vista di chi produce gli strumenti informatici, sorge spontaneo domandarsi che cosa debba essere cambiato nelle modalità di sviluppo, al fine di produrre degli strumenti generici efficaci e longevi. La risposta chiaramente non è univoca, in quanto i problemi rilevati sono molteplici e complessi. L'obiettivo del presente lavoro di tesi è quello di proporre una possibile soluzione a uno dei problemi enunciati, cioè l'inadeguatezza, focalizzandosi sugli strumenti che si occupano della visualizzazione di un'edizione scientifica digitale.

La progettazione di strumenti di visualizzazione è un compito complesso, per una serie di ragioni diverse:

- sono strumenti che si rivolgono a due utenti distinti, da un lato il filologo che desidera pubblicare la propria edizione, dall'altro il lettore, l'utente finale dello strumento informatico;
- è difficile progettare uno strumento che si adatti a più edizioni, in quanto edizioni diverse negli intenti e nei contenuti richiedono accorgimenti differenti in termini di modellizzazione dei dati, di funzionalità e di interfaccia grafica, per valorizzare i contenuti prodotti dal filologo e le peculiarità del testo o dei testi editi;

- il filologo presta massima attenzione alle modalità di presentazione dei contenuti, alle funzionalità offerte e alla resa complessiva del proprio prodotto editoriale, per cui lo strumento deve garantire al filologo pieno controllo su questi aspetti;
- le modalità e i flussi di lavoro in cui le edizioni scientifiche digitali vengono realizzate non sono omogenei, nonostante siano molto frequenti l'adozione di XML/TEI per l'allestimento del file sorgente e il ricorso alle specifiche di IIIF per le immagini.

Sulla base di queste considerazioni, si può dire che l'inadeguatezza riscontrata negli strumenti di visualizzazione può dipendere sia da come lo strumento è stato implementato, quindi le tecnologie utilizzate, la struttura e il formato dei dati previsti in input, ecc., sia da come è stato progettato su un piano più astratto e logico. Ci sono, dunque, due piani distinti: uno implementativo e l'altro più astratto.

Per spiegare la differenza tra questi due piani, si considerino due edizioni scientifiche digitali diverse. Ciascuna è basata su un file sorgente in XML/TEI, dunque entrambe potrebbero essere visualizzate mediante uno strumento che in input preveda tale formato. La prima edizione, però, è un'edizione sinottica in cui si raffrontano le trascrizioni di due diverse stesure della medesima opera letteraria, accompagnate da un fitto apparato di note di commento. La seconda edizione, invece, è un'edizione critica in cui il testo originale viene ricostruito sulla base di cinque testimoni. Il testo critico è quindi corredato da un apparato critico e dalla traduzione integrale in un'altra lingua. Lo strumento di visualizzazione generico sarà in grado di rappresentare entrambe le edizioni in modo molto elementare, quindi senza accorgimenti grafici o funzionalità particolari, a meno che non sia stato progettato per valorizzare i contenuti di entrambe le tipologie di edizioni. Nel primo caso, le due revisioni della prima edizione (così come il testo critico e la traduzione della seconda) potrebbero essere presentate una di seguito all'altra anziché affiancate. In uno strumento apposito, invece, i due testi figurerebbero nell'interfaccia affiancati e la loro lettura in parallelo potrebbe essere favorita da funzionalità *ad hoc*, quali lo scorrimento automatico di entrambi i testi insieme.

Quando si progetta uno strumento di visualizzazione, lo sviluppatore deve tenere in conto diversi aspetti, alcuni più tangibili e concreti altri più astratti, interrogandosi fondamentalmente su:

- 1) i **dati**: qual è il modello dei dati? Come sono formattati? Come devono essere recuperati? Come devono essere elaborati?
- 2) la **presentazione**: come deve essere strutturata l'interfaccia? Da quali e quanti componenti deve essere costituita? Come devono essere presentati i dati? In quali componenti?
- 3) le **funzionalità**: quali funzionalità devono essere implementate? Le funzionalità su quali dati operano?
- 4) l'**obiettivo**: a quali scopi scientifici si presta lo strumento? In che modo può essere applicato? Quali risultati consente di ottenere?

Nella letteratura scientifica, sono state proposte due diverse soluzioni al problema dell'inadeguatezza degli strumenti generici per la visualizzazione di edizioni scientifiche digitali. Una soluzione è più teorica e riflette il punto di vista sul problema del filologo che desidera realizzare l'edizione scientifica digitale. Tale soluzione, proposta da Pierazzo, consiste nella definizione da parte della comunità scientifica di modelli che rispecchino le metodologie e le basi teoriche del procedimento editoriale:

“In order to be able to offer a stable infrastructure for digital editions it is therefore necessary to reflect on the features that the scholarly community will consider essential to a particular type of text or scholarly problem and to **agree on some essential models** which take into account the new affordances offered by the digital.” (Pierazzo 2019, mia enfasi)

La seconda soluzione, invece, incarna il punto di vista dello sviluppatore informatico, che si domanda concretamente come progettare e sviluppare strumenti che risultino essere non solo adeguati, ma anche riusabili e facili da mantenere. Questa soluzione, elaborata da Del Grosso, Giovannetti e Marchi (2017), si prefigge di consentire lo sviluppo di componenti software che rappresentino le “entità” e le “procedure” alla base dell'*editing* filologico, per esempio il concetto di testo, di annotazione, di testimone, ecc., mediante l'applicazione dei principi della programmazione orientata agli oggetti:

“We are working on **the definition of some textual scholarship entities** (e.g. the document, the text, the edition, the witness, etc.) together with the functions needed to manipulate them as shared and implementation-independent objects. Figure 1 shows a diagram of **our Object-Oriented model for textual scholarship designed starting from the identification of the Domain Specific Abstract Data Types (DS-ADTs)**. An ADT is a high-level and mathematically-founded datatype the internal representation of which **is not directly accessible from users** (information hiding).” (Del Grosso et al. 2017, mia enfasi)

Una volta stabilite le rappresentazioni astratte delle “textual scholarship entities” da poter utilizzare nel codice del software, questa seconda soluzione prevede che l’implementazione degli strumenti informatici venga condotta seguendo cinque fasi principali, che bene o male corrispondono alle fasi in cui tradizionalmente si articola la realizzazione di un software:

“A) **the involvement of the scholar community to gather the requirements and define the specifications**; B) the design of the single components following the Domain-Driven and the User-Centered approach; C) the design of the general architecture using the Pattern-Oriented approach and the UML diagrams; D) the development of the software according to the S.O.L.I.D. principles; and E) the implementation of advanced Graphical User Interfaces through a specific UI framework.” (Del Grosso et al. 2017, mia enfasi).

Queste due diverse soluzioni, quella “filologica” dei modelli editoriali e quella “informatica” della definizione di entità della critica testuale nell’ottica della programmazione orientata agli oggetti, non si escludono a vicenda, al contrario, si completano, sopperendo l’una ai limiti dell’altra. La soluzione “informatica” è molto dettagliata e solida da un punto di vista dell’ingegneria del software. L’unico difetto è che si basa sulla definizione delle entità della critica testuale in una forma comprensibile e visibile solo agli sviluppatori informatici, in quanto i Domain Specific Abstract Data Types rimarrebbero nascosti nel codice sorgente dello strumento. La soluzione “filologica”, invece, porterebbe alla realizzazione di modelli facilmente comprensibili da parte degli studiosi, ma non è chiaro in che modo questi modelli potrebbero essere applicati dagli sviluppatori software per sviluppare uno strumento informatico.

Per risolvere il problema della inadeguatezza degli strumenti per la visualizzazione di edizioni scientifiche digitali, occorre adottare entrambe le prospettive, quella dello studioso e quella dello sviluppatore informatico. La prospettiva dell’ingegneria del software è quella che più di frequente viene trascurata nella realizzazione di strumenti informatici per la filologia digitale.

“The development of applications in the field of Digital Humanities (DH) does not adequately take into account domain modelling, software design principles and software engineering methodologies. In fact, many systems developed in the context of DH-related projects have not been conceived to be modular, extensible, and scalable: they only tend to solve specific problems such as data-driven and project-oriented tools. In addition, **most projects focus on the requirements of humanists (as end users), but leave out the needs of software developers**” (Del Grosso et al. 2016).

Molti aspetti della soluzione di Del Grosso, Giovannetti e Marchi sono stati ripresi per lo sviluppo della proposta di soluzione descritta nel prossimo capitolo, in particolare l’approccio *object-*

oriented e la centralità dell'analisi del dominio per la progettazione degli strumenti informatici. Tuttavia, si è ritenuto più efficace imbastire la soluzione partendo dalla prospettiva dell'utente, quindi dal bisogno da parte della comunità scientifica di "editorial models" (Pierazzo 2019).

Il presente lavoro di ricerca si propone di indagare a fondo, non solo dal punto di vista della critica testuale, ma anche da quello dell'ingegneria del software, che tipo di modelli occorranza e come debbano essere realizzati e condivisi. Gli obiettivi che hanno guidato la presente ricerca possono essere riassunti, dunque, in tre principali domande:

1. su quali basi teoriche devono essere fondati dei modelli per lo sviluppo di strumenti informatici per la visualizzazione di edizioni scientifiche digitali?
2. quale strategia di modellizzazione bisogna adottare per elaborare i modelli?
3. come devono essere formalizzati i modelli per poter risultare utili sia agli sviluppatori, sia come riferimento per la comunità scientifica?

Capitolo 2

La strategia di modellizzazione

Come illustrato nel precedente capitolo, la maggior parte delle edizioni scientifiche digitali viene realizzata con strumenti informatici sviluppati su misura, mentre i pochi strumenti generici disponibili faticano a imporsi, in quanto sono ritenuti poco usabili.

In questo capitolo si propone una possibile soluzione a uno dei problemi individuati analizzando lo stato dell'arte nel campo del software per la filologia digitale: l'inadeguatezza degli strumenti di visualizzazione rispetto ai requisiti ritenuti fondamentali per le edizioni scientifiche digitali da parte degli studiosi. La soluzione consiste nella definizione di una strategia di modellizzazione, da adottare collettivamente a livello di comunità scientifica, al fine di elaborare dei modelli condivisi, che siano in grado di rappresentare le esigenze scientifiche dei filologi e, al contempo, risultino essere uno strumento utile per chi si occupa dello sviluppo di software. In altre parole, i modelli devono rivelarsi capaci di colmare la distanza tra l'edizione e lo strumento informatico, per migliorare l'efficacia del secondo. Il fine ultimo dei modelli è quello di facilitare lo sviluppo di software, anche in un'ottica di riuso e di manutenzione nel tempo, e di consentire una maggiore adattabilità degli strumenti informatici a più edizioni.

Sono tre le principali domande che devono guidare la messa a punto della strategia di modellizzazione:

1. su quali basi teoriche devono essere fondati dei modelli per lo sviluppo di strumenti informatici per la visualizzazione di edizioni scientifiche digitali?
2. quale strategia di modellizzazione bisogna adottare per elaborare i modelli?
3. come devono essere formalizzati i modelli per poter risultare utili sia agli sviluppatori sia come riferimento per la comunità scientifica?

Il capitolo è sviluppato in modo tale da rispondere nell'ordine alle domande sopra elencate.

2.1 La base teorica per stabilire i requisiti di uno strumento di visualizzazione generico

Il problema dell'inadeguatezza si riscontra quando uno strumento informatico non è in grado di soddisfare i requisiti ritenuti indispensabili dalla propria comunità di utenti. Questi requisiti variano a seconda del campo e dello scopo per cui lo strumento informatico viene utilizzato e riguardano il modo in cui lo strumento dovrebbe essere sviluppato sia da un punto di vista tecnico-implementativo sia da un punto di vista logico-organizzativo.

Nel campo degli strumenti informatici per la visualizzazione di edizioni scientifiche digitali alcune pratiche si sono consolidate, come il modello sorgente-*output* e l'adozione di XML/TEI come formato di riferimento. Tuttavia, la comunità scientifica non ha formalizzato in modo esplicito delle indicazioni su come questo tipo di software debbano essere sviluppati.

La soluzione al problema della mancanza di riferimenti chiari, proposta da Pierazzo (2019), è quella di riflettere su quali sono i requisiti, ritenuti indispensabili dalla comunità scientifica, per un particolare tipo di testo o di problema della critica testuale, e di stabilire dei modelli essenziali che tengano in conto le potenzialità offerte dal *medium* digitale.

Esattamente a che tipo di “modelli” fa riferimento Pierazzo? A dei modelli editoriali (“editorial models”), come per esempio l'edizione critica, l'edizione diplomatica, ecc.

Per valutare se i modelli editoriali permettano o meno di inquadrare i requisiti per uno strumento di visualizzazione, si propone un ulteriore estratto del medesimo articolo come spunto di riflessione iniziale:

“print culture has produced a limited number of **very identifiable and long-lasting models**, such as, for instance, diplomatic editions and critical editions (regardless of the underlying theoretical approach on which they are based). Digital textual scholarship has in turn created other models, such as, for instance, the hypertextual edition, the paradigmatic edition, and the social edition, but none of these editions inform its user of what to expect from a textual point of view, since **their typology focuses mostly on the way in which they have been produced or the way in which they are offered to their users, not on the editorial approach.**” (Pierazzo 2019).

Il cambio di *medium*, dalla stampa al digitale, non ha completamente rivoluzionato le tipologie di edizioni che vengono pubblicate nel web:

“present work and future developments in digital scholarship evolve from critical models that we have inherited. The basic procedures and goals of scholarly editing will not change because of digital technology” (Buzzetti e McGann 2006).

La maggior parte delle edizioni scientifiche digitali può essere inquadrata in uno dei due modelli menzionati da Pierazzo: da un lato l’edizione critica, dall’altro l’edizione diplomatica/interpretativa. Questi due modelli corrispondono ai due pilastri della critica testuale “codex-based”: “facsimile editing of individual documents and critical editing of a set of related documentary witnesses” (Buzzetti e McGann 2006). Haugen propone le etichette di “reconstructive and non-reconstructive editing” (Roelli (a cura di) 2020, p. 366), ma il concetto è lo stesso.

Questi due modelli editoriali, che potremmo chiamare “tradizionali”, sono un punto di riferimento anche nel campo della filologia digitale. Vengono utilizzati, infatti, come etichette sia dai curatori di ESD per definire il proprio lavoro, sia come riferimento per valutare la qualità delle edizioni pubblicate *online*. Si pensi all’iniziativa dell’Osservatorio Permanente sulle Edizioni Digitali di autori Italiani (OPEDI), ideata da Michelangelo Zaccarello, che presenta tra i propri principali obiettivi proprio quello “di assicurare che la metodologia dell’ecdotica tradizionale sia tenuta in considerazione anche nel caso delle edizioni digitali o digitalizzate presenti in rete” (Zaccarello 2018).

I modelli editoriali tradizionali sono, dunque, dei riferimenti imprescindibili anche in ambito digitale. Tuttavia, nel contesto della presente ricerca il loro utilizzo per determinare come uno strumento di visualizzazione debba essere realizzato rischia di essere limitante, per due principali ragioni.

La prima ragione è individuabile nella citazione di Pierazzo: questi modelli editoriali forniscono delle indicazioni chiare sull’approccio adottato dall’editore e sulle caratteristiche del testo edito. Per stabilire i requisiti di uno strumento informatico occorrono però delle indicazioni anche su come le ESD vengono prodotte da un punto di vista tecnologico (gli strumenti utilizzati, le modalità di archiviazione, ecc.) e come devono essere offerte ai propri utenti. Uno strumento per la visualizzazione di ESD, se ancorato esclusivamente a dei modelli editoriali nati e perfezionati nel contesto delle edizioni a stampa, rischia di risultare limitato e inadatto. Bisogna tenere in considerazione, quindi, anche i modelli di produzione e fruizione digitali, quali la *social edition* e l’edizione ipertestuale menzionate da Pierazzo. In questo modo verrebbe sfruttato il potenziale offerto dal digitale, mentre i modelli

editoriali tradizionali possono essere utilizzati per determinare le finalità e l'organizzazione logica di uno strumento di visualizzazione da un punto di vista più astratto.

La seconda ragione è che i due modelli tradizionali sono basati su un oggetto di studio preciso: il “codex”, o più in generale un documento manoscritto o a stampa. Finché l'oggetto di studio della filologia digitale continua a essere il “codex”, è piuttosto ovvio che i modelli editoriali “codex-based”, consolidatisi per la realizzazione di edizioni a stampa, vengano utilizzati anche per produrre edizioni scientifiche digitali, in quanto i problemi testuali da risolvere e le finalità dell'editore sono le stesse a prescindere dal mezzo di pubblicazione. Nel momento in cui l'oggetto di studio cambia, emerge la necessità di nuovi modelli editoriali per poterlo rappresentare al meglio. Si consideri come esempio lo studio di testi *born-digital* e le problematiche riguardo all'individuazione delle diverse fasi di redazione e revisione del testo di un documento elettronico salvato su disco o in *cloud*.¹

Più in generale si può dire che ogni modello editoriale rifletta le pratiche che si sono consolidate in un certo ambito della filologia, per studiare un certo tipo di testo e per raggiungere degli obiettivi precisi. Nei diversi rami della filologia (come la filologia germanica, la filologia romanza, la filologia d'autore, ecc.) che tradizionalmente studiano il medesimo oggetto, cioè il libro manoscritto o a stampa, si possono comunque riscontrare delle differenze a livello metodologico, che inevitabilmente si riflettono anche nei modelli editoriali. Si pensi ai concetti di “apparato genetico” e di “apparato evolutivo” (Raboni e Italia 2010), che influiscono sul modo in cui un'edizione scientifica deve essere realizzata e fruita. Queste differenze possono scaturire non solo dall'ambito filologico, ma anche dall'approccio metodologico di fondo. In altre parole: “The edition [...] is not a single, monolithic entity but a product that can be representative of a number of academic traditions and scholarly practices.” (Roelli (a cura di) 2020, p. 364).

Per inquadrare i requisiti di uno strumento per la visualizzazione, occorre fare riferimento agli approcci e alle metodologie della filologia, non solo ai modelli editoriali tradizionali e/o ai modelli emersi nella pratica della filologia digitale. Altrimenti il rischio è che tali modelli vengano riproposti senza una riflessione più profonda da parte della comunità scientifica su come un'edizione scientifica

¹ Per approfondire il tema della filologia di testi *born-digital* e lo studio di archivi digitali si rimanda al volume *Editing Duemila. Per una filologia dei testi digitali* (Italia 2020), si veda inoltre Carbè 2023 e Ries 2023.

digitale debba essere realizzata e fruita. Le diverse pratiche filologiche e le rispettive esigenze scientifiche verrebbero nuovamente appiattite su poche soluzioni standard e il risultato porterebbe ad avere degli strumenti informatici ancora considerati dagli utenti poco o per niente usabili.

Un altro aspetto che porta a preferire l'approccio filologico in termini generali, invece del rispettivo modello editoriale, come base teorica per lo sviluppo di strumenti di visualizzazione, è che in questo modo gli strumenti sarebbero classificabili non solo rispetto al tipo di edizione che consentono di realizzare, ma anche in base al loro quadro teorico di riferimento. Di conseguenza, per i potenziali utenti, cioè gli editori che vogliono realizzare un'edizione scientifica digitale, sarebbe più facile valutare l'idoneità di uno strumento informatico rispetto alle proprie esigenze.

Il punto di partenza per la progettazione di strumenti di visualizzazione generici efficaci deve essere, dunque, la teoria della critica testuale, descritta tenendo conto delle varie metodologie e pratiche che si sono sviluppate in ambiti filologici diversi e per esigenze diverse.

2.2 Una strategia di modellizzazione invece di un unico modello

Da quanto fin qui osservato emerge in che misura la modellizzazione che si intende realizzare sia complessa. Rappresentare diverse metodologie filologiche in un unico modello si rivelerebbe essere poco efficace, in quanto si tratterebbe di un modello molto ampio e per forza di cose molto generico. È vero che a livello concettuale, quindi su un piano più astratto, diversi approcci filologici presentano svariati aspetti in comune. Tuttavia, per descrivere in modo approfondito uno strumento informatico sia su un piano astratto sia su un piano più tecnico e implementativo, conviene specificare modelli diversi in base alle esigenze dei singoli approcci filologici.

L'obiettivo della presente ricerca, dunque, non è la produzione di un unico modello ma l'elaborazione di una strategia di modellizzazione comune, che possa portare alla definizione di una rete di modelli "theoretically and editorially based" (Pierazzo 2019), utili sia per gli studiosi sia per gli sviluppatori che effettivamente implementano gli strumenti di visualizzazione generici.

Tale obiettivo è raggiungibile solo in una prospettiva a lungo termine e se il lavoro di modellizzazione viene svolto collaborativamente. Diversi gruppi di studiosi, anche non nel contesto di un particolare progetto di edizione digitale, potrebbero contribuire alla definizione di una determinata

metodologia filologica sulla base delle proprie sperimentazioni e dell'esperienza nel proprio campo di studi.

Inoltre, una rete di modelli potrebbe essere estesa *ad libitum*, descrivendo in modo più dettagliato le esigenze dei diversi approcci filologici e includendo eventuali metodologie che potrebbero essere sviluppate in futuro. In altre parole, questa potenziale rete di modelli deve costituire un punto di partenza e non di arrivo per la comunità scientifica, incoraggiando un consolidamento e una valorizzazione di quanto viene progressivamente sperimentato con le nuove tecnologie.

2.3 Una rete di modelli concettuali

Per elaborare una strategia di modellizzazione comune bisogna innanzitutto stabilire, da un punto di vista informatico, il tipo di modelli che meglio si prestano all'obiettivo scientifico di fondo: la definizione dei requisiti per gli strumenti di visualizzazione generici. Si procede, dunque, da un elenco delle caratteristiche generali che i modelli devono avere:

- A. devono descrivere l'organizzazione logica, i contenuti e le funzionalità previste per un'edizione scientifica digitale in modo abbastanza generico da adattarsi al maggior numero possibile di edizioni realizzate secondo il medesimo approccio filologico. Al contempo, devono essere sufficientemente dettagliati, affinché riescano a illustrare in modo chiaro e preciso il modo in cui l'applicativo deve essere sviluppato;
- B. devono essere espliciti. Molto spesso il modello logico di un applicativo è scritto direttamente all'interno del codice sorgente e lì rimane nascosto, risultando inaccessibile da parte di potenziali utenti o di altri sviluppatori che desiderano riutilizzarlo. Dei modelli espliciti, invece, sono verificabili e condivisibili;
- C. non devono dipendere da tecnologie né da piattaforme specifiche, specialmente da quelle proprietarie, in modo tale da essere più durevoli nel tempo e più versatili. In questo modo lo stesso modello potrebbe essere implementato più volte da sviluppatori diversi con le tecnologie da loro ritenute più adatte alle proprie necessità;
- D. devono risultare comprensibili sia agli sviluppatori sia ai potenziali utenti, cioè i filologi, così che quest'ultimi possano utilizzarli per verificare la compatibilità degli strumenti informatici con le proprie esigenze;

E. infine, devono essere espandibili e collegabili tra di loro, così da poter realizzare una rete di modelli e da poter essere adattati *ad hoc* per singoli progetti di edizione.

Come in molti campi del sapere, anche nel campo dell'ingegneria del software “modello” è un *umbrella-term*, che può essere usato per indicare diverse rappresentazioni di un sistema informatico, che assolvono a compiti differenti. Per fare chiarezza sui diversi tipi di modelli utilizzati in ambito informatico e scegliere quelli più adatti alla presente ricerca, si prendono a riferimento le definizioni fornite nella guida della *Model Driven Architecture* (MDA),² un approccio per la progettazione, lo sviluppo e l'implementazione di software promosso dall'Object Management Group (OMG).³ Le linee guida della MDA indicano come strutturare le specifiche software sotto forma di modelli, utilizzando UML e altri standard promossi dal consorzio OMG. L'approccio MDA si basa su alcuni concetti chiave: *system*, *model*, *modeling language*, *architecture* e *architectural layers* (livelli architetturali).⁴

Il termine “sistema” è definito come segue:

“A system is a collection of parts and relationships among these parts that may be organized to accomplish some purpose. In MDA, the term ‘system’ can refer to an information processing system but it is also applied more generally. Thus a system may include anything: a system of hardware, software, and people, an enterprise, a federation of enterprises, a business process, some combination of parts of different systems, a federation of systems” (MDA Guide, p. 5)

Il concetto di sistema definito dalla MDA si presta a rappresentare anche un qualsiasi tipo di strumento di visualizzazione generico.

Il modello è definito come “information selectively representing some aspect of a system based on a specific set of concerns” (*ibid.*, p. 5). Il modello è, dunque, una rappresentazione di un qualche aspetto di un sistema, creato per rispondere a delle domande specifiche. Il modello è collegato al sistema attraverso una mappatura esplicita o implicita. I modelli si differenziano tra di loro in base al punto di vista e al livello di astrazione adottati per rappresentare il sistema, come illustrato oltre. Alcuni tipi di modello sono standard e comuni alla maggior parte dei sistemi, mentre altri possono riguardare un aspetto specifico del dominio in cui è applicato il sistema.

² Sito web dedicato alla specifica dell'approccio: <https://www.omg.org/mda/>.

³ Per maggiori informazioni sul consorzio OMG si rimanda alla nota 26 del capitolo 1.

⁴ Cfr. MDA Guide, pp. 5 - 8.

Il modello è utilizzato in primo luogo come strumento di comunicazione all'interno di un gruppo di sviluppatori (per esempio all'interno di un'azienda) o di una comunità più ampia di sviluppatori che lavorano nel medesimo dominio (campo applicativo), per esempio gli sviluppatori di software gestionali per le banche. Perché il modello sia uno strumento di comunicazione efficace, deve essere espresso mediante un linguaggio di modellizzazione comune e condiviso. Grazie all'utilizzo di linguaggi di modellizzazione formali standard, il modello è interpretabile anche dalle tecnologie utilizzate per implementare il sistema. Il linguaggio di modellizzazione è definito come: "The structure, terms, notations, syntax, semantics, and integrity rules that are used to express a model constitute the modeling language" (*ibid.*, p. 6). Tra i linguaggi di modellizzazione più noti e utilizzati, citati nella Guida di MDA, ci sono UML (vd. 4.2), SQL Schema, BPMN, E/R, OWL (vd. 2.5.2), e XML Schema.⁵

Con *architecture* nella guida di MDA si intende sia un insieme di modelli che rappresentano un sistema sia l'attività o la pratica di realizzare tali modelli. Lo scopo ultimo di un'architettura è quello di definire o di migliorare un sistema, mettendo a fuoco l'ambito in cui il sistema deve essere applicato e quali sono le esigenze di chi ha commissionato il software o dei potenziali utenti,⁶ arrivando a una progettazione che soddisfi tali esigenze (Cfr. *ibid.*, p. 6)

Infine, all'interno di un'architettura si possono individuare diversi livelli, in base al grado di astrazione. I livelli architetturali possono essere molteplici, ma, principalmente, è possibile distinguerne tre:

1. **"Business or domain models"**, che rappresentano le persone, i luoghi, gli oggetti e le regole del dominio. Le istanze,⁷ ovvero gli oggetti delle diverse classi di questi modelli corrispondono a degli enti reali, non alle rappresentazioni di tali enti all'interno di un sistema informatico;

⁵ SQL Schema è specialistico per la definizione di database. BPMN (<https://www.bpmn.org/>) sta per Business Process Model and Notation ed è un linguaggio di modellizzazione standard sviluppato dall'OMG. Con 'schema XML' si intende, invece, un linguaggio per rappresentare i vincoli posti ai documenti XML (<https://www.w3.org/standards/xml/schema>). Il modello costituito dalle linee guida TEI, per esempio, è espresso anche sotto forma di schema XML. Ci sono diversi linguaggi per esprimere uno schema XML, tra cui i principali sono: Document Type Definitions (DTDs), Relax-NG, Schematron e W3C XSD (XML Schema Definitions).

⁶ In inglese si usa in generale il termine *stakeholder*, per indicare una persona o un gruppo che ha investito nel sistema o è interessato ad esso.

⁷ Per la definizione di "istanza" si rimanda alla sezione 2.5.4.

2. **“Logical system models”**, che rappresentano il modo in cui i componenti di un sistema interagiscono tra di loro, con le persone e con le organizzazioni, nell’assistere un’organizzazione o una comunità di utenti nel raggiungere i propri obiettivi;
3. **“Implementation models”**, che rappresentano come un determinato sistema è implementato in modo tale da svolgere le sue funzioni. Questi modelli sono legati a una particolare tecnologia o piattaforma di implementazione.

I requisiti B - E non dipendono tanto dalla tipologia di modelli in sé, quanto piuttosto dalle modalità in cui i modelli vengono formalizzati e dalla scelta dei linguaggi di modellizzazione. Il requisito A, invece, dipende dalla tipologia dei modelli. Tenendo a riferimento le tre tipologie sopra elencate, occorre individuare qual è la più adatta per soddisfare il requisito A.

Come illustrato nel precedente capitolo, negli strumenti di visualizzazione l’inadeguatezza rispetto alle ESD si manifesta più spesso su un piano astratto, piuttosto che implementativo, e riguarda le finalità scientifiche che lo strumento informatico permette di perseguire. Per questo motivo la strategia di modellizzazione dovrebbe puntare a realizzare per ogni approccio filologico *in primis* dei *business* o *domain model*. Questo tipo di modelli costituirebbero dei “modelli editoriali” o più in generale dei modelli della pratica ecdotica, in quanto definirebbero le finalità e gli approcci filologici per i quali gli strumenti informatici potrebbero essere applicati. L’altro aspetto interessante di un *domain model* è che da esso possono essere derivati gli altri due tipi di modello (2 e 3), applicando i principi del *domain-driven design*, illustrati nella prossima sezione. In altre parole, se a monte vengono inquadrati i contenuti testuali e gli obiettivi scientifici, in un secondo momento è possibile stabilire l’organizzazione logica, le funzionalità e le modalità di implementazione dello strumento di visualizzazione più consone. Infine, un *domain model* che descrive in modo efficace il dominio permette di individuare i requisiti basilari di un sistema applicato nel dominio, risultando più facilmente utilizzabile per la produzione di software diversi, come si evince dalla definizione di *domain model* fornita nel *Systems and Software Engineering Vocabulary*:

“The domain model identifies and describes the structure of data, flow of information, functions, constraints, and controls within the domain that are included in software systems in the domain. **The domain model describes the commonalities and variabilities among requirements for software systems in the domain**” (mia enfasi).

La strategia di modellizzazione dovrebbe puntare a realizzare prima di tutto una rete di *domain model* o “modelli concettuali”. L’etichetta “modello concettuale” in ambito informatico è sostanzialmente equivalente a quella di *domain model*.⁸ Nel contesto del presente lavoro di ricerca sembra più opportuno utilizzare l’etichetta “modello concettuale” invece di “modello del dominio”, in quanto permette anche a persone non esperte di informatica di intuire che si tratta di un modello che descrive uno strumento informatico su un piano astratto.

Per soddisfare i requisiti B e C, i modelli concettuali possono essere realizzati rispettando le linee guida dell’approccio MDA. L’assunto di base dell’approccio MDA, infatti, è che la piattaforma tecnologica, mediante la quale viene sviluppato un software (per esempio .NET, CORBA, e simili), non debba influire sul modo in cui il software viene sviluppato in termini di organizzazione logica e funzionalità. I modelli realizzati secondo l’approccio MDA sono cioè *platform-independent*: da un lato vengono espressi mediante linguaggi di modellizzazione standard, dall’altro documentano le funzionalità e il comportamento di un’applicazione, rimanendo separati e distinti dal codice che li implementa. In questo modo risultano essere dei modelli espliciti e riutilizzabili. L’ulteriore vantaggio che questa separazione comporta è che “[n]o longer tied to each other, the business and technical aspects of an application or integrated system can each evolve at its own pace - business logic responding to business need, and technology taking advantage of new developments - as the business requires”.⁹ Considerando la rapidità con la quale si alternano le tecnologie anche nel campo della filologia digitale, pare decisamente vantaggioso produrre dei modelli concettuali che durino nel tempo, dai quali derivare di volta in volta dei modelli implementativi sulla base delle caratteristiche delle tecnologie del momento.

Infine, i requisiti D ed E possono essere soddisfatti scegliendo un opportuno linguaggio standard di modellizzazione. La sezione 2.5 illustra nel dettaglio il linguaggio scelto e le relative motivazioni.

⁸ A sostegno della presente affermazione si riporta la definizione di “conceptual model” fornita nel *Systems and Software Engineering Vocabulary*: “model of the concepts relevant to some endeavor”.

⁹ Cfr. <https://www.omg.org/mda/>.

2.4 I principi del *Domain-Driven Design*

Il punto di partenza per la realizzazione di un modello concettuale è l'analisi del dominio. Nel campo dell'ingegneria del software è stato elaborato un approccio allo sviluppo di software complessi denominato *domain-driven design* (DDD), letteralmente traducibile in 'progettazione guidata dal dominio'. L'idea di fondo di questo approccio o "software philosophy", come la definisce Eric Evans, suo ideatore e autore del libro *Domain-Driven Design. Tackling Complexity in the Heart of Software* (2004), è che analizzare accuratamente il dominio applicativo facilita la realizzazione e la manutenzione del software.

I principi alla base del DDD si possono riassumere in tre punti (cfr. Evans 2015):

1. **concentrarsi sul dominio principale**, definito come "a sphere of knowledge, influence or activity. The subject area to which the user applies a program is the domain of the software" (*ibid.*). La progettazione di un software deve essere fondata su un'accurata analisi del dominio, dei problemi che lo contraddistinguono e delle entità che ne fanno parte insieme alle relazioni tra di loro;
2. **esplorare i modelli in una collaborazione creativa tra gli esperti del dominio e gli sviluppatori**. Solitamente gli sviluppatori si confrontano con i committenti del software (che rappresentano gli esperti del dominio) nella prima fase del lavoro di progettazione, per determinare i requisiti che il software deve avere. Nel DDD, invece, sviluppatori ed esperti del dominio analizzano insieme il dominio e creano un modello, "a system of abstractions that describes selected aspects of a domain and can be used to solve problems related to that domain" (*ibid.*). Si tratta in altre parole di un modello che rappresenta il dominio e al contempo determina la logica e il funzionamento del software: quali sono i dati gestiti nel software, quali sono i diversi componenti logici, come interagiscono i componenti tra di loro, ecc. Dunque la collaborazione tra sviluppatori ed esperti del dominio determina concretamente come viene scritto il codice, non solo il risultato finale atteso.

3. **parlare in un “ubiquitous language” (linguaggio onnipresente)**¹⁰ La collaborazione tra sviluppatori ed esperti del dominio porta prima di tutto alla definizione di un vocabolario comune e alla creazione di un vero e proprio linguaggio, che riflette il dominio, i suoi problemi e il suo funzionamento. Questo linguaggio è il ponte tra la fase di progettazione e la fase di sviluppo: nel momento in cui i termini stabiliti durante la progettazione vengono utilizzati anche nella scrittura del codice vero e proprio, il codice risulta comprensibile a tutti i membri del gruppo ed è più facile valutare la coerenza tra il codice e il dominio. Questo linguaggio è detto *ubiquitous*, in quanto è presente in tutte le fasi di lavoro e utilizzato da tutti i membri del gruppo di sviluppo. Il linguaggio, però, per essere pienamente comprensibile deve essere utilizzato in un contesto chiaro e riconoscibile (“explicitly bounded context” *ibid.*).

Riassumendo, si può dire che l’approccio DDD prevede che, analizzando il dominio, esperti del dominio e sviluppatori creino insieme un modello e definiscano un linguaggio comune da usare in tutte le fasi di sviluppo del software, dalla sua progettazione alla scrittura del codice. Il codice del software diventa un’espressione del modello.

L’adozione di questo approccio, secondo il suo autore, comporta molteplici vantaggi. *In primis* semplifica agli sviluppatori il lavoro di manutenzione del codice. Per gli esperti del dominio essere coinvolti nella progettazione della logica del software aiuta a riflettere in modo più concreto sui *desiderata* tecnici e su come il software debba funzionare. Inoltre l’utilizzo dell’*ubiquitous language* in fase di progettazione permette agli esperti di dominio di individuare velocemente le parti del modello inadatte ai propri scopi, contraddittorie oppure troppo vaghe (cfr. Evans 2004, p. 33). Infine, un modello solido che rappresenta il dominio efficacemente può rendere il software più usabile anche per l’utente finale:

¹⁰ Si riportano le definizioni fornite in Evans 2015 di “ubiquitous language” e di “bounded context”. Con la prima espressione si intende “a language structured around the domain model and used by all team members within a bounded context to connect all the activities of the team with the software”, mentre con la seconda “a description of a boundary (typically a subsystem, or the work of a particular team) within which a particular model is defined and applicable”.

“When a design is based on a model that reflects the basic concerns of the users and domain experts, the bones of the design can be revealed to the user to a greater extent than with other design approaches. Revealing the model gives the user more access to the potential of the software and yields consistent, predictable behavior” (Evans 2004, p. 59).

Volendo applicare questi principi all’argomento del presente lavoro di ricerca, il problema dell’inadeguatezza degli strumenti di visualizzazione potrebbe essere risolto non solo definendo una rete di modelli concettuali, ma anche assicurandosi che tali modelli concettuali siano sviluppati e applicati nello sviluppo di software secondo i principi del DDD. In primo luogo, i modelli dovrebbero essere elaborati dalla comunità scientifica in dialogo con gli sviluppatori degli strumenti informatici, in un’ottica fortemente interdisciplinare. In questo modo gli studiosi avrebbero la garanzia di elaborare dei modelli che risultino chiari e utili anche alle loro controparti informatiche. In secondo luogo, gli strumenti informatici dovrebbero essere sviluppati riflettendo in modo coerente e su tutti i diversi livelli il linguaggio che gli studiosi utilizzano per descrivere il proprio dominio. Grazie a questo accorgimento gli strumenti informatici risulterebbero immediatamente più familiari e usabili sia dagli utenti-editori sia dagli utenti finali. Infine, i modelli concettuali devono essere il punto di partenza per la definizione dei requisiti logici e funzionali di uno strumento di visualizzazione.

L’applicazione del DDD è utile anche per risolvere (o perlomeno sopperire) altri problemi individuati nel precedente capitolo, in quanto prevede che sviluppatori ed esperti del dominio nel mentre che elaborano insieme un modello, producano concretamente della documentazione tecnica. Questa documentazione serve prima di tutto alla manutenzione nel tempo del software, rende lo strumento informatico più facilmente adattabile e riutilizzabile, e aiuta i filologi interessati a utilizzare lo strumento a valutarne l’adeguatezza rispetto ai propri obiettivi scientifici.

2.5 La formalizzazione dei modelli concettuali

Nelle precedenti sezioni sono state delineate nell’ordine: la base teorica per descrivere i requisiti di uno strumento per la visualizzazione di edizioni scientifiche digitali; la necessità di elaborare una strategia di modellizzazione comune per produrre più modelli; la tipologia dei modelli da realizzare da un punto di vista informatico; l’approccio allo sviluppo di software da applicare affinché uno strumento sia coerente con un modello. Per chiudere la definizione della strategia di modellizzazione e

rispondere a tutte le domande che hanno fin qui guidato la presente ricerca, l'ultimo aspetto da determinare è il linguaggio di modellizzazione da adottare per formalizzare i modelli concettuali.

Come illustrato nella sezione 2.3, i modelli concettuali devono essere formalizzati in modo tale da risultare comprensibili sia agli sviluppatori sia ai potenziali utenti, cioè i filologi, ed essere espandibili e collegabili tra di loro. Inoltre, il linguaggio di modellizzazione deve essere standard e non proprietario. Sulla base dei requisiti elencati, la soluzione ritenuta più adatta è sviluppare i modelli concettuali sotto forma di ontologie espresse nel linguaggio OWL, uno dei linguaggi standard del W3C per definire e condividere ontologie attraverso il web (vd. 2.5.2).

Nella pubblicazione scientifica che ha introdotto ufficialmente il web semantico (Berners-Lee et al. 2001), il concetto di ontologia fu presentato nel seguente modo:

“In philosophy, an ontology is a theory about the nature of existence, of what types of things exist; ontology as a discipline studies such theories. Artificial-intelligence and Web researchers have co-opted the term for their own jargon, and for them an ontology is a document or file that formally defines the relations among terms.”

Le ontologie¹¹ o, più in generale, i vocabolari, sono uno degli elementi costitutivi del web semantico (il *web of data*). Sono state create allo scopo di stabilire e definire la natura dei collegamenti logici tra i dati contenuti in diverse risorse web:

“Ontologies are attempts to more carefully define parts of the data world and to allow interactions between data held in different formats.” (Shadbolt et al. 2006, p. 100)

A livello pratico, le ontologie vengono utilizzate prevalentemente per: facilitare l'integrazione tra dati provenienti da basi di dati differenti (es.: il dato “codice postale” del database A corrisponde al dato “zip code” del database B); scoprire nuove relazioni possibili tra le diverse tipologie di dati contenuti all'interno della stessa base di dati; stabilire dei termini e delle possibili relazioni per organizzare la conoscenza, ad esempio nell'ambito di una biblioteca o di un archivio.

Dato che un'ontologia consente di definire dei termini e di stabilire delle relazioni, si presta anche come strumento per descrivere e modellizzare un aspetto della realtà:

“In the context of computer and information sciences, an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations among class

¹¹ Per una definizione completa di ontologia si rimanda a Guarino et al. 2009.

members). The definitions of the representational primitives include information about their meaning and constraints on their logically consistent application.” (Ciotti e Tomasi 2016, p. 4)

2.5.1 Perché un’ontologia?

I vantaggi nell’utilizzo di un’ontologia come modalità di formalizzazione del modello concettuale sono molteplici. In primo luogo, un’ontologia si basa su pochi concetti chiave, come quello di “classe”, “attributo” e “relazione”, immediatamente comprensibili anche da parte di una persona non esperta di ontologie, a differenza di altri linguaggi di modellizzazione, caratterizzati da una notazione particolare (si pensi ai diagrammi UML). In secondo luogo, in un’ontologia è possibile fornire per ciascuna classe e proprietà delle etichette e delle definizioni espresse in linguaggio naturale, così da ancorare saldamente il modello alla propria base teorica. Le etichette possono corrispondere ai termini tecnici, utilizzati in quel determinato ambito filologico, in lingue diverse, mentre le definizioni possono includere anche in modo esplicito dei riferimenti bibliografici.

Si riporta come esempio un estratto dell’ontologia Scholarly Editing (SE), descritta nel dettaglio nella sezione 2.5.5.6, in cui si vede la definizione della classe “Reading”. All’interno dell’elemento `<rdfs:Class>`, che corrisponde alla classe, sono presenti diversi elementi `<rdfs:label>`, ciascuno contenente un’etichetta della classe in una lingua diversa (italiano, inglese, tedesco, francese e latino), e la definizione nell’elemento `<rdfs:comment>`.

```
<rdfs:Class rdf:about="http://e-editiones.ch/ontology/scholarly-editing#Reading">
  <rdfs:label xml:lang="it">lezione</rdfs:label>
  <rdfs:label xml:lang="en">reading</rdfs:label>
  <rdfs:label xml:lang="de">Lesung</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://e-editiones.ch/ontology/scholarly-editing#" />
  <rdfs:comment xml:lang="en">Copy of one or more words or phrases of an original or witness.</rdfs:comment>
  <rdfs:subClassOf rdf:resource="http://e-editiones.ch/ontology/scholarly-editing#EditorialStructure" />
  <rdfs:label xml:lang="fr">lecture</rdfs:label>
  <rdfs:label xml:lang="la">lectio</rdfs:label>
  <rdfs:label xml:lang="it">lettura</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://e-editiones.ch/ontology/text#CopiedText" />
</rdfs:Class>
```

In terzo luogo, le ontologie si prestano a sviluppare collaborativamente una rete di modelli concettuali. Ontologie diverse, infatti, possono essere unite insieme oppure è possibile stabilire tra di esse delle mappature, favorendo così il riuso e la valorizzazione delle ontologie esistenti, invece di dover

produrre i modelli concettuali ogni volta da zero. Inoltre, alla base della modellizzazione mediante un'ontologia è posto un assunto logico, chiamato *Open World Assumption*:

“The “Open World Assumption” is a term from knowledge base systems. It characterizes knowledge base systems that assume the information stored is incomplete relative to the universe of discourse they intend to describe. This incompleteness may be due to the inability of the maintainer to provide sufficient information or due to more fundamental problems of cognition in the system’s domain.” (CIDOC CRM v. 7.1.2, p. 17)

In altre parole, un'ontologia può essere utilizzata efficacemente, anche se non descrive il proprio dominio in modo completo ed esaustivo. Al contempo, un'ontologia può essere facilmente estesa e ampliata in un secondo momento.

Come in parte anticipato, si può presumere che nei modelli concettuali che si intende realizzare ricorreranno svariati concetti di base, come per esempio quelli di testimone e di lezione. Formalizzando i modelli come ontologie, è possibile sfruttare il lavoro di modellizzazione compiuto da altri studiosi, aggiungendo delle precisazioni o delle distinzioni, per rappresentare al meglio il dominio dal punto di vista di un particolare approccio filologico.

Infine, l'utilizzo delle ontologie permetterebbe ai modelli concettuali prodotti di venire applicati anche nel campo, ancora in fase di definizione e di sperimentazione,¹² delle edizioni semantiche digitali. Per spiegare che cosa si intende con questo termine, si riporta un estratto dell'introduzione all'edizione scientifica digitale semantica del quaderno di Paolo Bufalini, *Appunti 1981-1991*, realizzata dal Digital Humanities Advanced Research Centre dell'Università di Bologna:

“Il risultato è un'edizione basata interamente su *linked open data* che:

- utilizza URI persistenti per identificare le risorse e le relazioni fra di esse (tramite il provider <https://w3id.org>);
- utilizza ontologie esistenti per l'organizzazione delle informazioni [...];
- serve i dati mediante *content negotiation* e li rende interrogabili mediante uno SPARQL *endpoint* dotato di un'interfaccia grafica.”¹³

¹² “The assertive edition is not a well-defined type of scholarly editing yet. However, assertive editions exist. The methods according to which they are created, modelled, and made available online are becoming part of scholarship.” (Vogeler 2019, p. 318).

¹³ Il testo citato è tratto da <http://projects.dharc.unibo.it/bufalini-notebook/introduction>. Per approfondire come è stata realizzata l'edizione semantica digitale del quaderno di Paolo Bufalini si rimanda a Daquino et al. 2019.

L'idea alla base di un'edizione digitale semantica è che il testo edito contenga delle informazioni di interesse scientifico e che applicando le tecnologie del web semantico sia possibile pubblicare e condividere tali informazioni in modo che possano essere utilizzate:

“[assertive editions] help the user/reader understand the text and use the information conveyed in the text as structured data. This data includes interpretations of the text based on the context and the expertise of the editor. In fact, interpretation is part of the core of the critical activity of the editor.” (Vogeler 2019, p. 318)

2.5.2 Il linguaggio di modellizzazione

OWL (Web Ontology Language) è un linguaggio per la definizione di ontologie web e standard mantenuto dal W3C. La prima versione di OWL (OWL 1)¹⁴ è stata pubblicata nel 2004, mentre nel 2012 è stata pubblicata la seconda versione (OWL 2),¹⁵ che consiste in un'estensione e revisione della prima, allo scopo di facilitare lo sviluppo e la condivisione di ontologie attraverso il web. La principale differenza tra le due versioni è che in OWL 2 un'ontologia e ogni elemento al suo interno devono essere identificati mediante degli IRI (*Internationalized Resource Identifiers*). In questo modo sia l'ontologia nella sua interezza sia un singolo elemento sono riferibili dall'esterno facilitando la condivisione e il riuso di ontologie diverse attraverso il web. Per questo motivo OWL 2 è stato individuato come il linguaggio di modellizzazione più adatto agli scopi della presente ricerca, considerato anche che si tratta di un linguaggio molto espressivo e che è stato largamente utilizzato dalla comunità scientifica.

A livello concreto un'ontologia OWL 2 è un documento all'interno del quale sono definite classi, proprietà e le loro istanze attraverso degli elementi (o *tag*) specifici. Queste definizioni sono chiamate “assiomi”.¹⁶ Oltre agli assiomi, un'ontologia può contenere delle annotazioni, per fornire maggiori informazioni sull'ontologia stessa, descrivendo per esempio il dominio che descrive e le finalità per cui può essere utilizzata. Per essere formalmente valida, un'ontologia OWL 2 deve essere un'istanza della classe owl:Ontology, con associato un IRI che individua l'ontologia stessa e un IRI che indica

¹⁴ <https://www.w3.org/TR/2004/REC-owl-features-20040210/>.

¹⁵ <https://www.w3.org/TR/owl2-overview/>.

¹⁶ In generale, in un'ontologia OWL 2, un assioma indica ciò che è da considerarsi vero all'interno di un dominio. Ci sono assiomi generici, che definiscono una classe o una proprietà, per esempio la definizione della classe Persona, e assiomi che riguardano degli specifici individui del dominio, per esempio Paolo è una Persona. Questa seconda tipologia di assioma è detta anche “affermazione” (*assertion*) o “fatto” (<https://www.w3.org/TR/owl2-syntax/#Assertions>).

la versione dell'ontologia. Infine, OWL 2 consente di importare all'interno di un'ontologia altre ontologie, favorendo in questo modo il riuso.

Il linguaggio OWL 2 non richiede che gli assiomi vengano ordinati in modo particolare. Per una questione di leggibilità, si tende a raggruppare insieme le diverse tipologie di assiomi, ponendo per esempio prima tutte le classi, poi tutte le proprietà e infine tutte le istanze. Un'altra prassi comune è quella di organizzare gli assiomi dello stesso tipo in ordine alfabetico o secondo il criterio logico ritenuto più opportuno da chi edita il documento.

La semantica formale di OWL 2 specifica come derivare delle conseguenze logiche da un solo documento OWL oppure da molteplici documenti OWL che sono stati combinati insieme grazie ai meccanismi del linguaggio.

2.5.3 I limiti di un'ontologia

L'aspetto che rimane da valutare è se e quanto sia facile ed efficace integrare un modello concettuale espresso sotto forma di ontologia nel procedimento di sviluppo di un software. In altre parole, dal modello concettuale espresso come ontologia è possibile derivare le altre due tipologie di modelli, il modello logico e il modello di implementazione? Le premesse teoriche per condurre delle sperimentazioni in questo senso ci sono. Nel campo dell'ingegneria del software, infatti, è stato ideato un approccio di sviluppo detto *Ontology-Driven Software Development (ODSD)*,¹⁷ che ha come obiettivo l'integrazione delle ontologie nelle diverse fasi di produzione del software. L'idea alla base di questo paradigma è quella di sfruttare la potenza espressiva e le funzioni di *reasoning* automatico delle ontologie sia in fase di modellizzazione che in fase di scrittura del codice. Dall'applicazione di questo approccio deriverebbero diversi vantaggi, in particolare in termini di riuso della conoscenza acquisita e modellata nell'ontologia per produrre strumenti informatici diversi:

“An important benefit of using an ontology at development time is that it enables the developer to practice a “higher” level of reuse than is usually the case in software engineering (i.e. knowledge reuse instead of software reuse). Moreover, it enables the developer to reuse and share application domain knowledge using a common vocabulary across heterogeneous software platforms.” (Guarino 1998, p. 9)

¹⁷ Per approfondire i principi alla base dell'ODSD si rimanda a Guarino 1998 e Pan et al. 2013.

E il riuso degli strumenti informatici e dei rispettivi codici sorgente stessi:

“An important point to keep in mind in ontology-driven IS [(information system)] development is that an ontology not only can be used to build a new IS (IS engineering), but it can also be used equally profitably for IS re-engineering, in order to increase reuse and maintainability” (*ibid.*)

Un’obiezione alla scelta dell’ontologia come modalità di formalizzazione è che nel campo dell’ingegneria del software il linguaggio di modellizzazione più noto e utilizzato è UML. L’utilizzo di un’ontologia per descrivere il modello concettuale potrebbe essere limitante e poco pratico per diversi sviluppatori software. Considerando, invece, quanto l’utilizzo di ontologie faciliterebbe il lavoro di modellizzazione agli studiosi, favorendo un approccio collaborativo e improntato al riuso, una possibile soluzione potrebbe essere quella di utilizzare entrambi i linguaggi, OWL per i modelli concettuali e UML per i modelli logici e implementativi. Per derivare quest’ultimi dai modelli concettuali l’ideale sarebbe poter disporre di strumenti che effettuino una trasformazione automatica. A conoscenza di chi scrive, degli strumenti simili non sono stati sviluppati. L’OMG ha tuttavia elaborato l’Ontology Definition Metamodel (ODM), che comprende una serie di mappature tra OWL e UML che permettono di modellizzare alcune ontologie attraverso strumenti basati su UML.¹⁸ Uno di questi strumenti è OWLGrEd, descritto nel dettaglio nella sezione 3.2, che offre un’interfaccia grafica per sviluppare un’ontologia in OWL in forma di diagrammi UML. Strumenti come OWLGrEd possono risultare preziosi per effettuare il passaggio summenzionato.

2.5.4 Cenni sul linguaggio alla base di un’ontologia OWL 2

In questa sezione si presentano i termini alla base della modellizzazione di un dominio in forma di ontologia, e i corrispondenti componenti in OWL 2, allo scopo di facilitare la comprensione del modello concettuale di esempio presentato nel prossimo capitolo.¹⁹

¹⁸ “The Ontology Definition Metamodel (ODM), as defined in this specification, is a family of MOF metamodels, mappings between those metamodels as well as mappings to and from UML, and a set of profiles that enable ontology modeling through the use of UML-based tools. The metamodels that comprise the ODM reflect the abstract syntax of several standard knowledge representation and conceptual modeling languages that have either been recently adopted by other international standards bodies (e.g., RDF and OWL by the W3C), are in the process of being adopted (e.g., Common Logic and Topic Maps by the ISO) or are considered industry de facto standards (non-normative ER and DL appendices).” (ODM v. 1.1, p. 20)

¹⁹ Qualora il lettore avesse già dimestichezza con tale terminologia, può procedere alla seconda parte del capitolo. In caso contrario, si consiglia la lettura della presente sezione e al bisogno un ulteriore approfondimento. Per approfondire si

Come ogni altro tipo di modello, anche un'ontologia è caratterizzata da un **dominio**, che individua l'ambito o l'aspetto del mondo reale che si intende descrivere, e da uno **scope** (termine inglese traducibile con 'estensione'), che precisa l'estensione del dominio che l'ontologia intende ricoprire.

Gli **oggetti** del dominio che condividono una o più caratteristiche comuni sono categorizzabili in una **classe**. Queste caratteristiche o proprietà devono essere riconosciute dagli esperti del dominio come gli elementi distintivi di tale classe di oggetti. Un oggetto del mondo reale che appartiene a una classe è detto istanza della classe. Per es., per rappresentare in un'ontologia il concetto di scuola, si può definire la classe `Alunno`, mentre `Mario Rossi` è un'istanza della classe `Alunno`.

Le classi possono essere organizzate gerarchicamente in base al loro livello di astrazione o di dettaglio, per cui alcune classi (dette **superclassi**) possono comprendere al loro interno classi diverse (dette **sottoclassi**). Per esempio, la classe `Alunno` e la classe `Professore` sono sottoclassi di `Persona`. Allo stesso modo si può dire che `Persona` è la superclasse di `Alunno` e di `Professore`. Una sottoclasse, per essere tale, deve presentare tutte le caratteristiche definite per la propria superclasse e almeno una caratteristica in più. Una sottoclasse eredita, dunque, tutte le proprietà definite per la propria superclasse. Questo principio logico è chiamato '**ereditarietà**'. Una classe può avere più superclassi.

In OWL 2 la classe è definita con l'elemento `<owl:Class>`, mentre un'istanza con l'elemento `<owl:NamedIndividual>`. Per stabilire che una classe è sottoclasse di un'altra classe, nella definizione della prima si aggiunge l'elemento `<rdfs:subClassOf>`.

Una **proprietà** definisce il tipo di rapporto che unisce due classi, per esempio: la proprietà `insegna` associa `Pico De Paperis`, istanza della classe `Professore` a `Mario Rossi`, istanza della classe `alunno`.

In alcuni casi, una proprietà unisce due istanze diverse della medesima classe (**proprietà riflessiva**), per esempio la proprietà `compagno di classe` associa `Mario Rossi` a `Sara Bianchi`, due diverse istanze della classe `Alunno`. Le proprietà molto spesso vengono denominate con dei predicati, così da poter formare delle connessioni logiche immediatamente comprensibili tra

rimanda alla sezione dedicata alla terminologia all'interno della documentazione di CIDOC CRM v. 7.2.2 e alla specifica della sintassi di OWL 2.

la classe soggetto (detta *domain*) e la classe oggetto (detta *range*), analogamente alle triple semantiche di RDF.²⁰ In OWL 2 per unire due classi diverse o due istanze della medesima classe si usa `<owl:ObjectProperty>`. La classe soggetto si individua con `<rdfs:domain>`, mentre la classe oggetto con `<rdfs:range>`. Riprendendo l'esempio della proprietà `insegna`, `Professore` è la classe dominio, mentre `Alunno` è la classe *range*.

Una proprietà esprime un collegamento tra due classi che va in una sola direzione: dalla classe soggetto (o dominio) alla classe oggetto (o *range*). Per rappresentare il rapporto tra le due classi nella direzione opposta è necessario definire una nuova proprietà associata alla precedente mediante `<owl:inverseOf>`.

Anche le proprietà come le classi possono essere organizzate in rapporto gerarchico, grazie a `<owl:subPropertyOf>`. Inoltre, una proprietà può essere usata per collegare più istanze di una classe a molteplici istanze di un'altra classe. Per esempio la proprietà `insegna` può essere usata per associare a un'istanza della classe `Professore` molteplici istanze della classe `Alunno`. In OWL 2 è possibile determinare quante istanze della classe soggetto e della classe oggetto sono collegabili tra di loro da un punto di vista logico.

Quando i concetti rappresentati in un'ontologia coincidono con quelli definiti in altre ontologie, è possibile mappare le ontologie tra di loro, stabilendo delle equivalenze sia a livello di classe, associando a `<owl:class>` una o più `<owl:equivalentClass>`, sia a livello di proprietà, associando `<owl:objectProperty>` a `<owl:equivalentProperty>`. Quando un'ontologia descrive lo stesso dominio di un'altra ontologia in modo più dettagliato o, al contrario, in modo più astratto, le classi della prima possono essere definite rispettivamente come delle sottoclassi o come delle superclassi della seconda.

2.5.5 Ontologie attinenti il campo della filologia e della critica testuale

Come anticipato, uno dei vantaggi nello sviluppare i modelli concettuali in forma di ontologie è quello di potersi servire di ontologie esistenti, in un'ottica di riutilizzo e di valorizzazione della ricerca già

²⁰ RDF (Resource Description Framework) è un modello standard promosso dal W3C per rappresentare le informazioni sul web in forma di grafi costituiti da oggetto, soggetto e predicato (detti anche "triple"). <https://www.w3.org/TR/rdf12-concepts/>.

svolta da altri studiosi. Negli ultimi anni sono state prodotte diverse ontologie per l'ambito umanistico in generale, così come nello specifico per il campo della filologia e della critica testuale. Queste ontologie costituiscono un punto di riferimento prezioso per realizzare la rete di modelli concettuali. In questa sezione vengono presentate nel dettaglio, fornendo per ciascuna una breve descrizione, che metta in evidenza sia i limiti sia i punti di forza.

2.5.5.1 CIDOC Conceptual Reference Model (CRM)

CRM²¹ è un'ontologia sviluppata dall'International Committee for Documentation della ICOM (International Council of Museums), il cui obiettivo principale è quello di consentire lo scambio e l'integrazione tra informazioni provenienti da fonti diverse e che riguardino lo studio del passato:

“The primary role of the CIDOC CRM is to enable the exchange and integration of information from heterogeneous sources for the reconstruction and interpretation of the past at a human scale, based on all kinds of material evidence, including texts, audio-visual material and oral tradition. It starts from, but is not limited to, the needs of museum documentation and research based on museum holdings. It aims at providing the semantic definitions and clarifications needed to transform disparate, localised information sources into a coherent global resource, be it within a larger institution, in intranets or on the Internet, and to make it available for scholarly interpretation and scientific evaluation.” (CIDOC CRM v. 7.1.2, p. 9)

CRM è definito dai propri autori come una “ontologia formale”,²² ovvero la specifica di un insieme di concetti denominati utilizzati per descrivere e approssimare una parte della realtà (cfr. CIDOC CRM v. 7.1.2, p. 25).

Lo sviluppo di CRM è stato avviato nel 1996, mentre l'ultima versione del modello (la 7.2.2.) è stata pubblicata a ottobre del 2022. CRM è anche uno standard ISO.

Il modello è stato utilizzato prevalentemente nel campo dei beni culturali da musei e istituzioni. Tuttavia, CRM si presta a essere applicato a campi del sapere diversi e soprattutto a essere esteso grazie alla sua organizzazione logica e alla definizione di classi astratte in grado di comprendere oggetti di domini diversi.²³ La prima classe definita nel modello, infatti, è denominata E1 CRM Entity, che

²¹ Sito web del CIDOC CRM: <https://cidoc-crm.org/>.

²² Per approfondire il concetto di “ontologia formale” si rimanda a Guarino 1998.

²³ “A sufficient condition for the compatibility of an extension with the CIDOC CRM is that its classes, other than E1 CRM Entity, subsume all classes of the extension, and all properties of the extension are either subsumed by CRM properties” (CIDOC CRM v. 7.1.2, p. 28)

comprende: “all things in the universe of discourse of the CIDOC Conceptual Reference Model.” (CIDOC CRM v. 7.1.2, p. 57).

CRM sembra essere stato ideato per rispondere ad alcune delle esigenze che hanno guidato anche la presente ricerca:

“The CIDOC CRM aims to support the following specific functionalities:

- Inform developers of information systems as **a guide to good practice in conceptual modelling**, in order to effectively structure and relate information assets of cultural documentation.
- **Serve as a common language for domain experts and IT developers to formulate requirements and to agree on system functionalities** with respect to the correct handling of cultural contents. [...]

(CIDOC CRM v. 7.1.2, p. 9, *mia enfasi*)

Considerata l'autorevolezza del CRM e il suo largo utilizzo nella comunità scientifica, sembra opportuno sviluppare i modelli concettuali in modo tale che possano essere compatibili con il CRM. Procedendo in questo modo, le ontologie elaborate presenterebbero due vantaggi: il dominio e le classi sarebbero inquadrati più facilmente, facendo riferimento alle definizioni più ampie ed astratte fornite dal CRM; in un'ottica futura, le ontologie, qualora dovessero raggiungere uno stadio più stabile di sviluppo e rivelarsi uno strumento utile per la comunità scientifica internazionale, potrebbero essere proposte come una possibile estensione del CRM. Tuttavia, questo obiettivo è secondario rispetto alla definizione di modelli concettuali in grado di rappresentare il dominio in modo soddisfacente.

2.5.5.2 Critical Apparatus Ontology (CAO)

La CAO è stata sviluppata da Francesca Giovannetti e pubblicata online nel 2019 in formato OWL. Basata sulle linee guida TEI, rappresenta l'apparato delle varianti testuali e le argomentazioni filologiche che accompagnano un'edizione critica digitale.

La CAO descrive solo una parte del dominio della presente ontologia, adottando un punto di vista molto preciso, quello delle linee guida TEI, e “in line with the philosophy of the Text Encoding Initiative (TEI), CAO does not commit to any specific school of textual criticism” (Giovannetti 2019). L'assunto di fondo della presente strategia di modellizzazione è, al contrario, che i modelli concettuali debbano essere sviluppati secondo un determinato approccio filologico, così da riflettere

in modo più preciso e coerente il punto di vista dell'esperto del dominio. Ad ogni modo, molti dei concetti e delle relazioni modellati nella CAO si adattano anche alla descrizione di un'edizione critica realizzata secondo l'approccio neo-lachmanniano, l'approccio scelto per realizzare il modello concettuale di esempio. Inoltre, l'obiettivo di fondo della CAO è allineato con una delle applicazioni auspiccate anche per la presente ricerca:

“The development of CAO aims at being a first step towards **the publication of data-centric digital critical editions to facilitate the federation of cultural resources on the web**” (CAO, *mia enfasi*).²⁴

La CAO incorpora classi e proprietà di altre ontologie, tra cui: la Web Annotation Ontology, PROV-O, FRBR_{OO} e la HiCO. La dipendenza da FRBR_{OO} è problematica: la CAO prevede, infatti, dei collegamenti a classi dell'ontologia FRBR_{OO} che sono state deprecate con la nuova versione del modello, LRM_{OO}, presentato nella prossima sezione. Se la CAO sarà aggiornata rispetto al modello LRM_{OO}, allora potrà essere importata senza problemi anche nei modelli concettuali, altrimenti conviene utilizzare questa ontologia solo come riferimento teorico.

2.5.5.3 LRM_{OO} (formerly FRBR_{OO}) object-oriented definition and mapping from IFLA LRM

LRM_{OO}²⁵ è un'estensione del CIDOC-CRM, che rappresenta sotto forma di ontologia il Library Reference Model (LRM) elaborato dall'IFLA,²⁶ modello concettuale che “mira a rendere espliciti i principi generali che governano la struttura logica delle informazioni bibliografiche” (Bruni 2017, p. 9).

LRM_{OO} è una revisione del precedente FRBR_{OO} da parte di un gruppo di lavoro composto da membri dell'IFLA e di CIDOC, dopo che l'IFLA nel 2016 ha sostituito con LRM i precedenti *Functional Requirements for Bibliographic Records* (FRBR).

Il nucleo logico di LRM_{OO} consiste in quattro concetti definiti da LRM, che nel dettaglio sono:

- *work* (opera), “Il contenuto intellettuale o artistico di una determinata creazione” (Riva et al. 2017, p. 21);

²⁴ In merito a questo obiettivo la CAO e la HiCO sono state utilizzate per la realizzazione di un'edizione semantica come illustrato in Daquino, Giovannetti, e Tomasi 2019.

²⁵ Sito web del modello FRBR_{OO}/LRM_{OO}: <https://cidoc-crm.org/frbroo/>.

²⁶ International Federation of Library Associations and Institutions, <https://www.ifla.org/>.

- *expression* (espressione), “Una determinata combinazione di segni che veicola un contenuto intellettuale o artistico” (*ibid.*, p. 23);
- *manifestation* (manifestazione), “Un insieme di tutti i supporti che si presume condividano le stesse caratteristiche per quanto riguarda contenuto intellettuale o artistico e aspetti di forma fisica. L’insieme è definito sia dal contenuto complessivo sia dal piano produttivo del supporto o dei supporti” (*ibid.*, p. 25);
- *item* (singolo oggetto), “Un oggetto o oggetti che recano segni destinati a veicolare un contenuto intellettuale o artistico” (*ibid.*, p. 27).

Questi concetti sono utili anche per la descrizione di alcune parti fondamentali del dominio dei modelli concettuali che si intende realizzare con la presente strategia di modellizzazione. Inoltre, considerando quanto diffusamente il modello proposto dall’IFLA sia stato applicato in campo biblioteconomico, è opportuno fare riferimento alle definizioni in esso contenute anche nello sviluppo del modello concettuale di esempio.

Essendo LRM₀₀ un’estensione di CIDOC CRM, necessita di essere implementata unitamente a CRM. Le classi che rappresentano i concetti di opera (F1 Work), espressione (F2 Expression) e manifestazione (F3 Manifestation) sono sottoclassi delle classi E89 Propositional Object e E73 Information Object, come illustrato in figura.

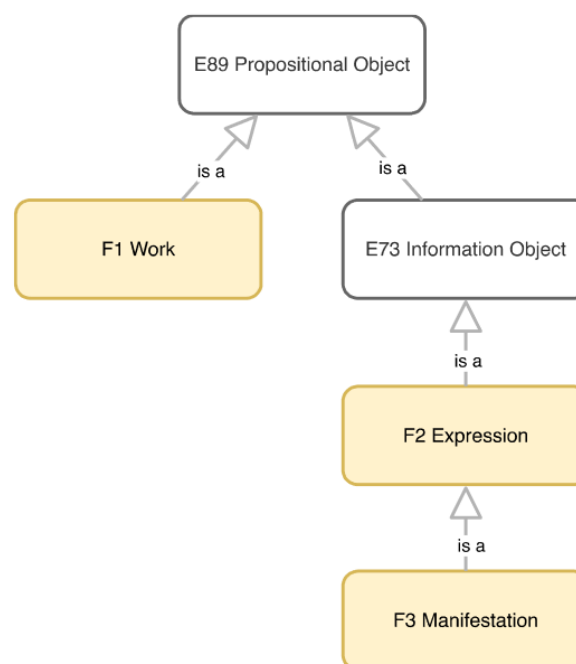


Figura 2.8 Schema del rapporto tra le classi principali di LRM₀₀ e CIDOC-CRM (LRM₀₀ v. 0.9, p. 10)

L'ultima versione della specifica di LRM_{OO} è la 0.9.3, pubblicata a febbraio del 2023. Al momento LRM_{OO} non è ancora stato pubblicato in una versione definitiva, ma solo come bozza in attesa di essere tradotta in RDF. Tuttavia, in un'ottica futura, conviene che i modelli concettuali rimandino alle classi e alle proprietà di LRM_{OO}, invece di quelle di FRBR_{OO}, considerando anche il fatto che il primo risulta essere molto più semplice e conciso rispetto al secondo.

2.5.5.4 Historical Context Ontology (HiCO)

La HiCO²⁷ è un'ontologia in OWL realizzata da Marilena Daquino e pubblicata nel 2020 allo scopo di rappresentare le attività ermeneutiche degli studiosi che generano una nuova informazione, e di renderle verificabili.²⁸ La principale classe definita nella HiCO è "Interpretation Act", definita "The hermeneutical activity performed by an agent in order to generate new information (e.g. an artwork attribution)" (HiCO).²⁹ Il dominio rappresentato dall'ontologia è piuttosto ristretto, ma applicabile in diversi ambiti di ricerca, tra cui anche nel campo della critica testuale. Nella CAO, infatti, alcune classi e proprietà della HiCO sono utilizzate per rappresentare il lavoro di interpretazione svolto dall'editore di un'edizione critica.

2.5.5.5 PROV-O

The PROV-Ontology³⁰ è una delle diverse specifiche sviluppate dal W3C per descrivere la provenienza (*provenance*) delle informazioni pubblicate sul web, in modo tale da poterne valutare l'affidabilità. Nella descrizione del PROV-DM³¹ (dove DM sta per *data model*) si legge:

"provenance is defined as a record that describes the people, institutions, entities, and activities involved in producing, influencing, or delivering a piece of data or a thing. In particular, the provenance of information is crucial in deciding whether information is to be trusted, how it should be integrated with other diverse information sources, and how to give credit to its originators when re-using it. In an open and inclusive environment such as the Web, where users find information that is often contradictory or questionable, provenance can help those users to make trust judgements."

²⁷ <https://marilenadaquino.github.io/hico/>.

²⁸ Cfr. "it addresses features characterising hermeneutical activities performed by scholars while generating new information (i.e. an interpretation act). It allows to represent and reason on reliability of argumentations around attributions, by evaluating features such as motivations, types of cited sources or criteria, dates, relations with other claims (e.g. agreement/disagreement)" (HiCO).

²⁹ <https://marilenadaquino.github.io/hico/#d4e229>.

³⁰ <https://www.w3.org/TR/prov-o/>.

³¹ <https://www.w3.org/TR/2013/REC-prov-dm-20130430/#dfn-provenance>.

Le tre classi alla base dell'ontologia sono: "Entity" (entità), un oggetto fisico, digitale, concettuale o di altro tipo, che può essere reale o immaginario e che presenta delle caratteristiche fisse; "Activity" (attività), qualcosa che si verifica per un periodo di tempo e che agisce su delle entità o con esse, può consistere, per esempio, nel consumo, nell'elaborazione, nella modifica, nella trasformazione, ecc. di entità; "Agent" (agente), qualcosa o qualcuno che detiene una qualche forma di responsabilità riguardo un'attività o un'entità.

PROV-O è utilizzata sia nella CAO che nella HiCO, in quanto offre molte proprietà diverse per descrivere l'origine di un'informazione di valore scientifico, partendo dalle relazioni che intercorrono tra enti, attività e agenti. L'ultima versione dell'ontologia è stata pubblicata nel 2013.

2.5.5.6 Scholarly Editing

L'ontologia Scholarly Editing (SE) è stata sviluppata da Hans Cools, Roberta Padlina e altri collaboratori, contestualmente al progetto NIE-INE (vd. 1.3.6) il cui obiettivo principale era quello di utilizzare le tecnologie del web semantico per l'espressione formale di edizioni critiche, in modo tale da consentire l'interoperabilità tra progetti di edizione diversi e, nel lungo termine, creare "a semantic space for an interdisciplinary formal knowledge domain for the Humanities" (Cools e Padlina 2021, p. 99). Oltre all'ontologia "scholarly editing" il team di Cools ha sviluppato, quindi, molte altre ontologie, di cui alcune funzionali alla descrizione del "domain for the Humanities" a un livello più alto e astratto, e altre, invece, realizzate *ad hoc* per le esigenze scientifiche di specifici progetti di edizione. Tutte le ontologie sono disponibili su GitHub.³² La figura sotto rappresenta in forma schematica le ontologie realizzate nell'ambito di NIE-INE e i loro rapporti reciproci.

³² Collegamento al repository GitHub in cui sono state pubblicate le ontologie sviluppate nel contesto di NIE-INE: <https://github.com/nie-ine/Ontologies>.

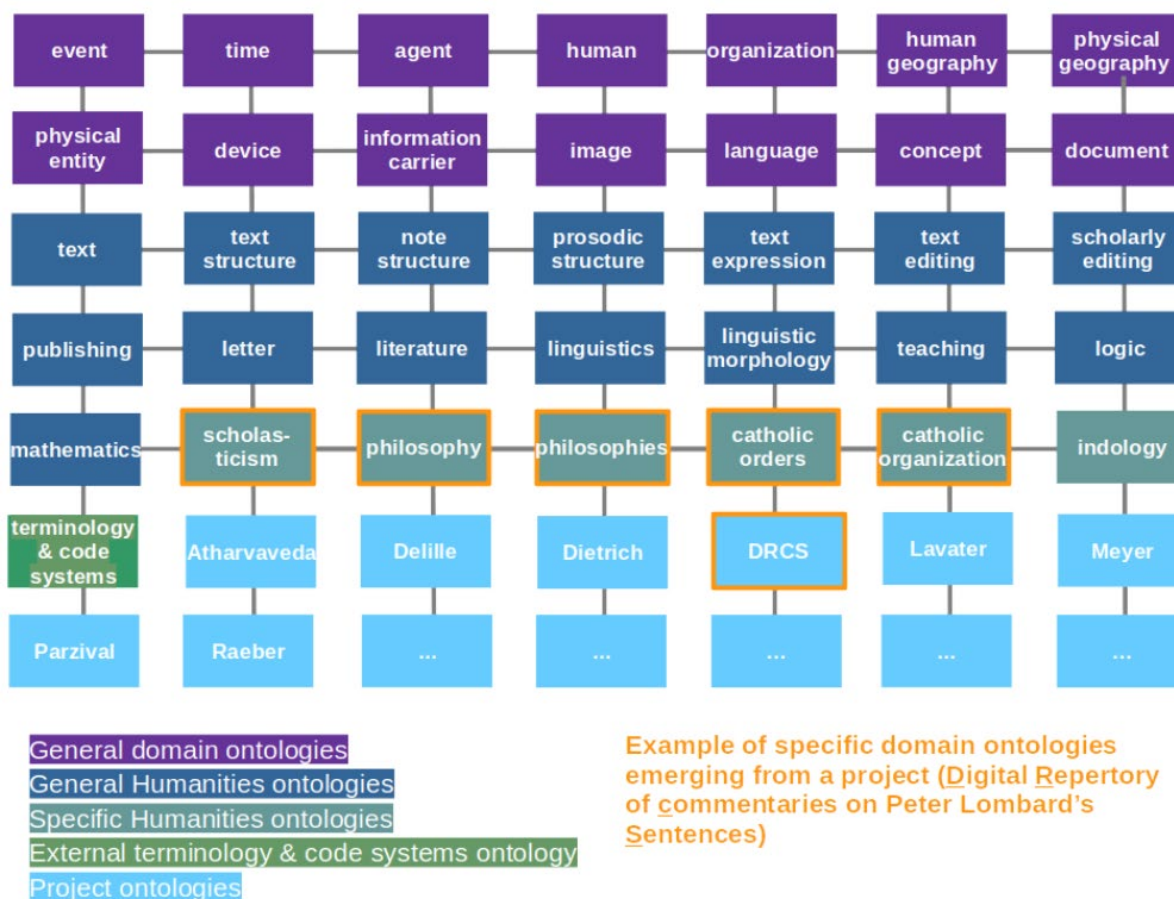


Figura 2.9 Schema delle ontologie sviluppate nell'infrastruttura NIE-INE (Cools e Padlina 2021).

L'ontologia SE offre un insieme di classi e proprietà per descrivere i contenuti di un'edizione critica e, in generale, l'attività di *editing*.³³ Tra tutte le ontologie analizzate questa è sicuramente quella più compatibile rispetto al dominio e agli obiettivi della strategia di modellizzazione. Tuttavia, si tratta di un'ontologia non più mantenuta. Il progetto NIE-INE, infatti, è terminato di recente e purtroppo la relativa ricerca sull'applicazione delle tecnologie del web semantico nel contesto delle edizioni critiche non è più proseguita in seguito alla morte improvvisa di Cools.

2.5.5.7 SPAR Ontologies

Le SPAR (Semantic Publishing and Referencing) Ontologies³⁴ sono una suite di ontologie in OWL 2, tra di loro complementari, che permettono di descrivere tutti gli aspetti che riguardano il processo

³³ Lo sviluppo di questa ontologia sembrerebbe essere stato sospeso, in quanto non figura più sul sito ufficiale dell'associazione e-editiones. Il sito è comunque accessibile ricorrendo agli *snapshots* di Web Archive. L'ontologia è stata comunque presa a riferimento in quanto è l'unica disponibile che cerca di descrivere un'edizione critica nella sua interezza. <https://web.archive.org/web/20210614221418/https://e-editiones.ch/ontology/scholarly-editing>.

³⁴ <http://www.sparontologies.net/ontologies>.

editoriale, attraverso delle asserzioni *machine-readable* in formato RDF. L'architettura delle SPAR ontologies, riportata in figura, comprende diciassette ontologie diverse, di cui alcune sono estensioni di altre.

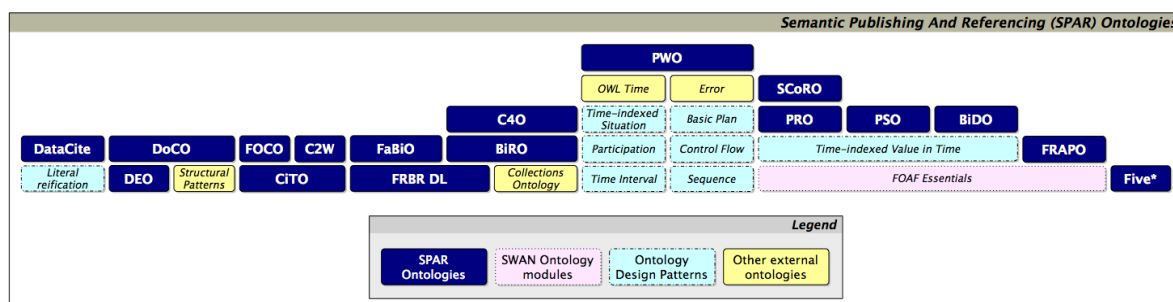


Figura 2.10 Architettura delle SPAR Ontologies.

Rispetto al dominio che si intende modellare con la presente strategia, le due ontologie più affini sono:

- la FRBR-aligned Bibliographic Ontology (FaBiO),³⁵ basata sui concetti principali di FRBR, mediante la quale è possibile descrivere delle entità pubblicate o pubblicabili, come per esempio un articolo, un libro, ecc.;
- e la Document Components Ontology (DoCO),³⁶ che fornisce i termini per descrivere i componenti di un documento, sia da un punto di vista strutturale (per esempio i concetti di paragrafo, sezione, capitolo, ecc.) sia da un punto di vista retorico (l'introduzione, l'appendice, i riconoscimenti, ecc.).

Entrambe le ontologie, ma in particolar modo la DoCO, sono utili per inquadrare che cos'è un'edizione critica e da quali elementi è costituita.

2.5.5.8 Web Annotation Ontology

La Web Annotation Ontology³⁷ è stata sviluppata sulla base del Web Annotation Data Model,³⁸ lo standard raccomandato dal W3C per rappresentare le annotazioni sul web. Alla base del modello c'è il concetto di annotazione, intesa come il rapporto tra due risorse, rappresentabile in forma di grafo

³⁵ <https://sparontologies.github.io/fabio/current/fabio.html>.

³⁶ <https://sparontologies.github.io/doco/current/doco.html>.

³⁷ <https://www.w3.org/ns/oa#>.

³⁸ <https://www.w3.org/TR/annotation-model/>.

orientato (come nella figura sotto). Le due risorse sono un singolo testo, che costituisce il corpo dell'annotazione (“body”), e l'oggetto dell'annotazione (“target”).

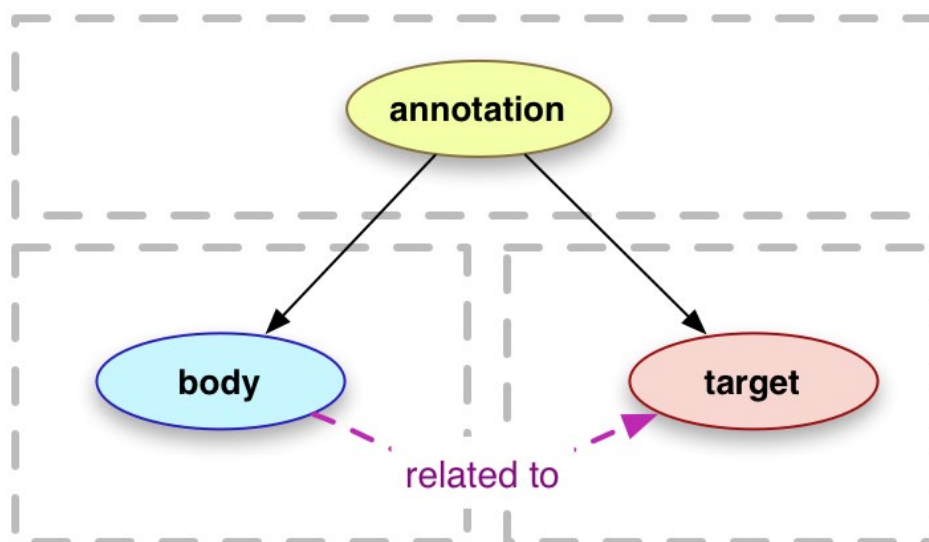


Figura 2.11 Il concetto di annotazione secondo il Web Annotation Data Model.

L'ontologia permette di descrivere il concetto di annotazione in modo dettagliato, specificando le diverse tipologie di annotazione, la motivazione per cui l'annotazione è stata creata e le diverse modalità in cui l'annotazione può essere collegata al proprio oggetto.

L'ultima versione della Web Annotation Ontology è stata pubblicata nel 2017.

Capitolo 3

Il modello concettuale di esempio

In questo capitolo si presenta il modello concettuale sviluppato per verificare l'efficacia della strategia di modellizzazione. Il modello concettuale mira a descrivere i componenti di un'edizione critica, realizzata secondo l'approccio filologico detto "neo-lachmanniano", come illustrato nel dettaglio nella prossima sezione.

Lo sviluppo del modello concettuale di esempio ha permesso di determinare diversi aspetti per mettere concretamente in atto la strategia di modellizzazione, ovvero:

- le diverse fasi di lavoro necessarie;
- gli strumenti informatici utili per la realizzazione di un'ontologia in OWL 2;
- le ontologie esistenti che si prestano alla descrizione del campo della critica testuale e delle edizioni scientifiche digitali, descritte nel precedente capitolo.

Il risultato ottenuto è un'ontologia in OWL 2 denominata "Critical Edition Ontology", consultabile integralmente in forma di documento OWL (Appendice C) oppure focalizzandosi sulle definizioni fornite al suo interno (Appendice B). L'ontologia è consultabile anche *online* al seguente indirizzo: purl.org/critical-edition-ontology/lode.

L'ontologia è composta da tre parti, ciascuna descrive il concetto di edizione critica da una prospettiva diversa: una prospettiva esterna, per inquadrare i concetti basilari del modello, ovvero l'edizione, il suo oggetto, i suoi responsabili e il pubblico cui si rivolge (sezione 3.5); una prospettiva che indaga a fondo il rapporto tra il testo critico e la tradizione dell'opera (sezione 3.6.3); un'ultima prospettiva che analizza i componenti di un'edizione critica, intesa come un prodotto destinato alla pubblicazione (sezione 3.6).

Il modello concettuale presentato è da considerarsi come il primo prodotto di una ricerca che necessita di essere ulteriormente sviluppata. Auspicabilmente, il modello concettuale allo stato attuale risulterà essere comunque una base efficace per avviare un dibattito all'interno della comunità

scientifico su quali siano i requisiti essenziali per lo sviluppo di strumenti informatici per la visualizzazione di un'edizione scientifica digitale.

3.1 Il dominio e lo scopo del modello

Il presente modello si propone di rappresentare il concetto di edizione critica, intesa come il prodotto finale di un processo editoriale, condotto secondo i principi della critica del testo, allo scopo di restituire un testo affidabile e scientificamente fondato di un'opera letteraria. Con "processo editoriale" non si intende il lavoro di pubblicazione svolto da una casa editrice, ma l'analisi e lo studio condotti dal filologo per preparare il testo critico, e tutti i materiali di corredo che costituiscono la sostanza di un'edizione critica. Il dominio è, dunque, quello dell'edizione critica, il modello editoriale che costituisce uno dei due pilastri della filologia "codex-based" (vd. sezione 2.1), e del suo allestimento.

La base teorica del modello e, di conseguenza, anche il suo *scope* sono stati circoscritti alle pratiche che si sono affermate nel corso dello scorso secolo perlopiù tra studiosi italiani di opere medievali in lingua romanza. Tali pratiche si basano su una versione aggiornata del "metodo stemmatico", che vede ridimensionata l'applicabilità dei criteri meccanici dipendenti dallo stemma e che è stata etichettata come "neo-lachmannismo" (cfr. Contini 2014, p. 11) o "approccio neo-lachmanniano". La scelta di questo particolare settore della disciplina filologica è stata dettata principalmente da ragioni pratiche: da un lato in virtù della mia formazione accademica incentrata su studi in cui tale metodo è stato prevalente, dall'altro in quanto ho avuto la possibilità di confrontarmi agevolmente con esperti di quel determinato campo di studi. Ad ogni modo, la maggior parte dei concetti e delle relazioni descritti nel modello si prestano alla descrizione di edizioni critiche prodotte con altre finalità e in altri ambiti della filologia.

Il modello si concentra sui concetti di base di un'edizione critica, a prescindere che sia stampata su carta o pubblicata in forma digitale. Nonostante lo scopo ultimo del modello sia quello di guidare lo sviluppo di uno strumento di visualizzazione di edizioni critiche digitali di stampo neo-lachmanniano, il modello definisce per il momento dei concetti generici e molto astratti comuni sia alle edizioni a stampa sia a quelle digitali. Per questo motivo il modello realizzato costituisce una base sulla quale poter elaborare ulteriormente eventuali requisiti specifici di un'edizione digitale.

3.1.1 Una scelta controcorrente?

La scelta di avviare la strategia di modellizzazione da una tipologia di edizione nettamente prevalente nel campo delle edizioni a stampa, ma che non ha riscosso il medesimo successo nel web, potrebbe sembrare controintuitiva. Nel web, infatti, il paradigma editoriale che si è affermato maggiormente è quello non ricostruttivo, consolidandosi nella forma della *digital documentary edition*.¹ Le ragioni di questo successo si possono ascrivere inizialmente alla facilità con la quale nel web è possibile offrire al lettore riproduzioni fotografiche ad alta risoluzione dei manoscritti, laddove le edizioni in facsimile prodotte a mezzo a stampa risultano essere, invece, poche e molto costose. Gradualmente nelle edizioni scientifiche digitali si è preferito porre maggiormente attenzione al documento invece che al testo ricostruito dall'editore. Tuttavia, la *constitutio textus* è tuttora percepita dalla maggior parte dei filologi come una fase fondamentale del lavoro editoriale, che non può essere trascurata. L'allestimento di un unico testo potrebbe sembrare una mera esigenza grafica, dovuta ai limiti materiali delle edizioni stampate su libri: non potendo offrire in un unico volume tutte le trascrizioni di tutti i testimoni, l'editore seleziona e condensa le informazioni più rilevanti all'interno di un'edizione critica. Sicuramente questo aspetto è vero e ha influito sulla codifica di un complesso sistema di simboli, abbreviazioni e rimandi, comunemente chiamato "apparato critico". Il testo critico, però, non ricopre una funzione esclusivamente documentaria, in quanto costituisce un'argomentazione scientifica, un'approssimazione ragionata all'"originale" di un testo. La ricostruzione del testo da parte del filologo suggella il proprio lavoro scientifico e mette a disposizione degli altri studiosi in modo chiaro un'ipotesi di lavoro. Inoltre, permette ai non addetti ai lavori di fruire di un'opera. Adempie, dunque, a una funzione ben distinta rispetto alla documentazione della tradizione del testo e, in virtù di questa distinzione, ha una sua ragion d'essere anche nel web.

¹ Per spiegare il concetto di *digital documentary edition* si riporta *in primis* la definizione di "documentary edition" tratta dal *Parvum Lexicon Stemmatologicum* (PLS): "A documentary edition is an edition based on a single manuscript, often the supposedly best manuscript, the codex optimus, but in some cases also a manuscript of particular literary or linguistic value. In the latter case, the codex optimus will usually have been edited, so making a new documentary edition is a way of supplementing the editions of the work in question. Quite a few editions of vernacular texts are documentary since they primarily are intended for linguistic studies." Questo tipo di edizione ha riscontrato un notevole successo nel campo della filologia digitale, nella misura in cui vi rientrano molte delle edizioni scientifiche digitali ad oggi pubblicate nel web. L'etichetta "digital documentary edition" è diventata parte del linguaggio corrente della filologia digitale, prevalentemente grazie agli studi di Elena Pierazzo (cfr. Pierazzo 2011 e Pierazzo 2014a).

Premettendo, dunque, che entrambi i modelli editoriali, l'edizione critica da un lato e quella diplomatica o documentaria dall'altro, sono entrambi necessari e complementari, occorre domandarsi come migliorare e potenziare l'edizione critica come modello editoriale per la realizzazione di edizioni scientifiche digitali.

Nell'era dei principi FAIR,² le edizioni di tipo ricostruttivo sembrano peccare di poca trasparenza. Appiattite in una sequenza di caratteri definita "testo critico", le lezioni derivate dai testimoni e le lezioni congetturali o le altre modifiche effettuate dal filologo non risultano immediatamente riconoscibili. Oltre a ciò l'apparato critico è uno strumento di lavoro che può risultare ermetico, mentre nelle edizioni documentarie, la riproduzione fotografica del testimone fa da garante della verità. Per rimediare a questi limiti dell'edizione critica, nella sua forma digitale il testo critico è stato proposto come un testo molteplice, a strati. Ma in questo modo il testo critico ha perso di autorevolezza:

"Within the typographic paradigm, the edited text is by far the most important feature, the core and the exclusive centre of the edition. All other forms of evidence, such as illustrative images, bibliographic information, details of script and typesetting, variant readings or semantic interpretations, are just *substrata to or fortifications of it*. Within the digital paradigm, the process is reversed: the editor does not write the edited text. Rather, it is developed gradually from the material documents, from visual evidence through the transcription and through the application of critical, historical, stylistic and philological knowledge. In the digital edition there is little reason to hide these other layers of textual representation from the user. But as one effect of this change in methodology, the edited text is relativised and the multiple text is facilitated" (Sahle 2016, mia enfasi).

La scelta di avviare la strategia di modellizzazione dal modello editoriale dell'edizione critica risponde a due principali obiettivi. Il primo rientra nella prospettiva di una ricerca futura nel campo delle edizioni semantiche digitali (presentata nel capitolo conclusivo) ed è quello di proporre un'ontologia che consenta la descrizione dei dati contenuti nel testo e nell'apparato critico in modo chiaro e verificabile. Il secondo obiettivo riguarda lo scopo ultimo della strategia di modellizzazione, cioè la produzione di strumenti di visualizzazione in grado di soddisfare le esigenze scientifiche anche di quella parte della comunità scientifica che tuttora preferisce pubblicare a stampa le proprie edizioni.

² FAIR sta per findability, accessibility, interoperability e reuse, che sono i tre aspetti da potenziare nei dati pubblicati online secondo le linee guida stabilite in (Wilkinson et al. 2016).

3.2 Modalità di sviluppo dell'ontologia

La presente ontologia di esempio è stata sviluppata in diverse fasi, nell'ordine:

1. l'inquadramento dei riferimenti teorici (presentati nel dettaglio nella prossima sezione);
2. la stesura di un glossario di riferimento, raccogliendo le definizioni fornite nella letteratura;
3. l'analisi di alcune edizioni critiche a stampa e digitali per individuare i componenti comuni, in termini di contenuto e di presentazione grafica;
4. l'analisi di ontologie esistenti che descrivono il concetto di edizione critica (vd. sezione 2.5.5);
5. lo studio delle ontologie CIDOC CRM e LRM₀₀, per acquisire maggiore familiarità con le tecniche di modellizzazione di un'ontologia;
6. la stesura di un documento per avviare la progettazione dell'ontologia, definendo le classi e le proprietà necessarie;
7. la definizione dell'ontologia in OWL 2;
8. la messa a punto delle definizioni per i vari componenti dell'ontologia;
9. la messa a punto del file OWL e del documento di descrizione dell'ontologia;
10. la pubblicazione dell'ontologia sul web.

Le fasi 7 - 9 sono state reiterate più volte, fino al raggiungimento della prima versione definitiva dell'ontologia pubblicata online.

La fase di progettazione dell'ontologia (6) è stata facilitata dall'adozione di alcune pratiche raccomandate per lo sviluppo di ontologie, riassunte nella pubblicazione *Ontology development 101: A guide to creating your first ontology* (Noy e McGuinness, 2001), in particolare la definizione delle *competency questions* e l'utilizzo di un approccio *middle-out*. Le '*competency questions*', ovvero le domande cui una base di conoscenza (*knowledge base*) dovrebbe essere in grado di rispondere, guidano lo sviluppatore dell'ontologia nella definizione dello *scope* (ambito) dell'ontologia. Nel caso del presente modello degli esempi di *competency questions* utilizzate sono:

- Quali sono i componenti dell'edizione critica?

- Il testo critico è preceduto da un'introduzione?
- L'apparato critico è suddiviso in fasce?
- Con quale sigla viene identificato il testimone?

Con approccio *middle-out* si intende che l'ontologia è stata sviluppata partendo dalla definizione di classi né troppo astratte né troppo concrete, ma che figurativamente si trovano nel mezzo del dominio. Queste classi sono solitamente più facili da individuare, perché rappresentano degli oggetti molto comuni e noti del dominio. Nel caso della presente ontologia, alcune delle prime classi individuate sono state quelle che corrispondono ai concetti di testo critico, apparato critico, testimone, lezione, ecc. Una volta individuate le classi "di mezzo", si procede con la progettazione sia verso il basso per descrivere il dominio nel dettaglio, sia verso l'alto per individuare classi e proprietà più generiche, in cui rientrano gli altri concetti. Nella fase di astrazione e generalizzazione delle classi, sono stati presi molto a riferimento CIDOC CRM e LRM₀₀.

Affinché il modello risultasse il più possibile familiare per gli esperti del dominio, i nomi delle classi e delle proprietà, così come le relative definizioni, sono stati stabiliti sulla base del lessico individuato studiando la letteratura di riferimento.

Per sviluppare e scrivere l'ontologia in OWL (fasi 7 e 9) è stato utilizzato prevalentemente il software *open source* OWLGrEd (OWL Graphical Editor).³ Come suggerisce il nome, si tratta di un *editor* grafico di ontologie in OWL, all'interno del quale l'ontologia si presenta come un diagramma, mentre le classi, le proprietà e le istanze sono formattate graficamente secondo la notazione UML. OWLGrEd consente di scaricare sia il file OWL sia le immagini dei diagrammi in formato SVG. La scelta è ricaduta su questo software in quanto si tratta di uno strumento intuitivo, facilmente utilizzabile anche da chi sviluppa un'ontologia per la prima volta. Inoltre, con OWLGrEd è stato possibile realizzare non solo l'ontologia vera e propria, ma anche abbozzare i *class diagram* UML presentati nel dettaglio nel prossimo capitolo.

³ OWLGrEd è stato creato nel 2013 dall'Institute of Mathematics and Computer Science dell'Università della Lettonia (Latvijas Universitāte). Sito web del software: <http://owlgred.lumii.lv/>.

Per perfezionare la scrittura dell'ontologia in OWL (fase 9) è stato utilizzato il software Protégé,⁴ in quanto più potente per il controllo della validità dell'ontologia.

In fase di studio delle ontologie esistenti (5) è stato utilizzato il *tool online* VOWL (Visual Notation for OWL Ontologies),⁵ disponibile anche come plugin di Protégé, per visualizzare in forma di grafo le diverse ontologie e analizzare i rapporti tra le classi in modo più intuitivo.

Infine, l'ontologia è stata pubblicata in un repository GitHub pubblico ed è consultabile come sito web grazie a GitHub Pages⁶ e al *tool open source* LODÉ (Live OWL Documentation Environment),⁷ un servizio che estrae automaticamente classi, proprietà e altri elementi da un'ontologia OWL e OWL 2, e li visualizza come elenchi ordinati, insieme alle relative definizioni testuali, in una pagina HTML progettata per la navigazione e la consultazione attraverso collegamenti ipertestuali. Gli IRI associati all'ontologia e ai diversi elementi, così come il dominio dell'ontologia ("critical-edition-ontology"), sono stati creati con il servizio PURL,⁸ che permette di creare degli URL persistenti.

3.3 Teoria di riferimento

Per mettere meglio a fuoco le caratteristiche del metodo neo-lachmanniano, sono state consultate le pubblicazioni di autorevoli studiosi italiani che hanno contribuito alla sua messa a punto e alla sua applicazione nel campo della filologia romanza, in particolare: Brambilla Ageno 1984, Avalle 1978 e Contini 2014.

Una ricostruzione storica dettagliata del metodo neo-lachmanniano dagli albori fino alle sue recenti applicazioni si trova nel volume *Everything You Always Wanted to Know about Lachmann's*

⁴ Protégé è stato realizzato nel 2016 dall'Università di Stanford. È stato applicato perlopiù in campo biomedico. Sito web del software: <https://protege.stanford.edu/software.php>.

⁵ Il *tool* è stato realizzato nel 2014 contestualmente alla specifica di una notazione per la visualizzazione di ontologie in OWL, opera di Stefan Negru (Alexandru Ioan Cuza University), Steffen Lohmann e Florian Haag (University of Stuttgart). La specifica è disponibile online: <http://vowl.visualdataweb.org/v2/>. Sito web del *tool*: <http://vowl.visualdataweb.org/>.

⁶ GitHub Pages (<https://pages.github.com/>) è un servizio di GitHub che consente di pubblicare un proprio *repository* in forma di sito web gratuitamente.

⁷ <https://github.com/essepuntato/LODE>.

⁸ <https://purl.archive.org/>.

Method: A Non-Standard Handbook of Genealogical Textual Criticism in the Age of Post-Structuralism, Cladistics, and Copy-Text (Trovato 2017).

La descrizione dei concetti del modello e delle relazioni tra di essi è stata fondata sullo studio di alcuni manuali di più recente pubblicazione e largamente adottati per l'insegnamento della filologia in ambito italiano: Chiesa 2002, Stussi 2006 e Beltrami 2017. A questi si aggiunge anche il volume *Critica del testo* a cura di Leonardi (2022), pubblicato lo scorso anno. La scelta di servirsi di pubblicazioni di tipo manualistico è giustificata dalla necessità di disporre, da un lato, di definizioni chiare dei concetti alla base della disciplina e, dall'altro lato, di accurate descrizioni dei principi che guidano il lavoro editoriale del filologo. Per lo stesso motivo è stato consultato il lessico *online Parvum Lexicon Stemmatalogicum* (PLS),⁹ curato da Philipp Roelli e Caroline Macé e realizzato presso l'Università di Helsinki.

Per valutare l'applicabilità del presente modello all'interno di un quadro disciplinare più ampio e variegato, è stato consultato *The Handbook of Stemmatalogy*, curato sempre da Roelli e pubblicato nel 2020, in cui confluiscono le esperienze di studiosi di paesi e di campi della filologia differenti.

Infine, per tradurre in inglese i nomi dei componenti del modello, è stato preso a riferimento il *Lexicon of Scholarly Editing* (LexiconSE).¹⁰ Tale lessico raccoglie per ogni concetto alla base della teoria e della pratica filologica un insieme di definizioni esistenti elaborate da studiosi autorevoli. Mentre le definizioni sono in lingue diverse, le etichette scelte per ogni concetto sono in inglese, fungendo, quindi, da denominatore comune tra le definizioni che provengono da metodologie e ambiti della filologia diversi. In virtù di questa loro natura interdisciplinare e internazionale, le etichette sono state riprese anche nel presente modello.

3.3.1 Edizioni di esempio

Le edizioni di esempio analizzate sono state circoscritte perlopiù alle edizioni critiche di opere letterarie del Medioevo in volgare italiano o francese. Come esempi di edizioni critiche a stampa, sono

⁹ <https://wiki.helsinki.fi/display/stemmatalogy/Parvum+lexicon+stemmatologicum>.

¹⁰ Il *Lexicon for Scholarly Editing* è stato creato da un gruppo di studiosi della European Society for Textual Scholarship (ESTS) e del Centre for Manuscript Genetics dell'Università di Anversa. Lo scopo principale del lessico è quello di fornire a livello internazionale e interdisciplinare il dibattito sulla teoria e la pratica della critica testuale in Europa.

state individuate delle pubblicazioni di diverse tipologie di opere e che riflettessero modi differenti in cui un'edizione critica può essere confezionata. Nel dettaglio le edizioni critiche a stampa analizzate sono:

- Alighieri, Dante. 2002. *Rime*. A cura di Domenico De Robertis. Vol. Testi. 3 voll. Edizione nazionale delle Opere di Dante Alighieri a cura della Società Dantesca Italiana, 2. Firenze: Casa Editrice Le Lettere, come esempio di edizione critica di un corpus di componimenti poetici che possono essere considerati come un'opera unitaria caratterizzata da una tradizione testuale molto ampia e complessa;
- Checchi, Davide, a c. di. 2020. *Libro della natura degli animali: bestiario toscano del secolo 13*. Archivio romanzo 36. Firenze: Edizioni del Galluzzo per la Fondazione Ezio Franceschini, come esempio di edizione critica pluritestimoniale di un testo in prosa;
- Gambino, Francesca e Cerullo, Speranza, a c. di. 2009. *Salutz d'amor: edizione critica del corpus occitanico*. Testi e documenti di letteratura e di lingua 29. Roma: Salerno, come esempio di corpus di componimenti poetici di autori differenti e dalle diverse tradizioni testuali;
- Lannutti, Maria Sofia, a c. di. 2012. *Vita e passione di Santa Margherita d'Antiochia: due poemetti in lingua d'oc del 13. secolo*. Archivio romanzo 23. Firenze: Edizioni del Galluzzo per la Fondazione Ezio Franceschini, come esempio di edizione critica di un componimento poetico di maggiori dimensioni.

Per individuare edizioni critiche digitali che rientrassero nella medesima categoria è stato utilizzato il *Catalogue of Digital Editions* curato da Franzini. Purtroppo il numero di edizioni critiche digitali individuate è piuttosto esiguo e nello specifico comprende:

- Dante Alighieri - Commedia (2nd edition) (<https://www.dantecommedia.it/>);
- Dante Lab (<http://dantelab.dartmouth.edu/>);
- Dante Online (<https://www.danteonline.it/index.html>);
- Digital Dante (<https://digitaldante.columbia.edu/>);
- I testimoni della Vita Nova (<http://vitanova.unipv.it/>);
- Oregon Petrarch Open Book Project (<https://petrarch.uoregon.edu/>);
- Petrarchive (<https://dcl.ils.indiana.edu/petrarchive/>);

- The Princeton Dante Project (<https://dante.princeton.edu/>);
- Vespasiano da Bisticci, Lettere (<http://projects.dharc.unibo.it/vespasiano/>).

Infine, alcuni degli esempi sono stati ricavati direttamente dai manuali teorici, in particolare da Brambilla Ageno 1984.

3.4 Utilizzo delle ontologie esistenti

Nel precedente capitolo sono state presentate le ontologie esistenti che si prestano a essere utilizzate per elaborare la rete di modelli concettuali, in quanto inerenti al campo della filologia e della critica testuale. Per realizzare il modello concettuale di esempio, le ontologie individuate sono state utilizzate come segue:

- CIDOC CRM è stato utilizzato per generalizzare, quindi per definire in termini più astratti, i principali componenti di un'edizione critica, laddove è stato possibile stabilire dei rapporti di parentela adeguati da un punto di vista logico e semantico. È stata utilizzata la versione 7.1.2¹¹ di CRM, in quanto compatibile con LRM_{OO};
- l'ontologia CAO non è stata inclusa nell'ontologia di esempio nella sua interezza, ma molte delle sue classi e proprietà sono state utilizzate come superclassi e superproprietà, o classi e proprietà equivalenti a quelle proposte;
- LRM_{OO} è stato un riferimento teorico fondamentale per inquadrare alcuni concetti basilari del modello concettuale di esempio, tra cui quelli di edizione, testimone e opera. Dato che una versione definitiva di LRM_{OO} non è ancora stata pubblicata né tradotta in RDF, nell'ontologia di esempio è stata predisposta una mappatura con LRM_{OO}, che dovrà essere perfezionata una volta che quest'ultimo sarà stato pubblicato ufficialmente. La mappatura è stata effettuata con la versione 0.9 del modello, pubblicata a gennaio 2023;

¹¹ La sigla bibliografica usata d'ora in avanti nella tesi per fare riferimento alla specifica di CIDOC CRM secondo la versione 7.1.2 è CIDOC CRM v. 7.1.2. La versione stampabile della specifica è disponibile online all'indirizzo: https://cidoc-crm.org/sites/default/files/cidoc_crm_version_7.1.2.pdf.

- L'ontologia HiCO è stata utilizzata nell'ontologia di esempio in modo analogo all'uso che se ne fa nella CAO, ovvero per rappresentare attraverso la classe "Interpretation Act" e le relative classi il lavoro editoriale svolto dal filologo;
- PROV-O è stata presa a riferimento come strumento per descrivere da un punto di vista più astratto alcuni aspetti del lavoro editoriale;
- l'ontologia Scholarly Editing, non essendo più mantenuta, non è stata importata nell'ontologia di esempio né mappata, ma utilizzata solo come riferimento teorico per individuare delle soluzioni di modellizzazione;
- le ontologie FaBiO e DoCO sono state utilizzate come riferimenti teorici per definire che cos'è un'edizione critica e da quali elementi è costituita;
- infine, il concetto di annotazione alla base della Web Annotation Ontology è stato utilizzato per modellizzare le voci dell'apparato critico e degli altri apparati affini.

Le classi e le proprietà esistenti sono state riutilizzate il più possibile ogni volta che risultavano essere idonee, non solo dal punto di vista dell'organizzazione logico-gerarchica degli oggetti del dominio, ma anche per le loro denominazioni e definizioni. Molto spesso delle classi e delle proprietà adatte in termini logici sono state sostituite nell'ontologia con un elemento equivalente, per poter stabilire un nome e una definizione più calzanti rispetto al dominio analizzato. L'obiettivo ultimo, infatti, è quello di utilizzare l'ontologia come base per stabilire un *ubiquitous language* tra filologi e sviluppatori informatici.

All'interno dell'ontologia così come nel corso di questo capitolo, per distinguere le classi e le proprietà native da quelle derivate da ontologie esistenti, le seconde sono precedute dal *namespace* o da una sigla dell'ontologia, più precisamente:

- `crm:` per CIDOC CRM, seguito da `E` per le classi e `P` per le proprietà;
- `lrmo:` per LRM₀₀, seguito da `F` per le classi e `R` per le proprietà. Si ricorda che `lrmo` ancora non è disponibile come documento OWL, per cui la sigla proposta non corrisponde a un *namespace* effettivo;
- `cao:`, `hico:`, `prov:`, `fabio:` e `doco:` per le ontologie omonime;

- oa : per la web annotation ontology;
- se : per l'ontologia Scholarly Editing. Il *namespace* effettivo sarebbe “scholarly-editing”, ma nel contesto del presente documento si è preferito usare una forma più breve.

3.5 I concetti basilari

3.5.1 Definizione di edizione critica

Alla base del procedimento di modellizzazione vi è il concetto di edizione critica (EC): tutti gli altri concetti rientrano nel concetto di EC o sono riconducibili ad esso. Per illustrare il concetto di EC, si parte da quello più generico di edizione, presentando la definizione di Leonardi 2022: “l’edizione è l’atto con il quale si pubblica un testo del passato, cioè lo si rende di nuovo pubblico, accessibile alla lettura”. La presente ontologia è stata progettata per guidare lo sviluppo di strumenti informatici per la visualizzazione di edizioni critiche digitali, non per la loro produzione. In questo particolare contesto l’edizione è da intendersi come il risultato del lavoro filologico, come il contenuto di una pubblicazione che rende accessibile al pubblico un testo.

Con “critica” si intende che l’editore ha realizzato l’edizione applicando i principi della critica testuale, disciplina così definita da Chiesa (2002):

“La critica testuale è la disciplina che, attraverso apposite tecniche, indaga la genesi e l’evoluzione di un’**opera di carattere letterario**, individuando le sue varie forme – in particolare la forma originaria, o le forme originarie, se esse sono più di una – e studiandone le trasformazioni nel corso del tempo. Il suo obiettivo è quello di consentire la pubblicazione di un testo ‘affidabile’ di una determinata **opera**” (mia enfasi).

Nella definizione di Chiesa emerge come l’oggetto di studio della critica testuale siano opere di carattere letterario. L’oggetto di un’edizione in senso largo può essere sia un’opera di carattere letterario, sia di altro genere ma sempre di interesse culturale, storico e/o artistico, per esempio documenti cancellereschi, lettere, ecc., anche se le opere letterarie sono più spesso oggetto di EC. Ricapitolando, con “edizione critica” si intende il contenuto di una pubblicazione che rende accessibile al pubblico il testo di un’opera letteraria o di interesse storico-culturale, perlopiù del passato, realizzata per cercare di ristabilire la forma originaria del testo attraverso l’applicazione dei principi della critica testuale.

3.5.2 L'opera, la tradizione e il testimone

Tra i concetti alla base del modello, oltre al concetto di edizione critica, vi sono quelli di opera, tradizione e testimone. Il concetto di opera è molto complesso e nella letteratura scientifica sono state fornite diverse definizioni: solo all'interno del *Lexicon* al lemma "work" sono associate le definizioni di dieci diversi studiosi. Per questo motivo è parso opportuno fare ricorso alla definizione dell'IFLA di *work*, che rappresenta uno standard di riferimento nella comunità scientifica internazionale. Nell'ontologia LRM₀₀ il concetto di *work* corrisponde alla classe `lrm00:F1 Work`, descritta nel seguente modo:

"This class comprises **distinct intellectual ideas conveyed in artistic and intellectual creations**, such a[s] poems, stories or musical compositions. A Work is the outcome of an intellectual process of one or more persons. **Inherent to the notion of work is the existence of recognisable realizations of the work in the form of one or more expressions.** Works are often regarded as finished and discrete e.g. when declared as such by the creator of the work or based on the elaboration or logical coherence of its content. However, works may be recognized as existing but unfinished e.g. if the creators deliberately or accidentally never explicitly finished a particular Expression but have left behind partial expressions. In the absence of explicit information about the initial conception, which is rarely available, the first expression created constitutes witness of the beginning of existence of a Work. A Work can evolve over time, such as through revised editions. A Work may be elaborated by one or more Actors simultaneously, in parallel, or over time. **Additional expressions of a Work can continue to be created over time.** The boundaries of a Work have nothing to do with the value of the intellectual achievement but only with the dominance of a concept. **The main purpose of this class is to enable bringing together intellectually equivalent Expressions in order to display to a user all available alternatives of the same intellectual or artistic content"** (LRM₀₀ v. 0.9, p. 23, *in* enfasi).

Come si evince dal testo sopra citato, il concetto di *work*, così com'è stato definito dall'IFLA, è strettamente legato al concetto di *expression* (espressione), ovvero la realizzazione concreta, costituita da segni, dell'opera. Si riprende un esempio contenuto nella specifica di LRM₀₀, per dimostrare la differenza tra *work* ed *expression*: il romanzo *Assassinio sull'Orient Express* di Agatha Christie è un'opera, il testo originale in inglese del romanzo e la sua prima traduzione in lingua italiana sono, invece, due espressioni distinte.

Il concetto di testimone si presta a essere rappresentato come un tipo di espressione, in quanto il testimone è una copia dell'opera conservata in un documento manoscritto o a stampa che tramanda

il testo in modo diretto o indiretto,¹² in forma integrale, parziale o frammentaria.¹³ La classe `lrmoo:F2 Expression` è definita:

“This class comprises the intellectual or artistic realisations of Works in the form of identifiable immaterial objects, such as **texts**, poems, jokes, musical or choreographic notations, movement pattern, sound pattern, images, multimedia objects, or any combination of such forms. **The substance of F2 Expression is signs.** An Expression is the outcome of the intellectual or creative process of realizing a Work. **Subsequent expressions conveying the same work may be created over time. Expressions do not depend on a specific physical carrier and can exist on one or more carriers simultaneously.** As far as bibliographic practice is concerned, only instances of F2 Expression that are externalised on physical carriers other than both the creator’s brain and an auditor’s brain are taken into account. The form of F2 Expression is an inherent characteristic of the F2 Expression. Differences in form imply different Expressions (e.g., from text to spoken word, a transcript of a recording). Similarly, differences in language or means of performance imply different Expressions (e.g., translations or arrangements for different instruments). Thus, **if a text is revised or modified, the result is considered to be a new F2 Expression.** While theoretically any change in signs will result in a new Expression, conventionally the context and use will determine the rules for distinguishing among expressions” (LRMoo v. 0.9, p. 24, enfasi di chi scrive).

Per modellare il concetto di testimone bisogna stabilire in che termini si pone rispetto alla nozione di *expression*. Dalla definizione sopra citata, emerge chiaramente che il primo è incluso nella seconda, per cui il testimone può essere rappresentato come una sottoclasse di `lrmoo:F2 Expression`. Nella CAO, invece, il testimone non è rappresentato da una classe a parte ma direttamente dalla classe `Expression` di `FRBRoo`. Tecnicamente, il concetto di testimone è più ristretto, in quanto, in molti rami della filologia, tra cui la filologia romanza, si tratta di un testo composto da segni grafici. Un’espressione, invece, può essere anche la realizzazione concreta di un’opera di un’altra natura, per esempio musicale. Per questo motivo, si propone la classe `Witness`, sottoclasse di `Expression`.

Il concetto di tradizione, intesa banalmente come l’insieme dei testimoni di un’opera, è rappresentato nel presente modello dalla classe nativa `Textual Tradition` (tradizione testuale), in quanto `LRMoo` non offre una classe che rappresenti un insieme di espressioni della medesima opera.

¹² Con “indiretto” si intende che il testo dell’opera è “citato o interpolato all’interno di altri testi”, motivo per cui si parla di “tradizione indiretta” (cfr. Leonardi 2022). Si tratta, dunque, di traduzioni, citazioni e altri tipi di rifacimenti.

¹³ Alcune parti del testo conservato in un testimone possono andare perdute a causa di lacune meccaniche, dovute per esempio alla caduta di alcuni dei fogli che costituiscono il manoscritto, o per agenti esterni che danneggiano il manoscritto, per esempio bruciature o macchie di muffa.

Per definire le proprietà mediante le quali poter collegare tra di loro le tre classi finora definite, occorre partire dal linguaggio usato dai filologi per descrivere il concetto di tradizione di un'opera. Banalmente si può dire che l'opera è tramandata (*is transmitted in*) da un testimone e che il testimone è parte (*is part of*) di una tradizione.

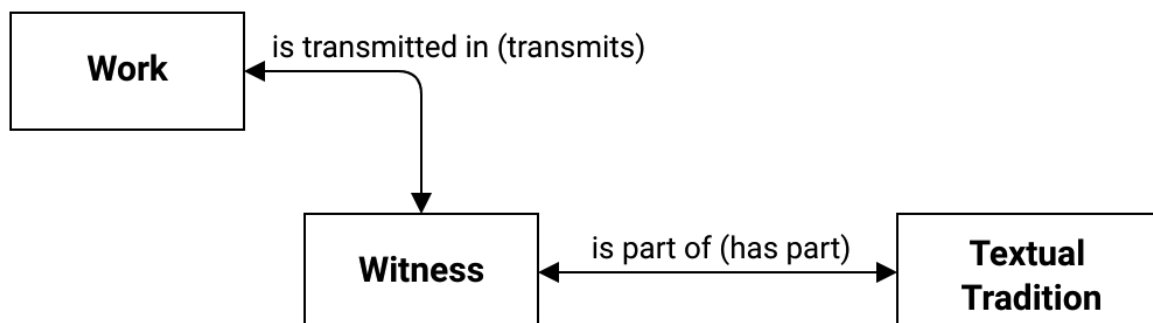


Figura 3.1 Schema delle classi *Work*, *Witness* e *Textual Tradition* dell'ontologia di esempio.

La proprietà *is transmitted in (transmits)* è una sottoproprietà di *lrmo0:R3 is realised in (realises)*, che collega le classi *lrmo0:F1 Work* e *lrmo0:F2 Expression*.

In un'edizione critica, l'editore fornisce delle informazioni dettagliate sul testimone sotto diversi aspetti, da quelli prettamente linguistici al modo in cui appare fisicamente all'interno del proprio supporto materiale. Per completezza, occorre formalizzare il concetto che descrive il supporto fisico all'interno del quale il testimone è conservato. Il supporto fisico del testimone, nel campo della filologia romanza, corrisponde tendenzialmente a un libro manoscritto o a stampa, ma può avere anche altre forme. L'analisi del supporto materiale del testimone è di estrema importanza, poiché fornisce degli elementi che interessano la produzione del testimone, dalla sua datazione alla collocazione geografica del suo luogo di produzione. Inoltre, il distinguo tra il testimone e il suo supporto materiale permette di descrivere il caso in cui in un medesimo documento sono conservati testimoni di opere distinte. È un caso molto frequente nello studio di testi medievali, in quanto a quel tempo era comune realizzare delle sillogi, quali canzonieri, laudari, ecc. nella forma di un unico manoscritto.

Tra i concetti fondamentali individuati dai FRBR figura quello di *manifestation*, direttamente collegato alle nozioni di *work* ed *expression*, che rappresenta la materializzazione (“embodiment”) dell'espressione. Riprendendo l'esempio del romanzo di Agatha Christie, la prima edizione di *Assassinio sull'Orient Express*, pubblicata nel 1934 dalla casa editrice Collins, è la manifestazione del testo

originale inglese del romanzo. La classe `lrmo0:F3 Manifestation` si presta alla rappresentazione del concetto di supporto del testimone, a prescindere che sia manoscritto o a stampa. Si riporta la definizione della classe:

“This class comprises products rendering one or more Expressions. A Manifestation is defined by both the overall content and the form of its presentation. **The substance of F3 Manifestation is not only signs, but also the manner in which they are presented to be consumed by users, including the kind of media adopted.** An F3 Manifestation is the outcome of a publication process where one or more F2 Expressions are prepared for public dissemination, but it may also be a unique form created directly on some material carrier without the intent of being formally published. An instance of F3 Manifestation typically incorporates one or more instances of F2 Expression representing a distinct logical content and all additional input by a publisher such as text layout and cover design. Additionally an F3 Manifestation can be identified by the physical features for the medium of distribution, if applicable. For example, publications in the form of hard-cover and paperback editions would be two distinct instances of F3 Manifestation, even though authorial and editorial content are otherwise identical in both publications. In the case of industrial products such as printed books or music CDs, but also digital material, an instance of F3 Manifestation can be regarded as the prototype for all copies of it. In these cases, an instance of F3 Manifestation specifies all of the features or traits that instances of F5 Item display in order to be copies of a particular publication” (LRMoo v. 0.9, p. 25, *mia enfasi*).

Per le stesse ragioni che hanno portato alla creazione della classe nativa `Witness`, anche per il supporto del testimone si propone la classe nativa `Witness Carrier` (supporto del testimone), sottoclasse di `lrmo0:F3 Manifestation`. In questo modo, si può precisare che nel presente ambito di studi il supporto fisico del testimone è tendenzialmente un libro, un codice o un frammento di codice manoscritto o a stampa. Per collegare le classi `Witness` e `Witness Carrier` si propone la proprietà `preserves (is preserved in)` sottoproprietà di `lrmo0:R4 embodies (is embodied)` che nel modello LRM unisce `lrmo0:F2 Expression` a `lrmo0:F3 Manifestation`. La proprietà `preserves (is preserved in)` prevede che ogni testimone sia collegato a un solo supporto fisico, mentre nel medesimo supporto fisico possono essere conservati i testimoni di opere diverse, come nei casi sopra citati delle sillogi.¹⁴

Prendendo come esempio di opera la *Commedia* di Dante, tra i suoi testimoni c'è il testo conservato all'interno del codice Trivulziano (Milano, Biblioteca dell'Archivio Storico Civico e Trivulziana, codex 1080), mentre il codice stesso è il supporto di quel determinato testimone.

¹⁴ Questa proprietà si presta a essere utilizzata anche nei casi in cui all'interno di un medesimo supporto fisico è possibile individuare due (o più) redazioni diverse dello stesso testo, come per esempio negli appunti di un autore. In questo caso si tratta di diverse espressioni della medesima opera.

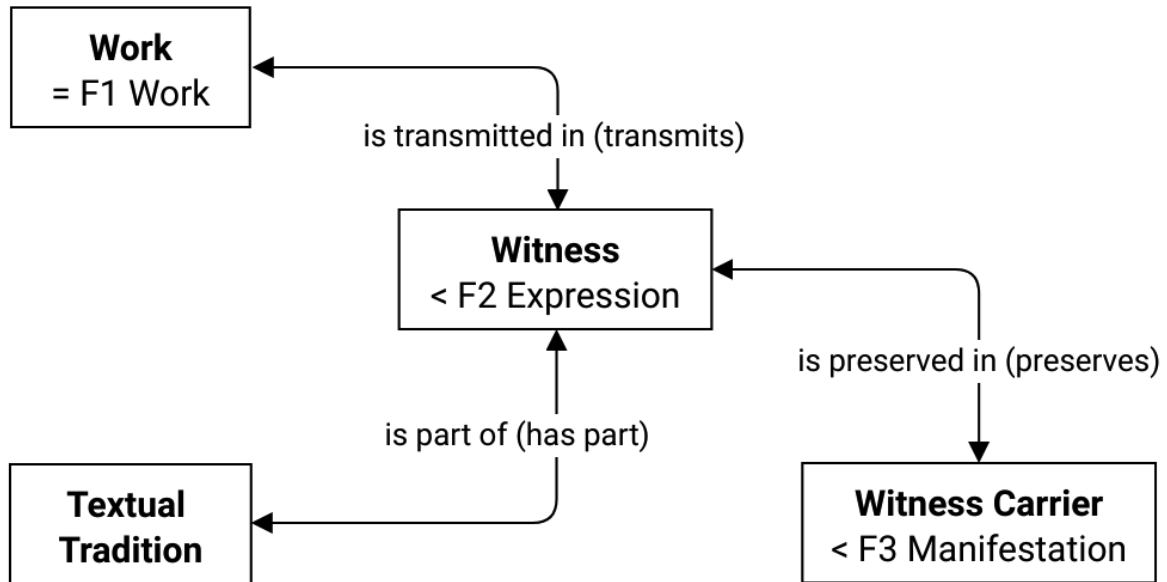


Figura 3.2 Schema delle classi che rappresentano la tradizione testuale di un'opera.

Ricapitolando, l'opera è una creazione artistica e intellettuale che funge da denominatore logico comune tra le sue diverse espressioni, che in campo filologico vengono chiamate 'testimoni'.¹⁵ Il testimone è una delle realizzazioni concrete di un'opera, conservata all'interno di un supporto materiale e costituita da una sequenza di caratteri grafici. I testimoni e i relativi supporti fisici sono il punto di partenza per lo studio della tradizione dell'opera e per la realizzazione di un'edizione critica.

3.5.3 Formalizzazione del concetto di edizione critica

Il concetto di edizione critica è stato formalizzato in due delle ontologie sopra elencate, FaBiO e SE, in entrambi i casi con delle classi dedicate.

La classe `fabio:Critical Edition` è definita come:

“A new edition of a historical publication, edited by a scholar other than the original author, containing within the body text the supposedly best version of the original work, with footnotes detailing and commenting on textual variations between different versions, typically with an introduction to the original work written by the scholar, and with a bibliography listing related publications.”¹⁶

Nell'ontologia SE la classe `se:Critical Edition` è definita:

¹⁵ Cfr. con le definizioni fornite in Bordalejo 2013: “the work is a conception in the mind of an author at a particular point in time that serves as a minimal denominator to identify its remaining physical manifestations” e Smiraglia 2001: “A work is a signifying, concrete set of ideational conceptions that finds realization through semantic or symbolic expression”.

¹⁶ Cfr. <https://sparontologies.github.io/fabio/current/fabio.html#d4e2738>.

“Edition to which a critical editor added evidence and interpretation of a text creation and historical transmission, using a critical apparatus, allowing readers to assess the editor’s choice.”¹⁷

Ad avviso di chi scrive, la definizione fornita nella FaBiO non si allinea con quella presentata alla sezione 3.5.1, in quanto è espressa in modo tautologico (“a new edition”) e l’oggetto dell’edizione critica viene definito “historical publication” invece di un’opera. In entrambe le definizioni di edizione critica, però, vengono rimarcati due aspetti: il primo è che l’edizione critica è legata a una particolare figura, definita “studioso” (“scholar”) in FaBiO ed “editore critico” (“critical editor”) in SE; il secondo è che l’edizione critica si distingue per la presenza di altri contenuti distinti dal testo dell’opera e ad esso subordinati. Il concetto di editore è modellato nella presente ontologia come illustrato nella sezione 3.5.4. Il rapporto tra l’edizione critica, i suoi contenuti e l’opera è un aspetto fondamentale per mettere a fuoco che cos’è un’edizione critica e come può essere descritta in modo astratto coerentemente con le ontologie prese a riferimento, in particolare LRM₀₀ e CRM.

La classe `fabio:Critical Edition` è modellata come una sottoclasse di `fabio:Work`, che però non equivale alla nozione di *work* definita dai FRBR, ma è una sua sottoclasse descritta come segue:

“A subclass of FRBR work, restricted to works that are published or potentially publishable, and that contain or are referred to by bibliographic references, or entities used to define bibliographic references. FaBiO works, and their expressions and manifestations, are primarily textual publications such as books, magazines, newspapers and journals, and items of their content.”¹⁸

Nella FaBiO l’edizione critica è descritta come un tipo di opera destinata alla pubblicazione, distinta però dall’opera di cui tratta. Questa definizione potrebbe risultare poco chiara e fraintendibile nel dominio della presente ontologia, in cui deve essere inclusa anche l’opera oggetto dell’edizione, rappresentata da `lrmo0:F1 Work`.

Nell’ontologia SE, `se:Critical Edition` è sottoclasse della più generica `se:Edition`, definita: “Derived **text expression** as scholarly edited text” (mia enfasi). L’edizione critica, effettivamente, può essere descritta alla stessa stregua del testimone come una delle espressioni dell’opera, in base a quanto si legge nella definizione di `lrmo0:F2 Expression`: “Subsequent expressions

¹⁷ <https://github.com/nie-ine/e-editiones/blob/master/ontology/scholarly-editing.rdf>.

¹⁸ <https://sparontologies.github.io/fabio/current/fabio.html#d4e6036>.

conveying the same work may be created over time. Expressions do not depend on a specific physical carrier and can exist on one or more carriers simultaneously” (LRMoo v. 0.9, p. 24). Per descrivere il rapporto che lega `lrmo0:F1 Work` a `lrmo0:F2 Expression`, la documentazione di LRMoo offre un esempio di edizione critica descrivibile come *expression* decisamente pertinente al nostro dominio:

“Dante’s work entitled ‘Inferno’ (F1) is realised in the Italian text of Dante’s ‘Inferno’ as found in the authoritative critical edition ‘La Commedia secondo l’antica issolu’ a cura di Giorgio Petrocchi, Milano: Mondadori, 1966-67 (= ‘Le Opere di Dante Alighieri’, Edizione Nazionale a cura della Società Dantesca Italiana, VII, 1-4) (F2).” (LRMoo v. 0.9, p. 37)

La differenza tra un testimone e l’edizione critica è che la seconda è stata fondata su un attento esame dell’intera tradizione dell’opera e persegue un chiaro obiettivo scientifico: restituire il testo dell’opera in una versione il più possibile vicina a quella originaria, rendendo conto di quello che è lo stato della tradizione. Come già evidenziato, l’edizione critica non offre solo il testo edito dell’opera, ma anche dei materiali di corredo che consentono di conoscere la tradizione e di verificare la validità del lavoro filologico svolto.

Occorre domandarsi se l’edizione è da intendersi come il solo testo dell’opera curato dall’editore (come nell’ontologia SE) oppure come l’insieme dei materiali approntati dall’editore, tra cui anche il testo edito dell’opera (come nella FaBiO). Il termine edizione nel linguaggio filologico viene usato per indicare entrambi i concetti. Ad avviso di chi scrive, il testo dell’opera curato dall’editore, che nel caso di un’edizione critica viene chiamato testo critico, non può essere separato dai materiali di corredo, *in primis* dall’apparato critico, per poter essere fruito correttamente dal lettore. Inoltre, la definizione deve essere funzionale all’obiettivo ultimo del modello concettuale: guidare lo sviluppo di uno strumento informatico per la visualizzazione di un’edizione critica digitale. Per cui l’edizione critica deve essere intesa come l’insieme dei materiali che l’editore intende pubblicare.

Stabilendo che l’edizione critica non è il solo testo critico, ma l’insieme del testo critico e di tutti i materiali di corredo, la soluzione più adatta agli scopi della presente ricerca consiste nel rappresentare il concetto di edizione critica mediante una classe nativa, denominata `Critical Edition`, che sia sottoclasse di `lrmo0:F2 Expression`. Modellando l’edizione critica come un tipo di *expression*, si esplicita che si tratta di un oggetto destinato alla pubblicazione e che la sua pubblicazione

può essere catalogata. Questa caratteristica è fondamentale sia per rappresentare un'edizione critica a stampa che per una pubblicata sul web. Inoltre, la definizione di `lrmo0:F2 Expression` si presta a rappresentare un'edizione scientifica digitale, che spesso corrisponde a un oggetto multimediale o un insieme di oggetti di diversa natura (per esempio basi di dati):

“This class comprises the intellectual or artistic realisations of Works in the form of identifiable immaterial objects, such as texts, poems, jokes, musical or choreographic notations, movement pattern, sound pattern, images, multimedia objects, or any combination of such forms.”

In un'ottica di possibile riuso del modello concettuale di esempio, occorre domandarsi come i concetti definiti possano essere espansi o generalizzati, determinando i punti di riferimento per avviare lo sviluppo di altri modelli concettuali. L'utilizzo di classi molto generiche come `lrmo0:F1 Work`, `lrmo0:F2 Expression` e `lrmo0:F3 Manifestation` per inquadrare i concetti di opera e testimone consente al modello di essere espanso e adattato virtualmente a ogni tipo di edizione scientifica. Altri concetti che inquadrano l'edizione critica, per esempio quello di editore menzionato nella precedente sezione, possono essere facilmente riutilizzati per diverse tipologie di edizioni. Occorre stabilire una strategia per generalizzare anche il concetto di edizione critica.

3.5.3.1 Generalizzazione del concetto di edizione critica

Come illustrato in precedenza, nell'ontologia SE la classe `se:Critical Edition` è sottoclasse di `se:Edition`. Altre classi dell'ontologia SE sono derivate da `se:Edition`, ciascuna rappresentante un particolare tipo di “Derived text expression as scholarly edited text”, come schematizzato in figura.

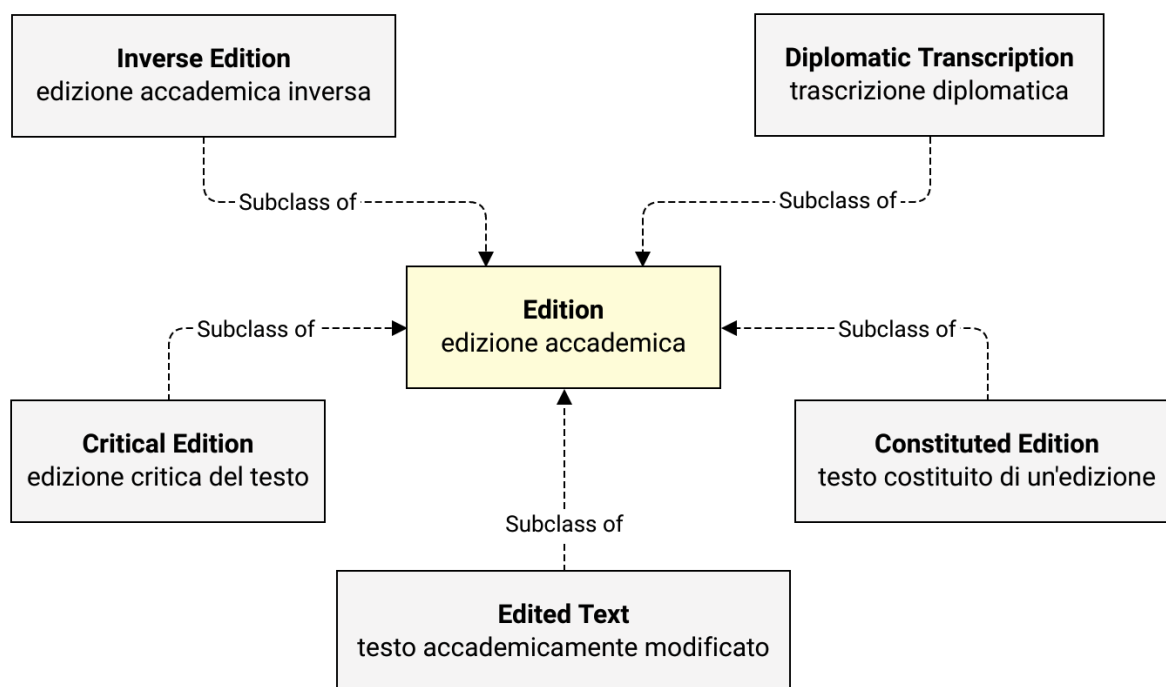


Figura 3.3 Schema che rappresenta la classe *se:Edition* e le sue sottoclassi dirette.

Non sono molto chiare le nozioni espresse dalle classi proposte nell'ontologia SE. Ad ogni modo, il meccanismo può essere riprodotto anche nella presente ontologia per generalizzare il concetto di edizione critica, fermo restando che nel presente caso con edizione critica si fa riferimento a tutti i materiali preparati dall'editore invece del solo testo critico.

La classe *Critical Edition* è, dunque, sottoclasse di un'altra classe nativa denominata *Scholarly Edition*, che a sua volta è sottoclasse di *lrmo:Expression*. La scelta è ricaduta sull'etichetta *scholarly edition*, invece di un generico *edition*, per circoscrivere il concetto di edizione a quelle prodotte in contesto accademico, applicando le metodologie della filologia. Nel *Lexicon* l'aggettivo *scholarly* (che in italiano è stato tradotto con scientifica, cfr. Mancinelli e Pierazzo 2020) è associata alla definizione di *scholarly editing*:

“Scholarly editing [...] is used to describe any sort of edition prepared by a person claiming to be a scholar. [...] The term should be reserved for editorial efforts designed to make available for scholarly use works not ordinarily available or available only in corrupt or inadequate forms” (Shillingsburg 1996).¹⁹

¹⁹ Cfr.: <https://lexiconse.uantwerpen.be/lexicon/editingScholarly.html>.

Con *scholarly* si individua un'edizione prodotta da uno studioso in ambiente accademico, ma non si precisano le metodologie utilizzate né la natura dell'oggetto di studio o gli obiettivi scientifici, se non quello molto generico di mettere a disposizione della comunità scientifica opere altrimenti inaccessibili. All'interno del concetto di edizione scientifica possono rientrare quelli di edizione critica, edizione diplomatica o interpretativa, edizione genetica, *digital documentary edition*, ecc. Ciascun modello editoriale e il rispettivo approccio filologico può corrispondere a una sottoclasse di *Scholarly Edition* e divenire il punto di partenza per la modellizzazione di altre ontologie.

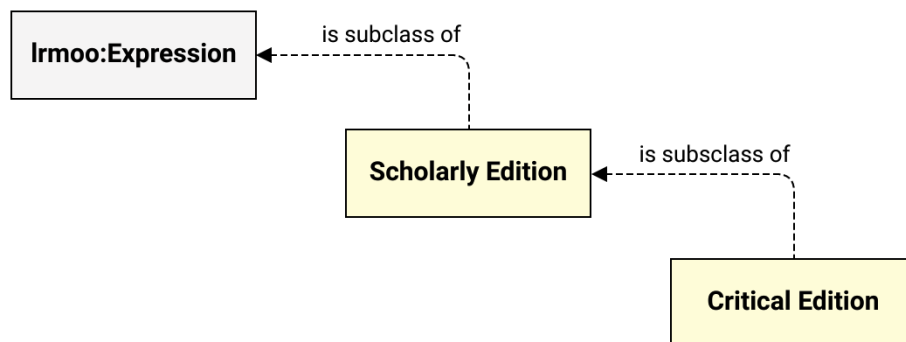


Figura 3.4 Schema della classe *Critical Edition* e delle sue superclassi.

Sia l'edizione critica che l'edizione scientifica in generale sono sottoclassi di *lrmoo:F2 Expression*, per cui entrambe possono essere collegate alla classe *Work* mediante la proprietà *lrmoo:R3 is realises (is realised in)*. Dato che questa proprietà unisce un'opera a un qualsiasi tipo di *Expression*, per cui anche a un testimone, si propone una sottoproprietà di *lrmoo:R3* nativa, denominata *is edited in (is edition of)*, per collegare l'opera all'edizione scientifica.

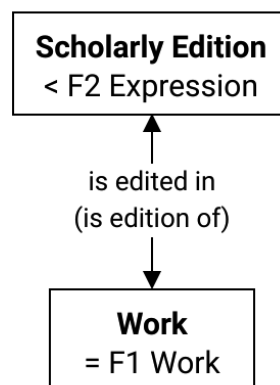


Figura 3.5 Schema del rapporto tra le classi *Scholarly Edition* e *Work*.

3.5.4 L'allestimento dell'edizione critica e l'editore

Il lavoro filologico ed editoriale per l'allestimento di un'edizione critica può essere svolto da una persona sola, che possiamo genericamente definire l'editore, oppure da più editori in *équipe*. L'editore deve possedere buone competenze filologiche e realizzare l'edizione applicando metodologie condivise, che nel caso del presente modello sono riconducibili ad un approccio di tipo neo-lachmanniano, così da garantire la validità scientifica e la verificabilità del proprio lavoro. L'editore deve essere noto, in quanto responsabile dell'edizione.

L'ontologia SE propone tre classi diverse per descrivere il responsabile di un'edizione critica:

- `se:AgentEditor`, “Role of a human agent - person or organization - editing a text expression scientifically”;
- `se:PersonEditor`, “Role of a person editing a text expression scientifically.”;
- `se:OrganizationEditor`, “Role of an organization editing a text expression scientifically.”.

Nelle ontologie CAO e HiCO viene utilizzata la classe `prov:Agent`²⁰ della PROV-O, per indicare rispettivamente l'editore che prepara un apparato critico o lo studioso che compie un atto interpretativo.

Tutte queste soluzioni si adattano anche al nostro modello concettuale.²¹ Tuttavia, si preferisce scegliere una soluzione conforme al CRM, nell'eventualità che il modello possa essere proposto come un'estensione di CRM. Per rappresentare il concetto di editore nel modello si propone, dunque, una classe nativa, denominata `Editor`, sottoclasse di `crm:E39 Actor`.²² Nella definizione di CIDOC CRM, infatti, la classe `Actor` comprende persone che, individualmente o in gruppi, “have

²⁰ Il concetto espresso da `prov:Agent` è così descritto “An agent is something that bears some form of responsibility for an activity taking place, for the existence of an entity, or for another agent's activity.” (cfr. <https://www.w3.org/TR/prov-o/#Agent>).

²¹ Una delle SPAR ontologies, la Publishing Roles Ontology (PRO, <http://www.sparontologies.net/ontologies/pro>), è stata ideata per rappresentare in modo dettagliato le responsabilità da parte di diversi agenti (persone o organizzazioni), specificando il ruolo ricoperto e il periodo di tempo in cui hanno ricoperto il ruolo. Questa ontologia offre un livello di dettaglio eccessivo e superfluo rispetto al dominio del presente modello concettuale. Tuttavia, potrebbe rivelarsi utile per descrivere edizioni condotte in modo fortemente collaborativo, da un grande numero di collaboratori, aventi ruoli e responsabilità differenti, per descrivere, dunque, dei casi di *social edition*.

²² http://cidoc-crm.org/cidoc-crm/7.1.2/E39_Actor.

the potential to perform intentional actions of kinds for which someone may be held responsible” (CIDOC CRM v. 7.1.2, p. 83). Nel caso in cui ci siano più responsabili scientifici, l'intero gruppo di editori può essere modellato con la classe nativa `Editors Group`, che è una sottoclasse di `crm:E74 Group`.²³ `Editors Group` è sottoclasse di `Editor`, così come `crm:E74 Group` è sottoclasse di `crm:E39 Actor`. Per specificare che un editore è parte di un gruppo di editori è possibile utilizzare la proprietà `crm:P107 has current or former member (is current or former member of)`.

Il concetto di editore è collegabile sia all'edizione critica, che in generale all'edizione scientifica. Per questo motivo si propone un collegamento tra `Editor` e `Scholarly Edition`, che possa essere usato anche per `Critical Edition`. Per determinare il collegamento sono state vagliate due soluzioni diverse. La prima è pienamente compatibile con i modelli LRM₀₀ e CIDOC-CRM, ma prevede il ricorso a una serie di classi in più, che rischiano di appesantire inutilmente il modello. Come si vede nella figura sotto, la classe `lrmo0:F2 Expression` è collegata alla classe `crm:E39 Actor` non direttamente ma attraverso le classi `lrmo0:F28 Expression Creation`, `crm:E65 Creation` e `crm:E7 Activity`. La prima soluzione consiste nel proporre delle classi native equivalenti a quelle appena menzionate, al solo scopo di collegare i concetti di edizione critica e di editore in modo coerente rispetto alla soluzione di LRM₀₀.

²³ La classe `crm:E74 Group` (http://cidoc-crm.org/cidoc-crm/7.1.2/E74_Group) è una sottoclasse di `crm:E39 Actor` e comprende “gatherings or organizations of human individuals or groups that act collectively or in a similar way due to any form of unifying relationship” (CIDOC CRM v. 7.1.2, p. 102).

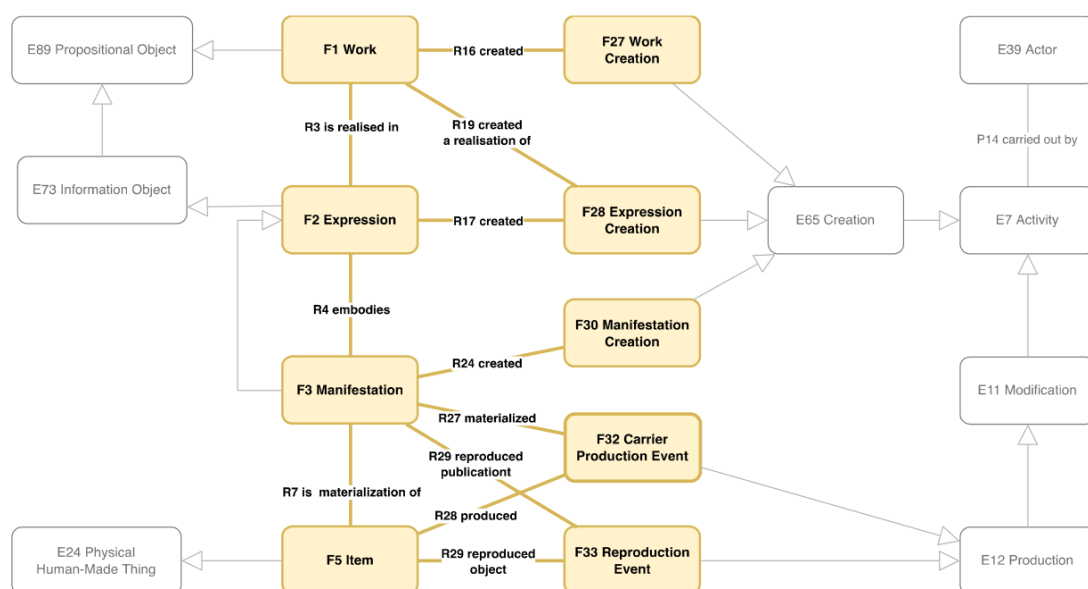


Figura 3.6 Le principali classi e proprietà di LRMOO e i loro collegamenti alle classi di CIDOC CRM.

La seconda soluzione, invece, prevede la definizione di una proprietà nativa, denominata *created* (was created by), che funga da *shortcut* (scorciatoia),²⁴ collegando direttamente *Scholarly Edition* a *Editor* (o *Editors Group*, dato che è sottoclasse di *Editor*). Questa soluzione è preferibile, perché è sufficiente rispetto agli scopi del modello ed evita l’inserimento di classi funzionali esclusivamente alla definizione della relazione tra i concetti di edizione critica e di editore.

²⁴ “A shortcut is a formally defined single property that represents a deduction or join of a data path in the CIDOC CRM. The scope notes of all properties characterized as shortcuts describe in words the equivalent deduction” (CIDOC CRM v. 7.1.2, p. 16).

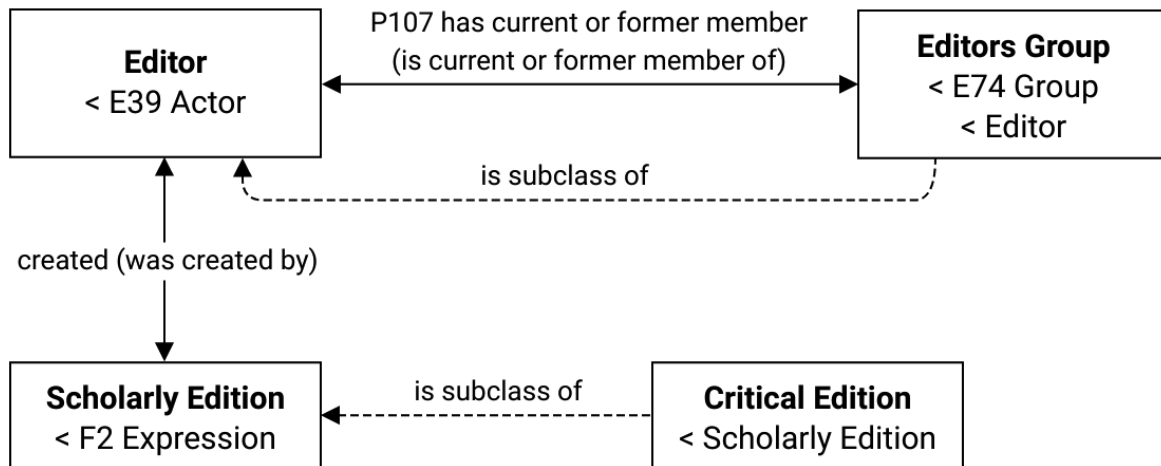


Figura 3.7 Schema che rappresenta il rapporto tra le classi Editor, Editors Group e Scholarly Edition.

3.5.5 La pubblicazione dell'edizione critica e il lettore

Nella sezione 3.5.1 il concetto di edizione critica è stato definito come il prodotto del lavoro filologico svolto dall'editore. Questo prodotto a sua volta è stato formalizzato come un tipo particolare di espressione di un'opera, dato che oltre al testo dell'opera vero e proprio include una serie di materiali di corredo prodotti dall'editore a complemento del testo critico. La scelta di formalizzare l'edizione critica non come una pubblicazione ma come il contenuto di una pubblicazione è dipesa dalla necessità di separare il concetto di edizione critica dalle sue possibili implementazioni concrete. In questo modo il modello descrive i contenuti di un'edizione critica su un piano più astratto e generico, che sono comuni sia alle edizioni a stampa sia a quelle digitali.

Tuttavia, l'edizione critica è intrinsecamente destinata alla pubblicazione, nel senso che viene creata per rendere il testo di un'opera accessibile al pubblico. Per completezza occorre specificare, dunque, un'ulteriore classe che rappresenti la pubblicazione dell'edizione critica e dell'edizione scientifica in generale. Tale classe può coincidere con `lrmo0:F3 Manifestation`, sulla base di quanto scritto nella sua definizione:

“The substance of F3 Manifestation is not only signs, but also the manner in which they are presented to be consumed by users, including the kind of media adopted. An F3 Manifestation is the outcome of a publication process where one or more F2 Expressions are prepared for public dissemination” (LRM00 v. 0.9, p. 25).

Si sottolinea, inoltre, come nel modello LRM00 la classe `lrmo0:F3 Manifestation` sia una sottoclasse di `lrmo0:F2 Expression`. Questa parentela evidenzia come, da un punto di vista

logico, la pubblicazione di un testo comprenda il testo stesso e la sua presentazione nel supporto materiale.

Nel presente modello si propone una classe nativa denominata `Scholarly Edition Publication`, sottoclasse di `lrmo:F3 Manifestation`. Per collegare l'edizione scientifica (e di conseguenza anche l'edizione critica) alla sua pubblicazione si propone una proprietà derivata da `lrmo:R4 embodies (is embodied in)` e denominata `is published in (publishes)`.

La presenza di due classi distinte, `Scholarly Edition` e `Scholarly Edition Publication`, permette di rappresentare i casi in cui la medesima edizione critica viene pubblicata in formati e modalità diverse. Come già illustrato nel capitolo 2, da un medesimo file sorgente, tendenzialmente in formato XML/TEI, è possibile derivare molteplici output. Un esempio è l'edizione critica del *De rebus siculis carmen* di Petrus de Ebulo, curata da Fulvio Delle Donne e pubblicata sia come edizione digitale grazie al software EVT sia in una versione stampabile come documento PDF.²⁵

Al contempo, la distinzione tra l'edizione scientifica e la sua pubblicazione, permette di descrivere i casi in cui all'interno di un unico volume vengono pubblicate le edizioni di più opere, come nel caso del volume *Salutz d'Amor* (Gambino e Cerullo 2009), che raccoglie le edizioni critiche di un corpus di testi occitanici. La proprietà `is published in (publishes)` proprio come `lrmo:R4 is embodied in (embodies)` permette di collegare a un'istanza di `Scholarly Edition` più istanze di `Scholarly Edition Publication` e a un'istanza della prima classe più istanze della seconda, così da poter rappresentare entrambe le situazioni offerte come esempio.

Il destinatario della pubblicazione dell'edizione critica è il lettore, ovvero l'altro attore che agisce nel dominio descritto dal modello oltre all'editore. A differenza dell'editore, il lettore è una persona che può rimanere anonima, perciò è rappresentato nel modello mediante una classe nativa denominata `Reader`, sottoclasse di `crm:E21 Person`.²⁶

²⁵ Versione digitale dell'edizione critica del *De rebus siculis carmen*: <http://web.unibas.it/bup/evt2/pde/index.html>. Versione stampabile in PDF: <http://web.unibas.it/bup/Libri/Petrus%20de%20Ebulo.pdf>.

²⁶ "This class comprises real persons who live or are assumed to have lived" (CIDOC CRM v. 7.1.2, p. 74). http://cidoc-crm.org/cidoc-crm/7.1.2/E21_Person.

Per descrivere la relazione tra lettore e pubblicazione dell'edizione critica si propone una proprietà nativa: *reads (is read by)*. La scelta del termine leggere, potrebbe sembrare semplicistica, alla stessa stregua del termine lettore. La lettura di un'edizione critica può essere condotta a diversi livelli, a seconda degli obiettivi del lettore: può essere una mera lettura dell'opera; lo studio della tradizione e una verifica del lavoro filologico condotto dall'editore; la consultazione di alcune parti specifiche dell'edizione. Rasmussen (2017) individua tre diversi ruoli che il lettore di un'edizione scientifica digitale può assumere: lettore, utente e collaboratore ("co-worker"). Il lettore si serve delle edizioni critiche per poter leggere delle versioni affidabili di opere letterarie. L'utente consulta le edizioni critiche in modo più approfondito, per indagare il rapporto tra l'opera e i suoi molteplici testimoni oppure tra il testo critico dell'edizione e i diversi paratesti. La distinzione tra lettore e utente proposta da Rasmussen è, a mio avviso, piuttosto sfumata. La definizione di collaboratore, invece, rappresenta un caso particolare di edizione critica digitale, che fuoriesce dal dominio del presente modello. Potrebbe rientrare, semmai, in una futura espansione del modello, che descriva le edizioni collaborative (vd. Siemens et al. 2012). E anche in tale caso la collaborazione potrebbe assumere forme molto diverse, dalla semplice segnalazione di un refuso alla proposta di integrare il testo del commento con ulteriori informazioni fino all'assunzione di responsabilità editoriali più ampie.

Al termine leggere si potrebbero preferire, dunque, altri termini come studiare, consultare o anche usare, sia in virtù del possibile (ri-)uso dei dati contenuti in un'edizione sia perché, nel contesto di un'edizione pubblicata in forma digitale, il lettore deve essere concepito anche come l'utente di un'interfaccia.²⁷ Per gli scopi del presente modello tutte queste sfumature di significato vengono raccolte in un generico legge, che mira a sintetizzare i diversi modi in cui un'edizione scientifica può essere fruita.

²⁷ Il concetto di utente è qui da intendersi nella prospettiva della progettazione software e non nella definizione proposta in Rasmussen 2017 (vd. nota precedente).

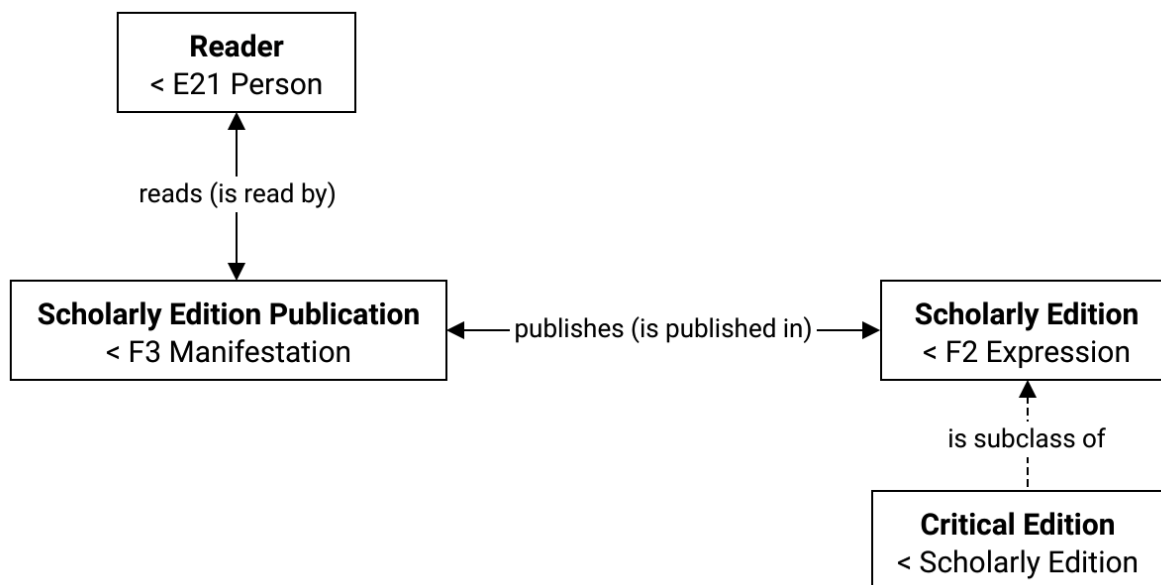


Figura 3.8 Schema del rapporto tra la classe *Scholarly Edition* e *Reader*.

Potrebbe sembrare che in questo modello manchi un terzo attore, cioè l'autore dell'opera. Il concetto di autore è stato volutamente omesso dalla descrizione del presente dominio, per circoscrivere il modello di esempio. È sembrato opportuno focalizzarsi sulla descrizione di edizioni critiche realizzate allo scopo di ricostruire la forma originaria di opere per le quali non sono disponibili testi autografi, idiografi o attribuibili all'autore. In questo particolare contesto applicativo, il concetto di autore è parso superfluo. Nel contesto di edizioni che possano usufruire di testimoni redatti direttamente o indirettamente dall'autore, realizzate con gli accorgimenti e le metodologie che pertengono alla filologia d'autore, il concetto di autore è ovviamente indispensabile. Ad ogni modo, il presente modello si presta a essere esteso mediante la definizione di concetti necessari per la corretta descrizione di edizioni realizzate con approcci filologici diversi da quello scelto come riferimento.

3.5.6 Esempi di applicazione dei concetti basilari

Per mostrare come i concetti base del modello possono essere usati per descrivere un'edizione critica prendiamo due esempi dal mondo reale: un'edizione a stampa e un'edizione digitale.

Come edizione a stampa si propone quella del *Libro della natura degli animali* (LdN), un bestiaro toscano del XIII secolo, curata da Davide Checchi e pubblicata nel 2020. Le informazioni che riguardano la pubblicazione e l'editore si evincono dal riferimento bibliografico dell'edizione critica oppure dal frontespizio del volume. I lettori possono essere molteplici, a titolo esemplificativo nello

schema si presenta il nome della sottoscritta. Mentre le informazioni sulla tradizione dell'opera si evincono dall'introduzione dell'edizione critica. I testimoni che tramandano il testo del LdN in forma diretta sono quindici. Nello schema se ne presentano solo tre in forma completa allo scopo di mostrare come le informazioni vengono formalizzate.

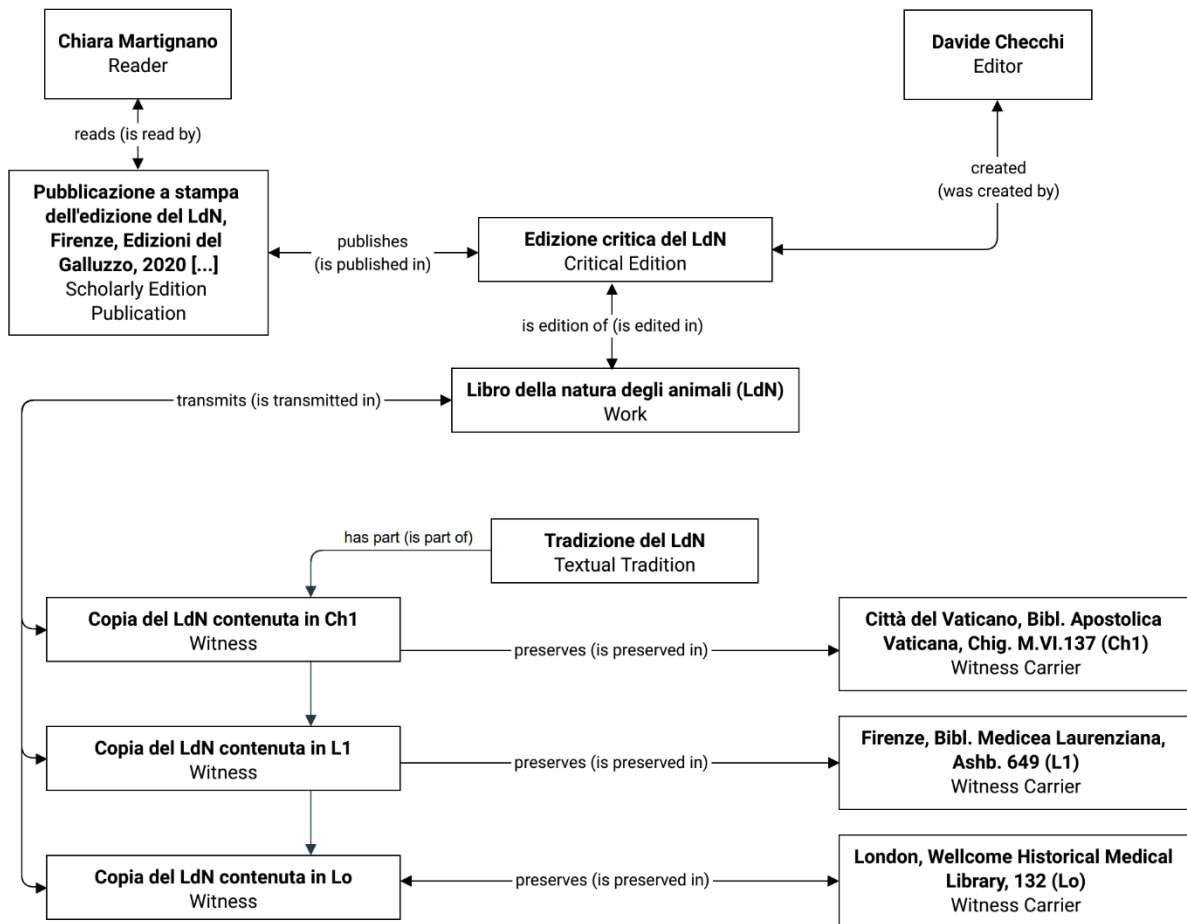


Figura 3.9 Formalizzazione di un esempio di edizione a stampa attraverso i concetti basilari.

Come edizione digitale si propone quella dei *Rerum Vulgarium Fragmenta* di Francesco Petrarca, meglio nota con il nome di 'Petrarchive' e curata da un gruppo di studiosi dell'università dell'Indiana.²⁸ Rispetto all'edizione precedente, questa presenta due particolarità: in primo luogo è basata su un solo testimone, il celebre codice Vat. lat. 3195, in parte autografo e in parte copiato da un copista al servizio di Petrarca,²⁹ e sotto il diretto controllo di Petrarca,³⁰ per cui si tratta di un testo

²⁸ <https://dcl.ils.indiana.edu/petrarchive/index.php>.

²⁹ Che probabilmente non è Malpaghini (Berté 2015).

³⁰ Nei casi in cui la copia di un'opera è stata realizzata da una persona diversa dal suo autore, ma comunque sotto la sua stretta sorveglianza, si dice che è una copia idiografa.

che può essere considerato originale (una volta epurato da eventuali errori commessi involontariamente); in secondo luogo il *Canzoniere* di Petrarca è considerato come un'opera unitaria, nonostante sia composto da più componimenti. Le informazioni sul lavoro editoriale svolto, sull'opera e sul suo testimone sono disponibili nelle pagine che presentano l'edizione critica. Per dimostrare il rapporto tra un gruppo di responsabili scientifici e un singolo editore, si riporta come esempio il nome del principale editore dell'edizione. Trattandosi di un solo testimone, la classe Tradizione può essere omessa, in quanto risulta superflua.

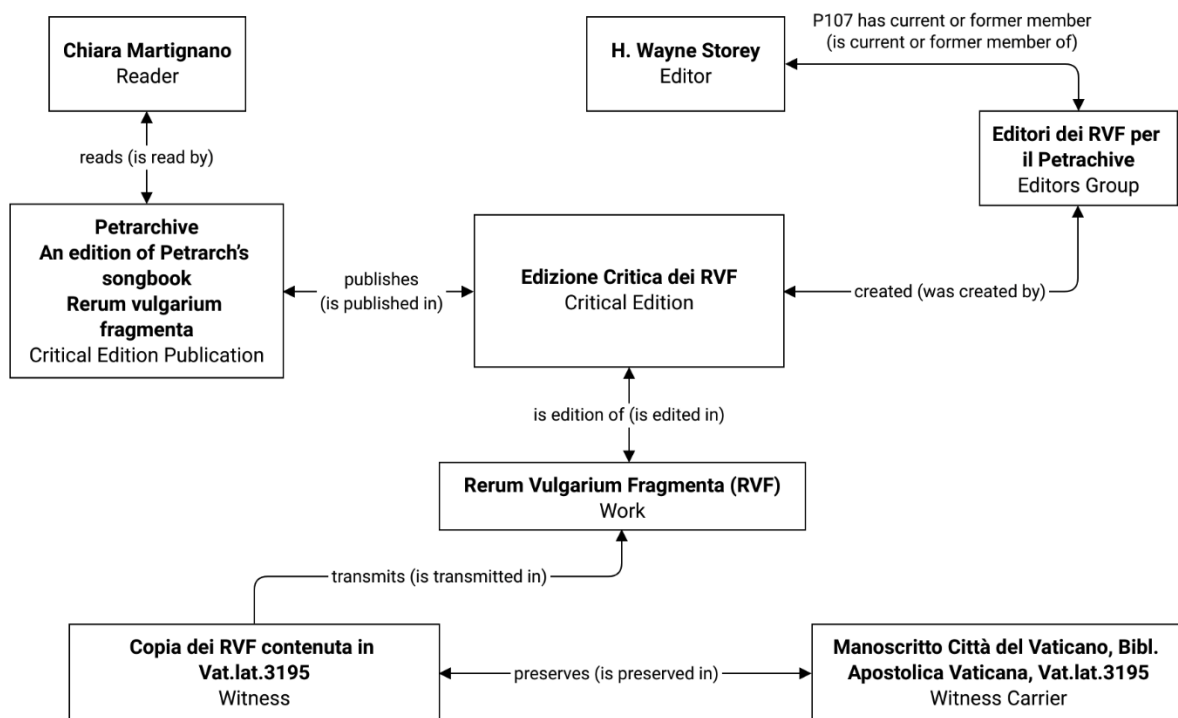


Figura 3.10 Formalizzazione di un esempio di edizione digitale attraverso i concetti basilari.

3.6 I componenti dell'edizione critica

Il testo critico rappresenta il principale prodotto del lavoro svolto dall'editore e costituisce il fulcro di un'edizione critica. All'interno dell'edizione il testo critico è accompagnato da una serie di materiali di corredo, che guidano il lettore nella comprensione del testo e permettono di verificare il lavoro svolto dall'editore, rendendolo trasparente:

“Il primo presupposto di un'edizione critica è [...] la trasparenza: chi è responsabile di un'edizione deve esporre i criteri seguiti per stabilire il testo, in modo che chi legge abbia ben chiaro che rapporto abbia il testo edito con la tradizione manoscritta e con la storia redazionale dell'opera, e deve fornire ogni elemento utile a consentire al lettore di ripercorrere le tappe del suo lavoro [...]. Il secondo pre-

supposto generale è la comprensione del testo: l'editore deve assicurare l'interpretazione del significato almeno letterale dell'opera che pubblica e deve indicare i luoghi in cui vi siano incertezze o problemi testuali che non è riuscito a risolvere. [...] **Un'edizione critica dunque, oltre al testo, fornisce in genere uno o più strumenti complementari:** un'introduzione che affronta i vari aspetti (descrizione e classificazione dei manoscritti, analisi linguistica, criteri di trascrizione e di edizione), un apparato delle correzioni e delle varianti, una parafrasi o traduzione, parziale o totale, un glossario, un commento filologico e/o letterario” (Leonardi 2022, pp. 132-133).

Nella presente sezione viene illustrato il modo in cui i principali componenti che costituiscono un'edizione critica, ovvero il testo critico, l'apparato critico, il commento filologico e l'introduzione, sono stati formalizzati nel modello concettuale. Per gli altri componenti menzionati nel testo sopra, quali la parafrasi, la traduzione e il glossario, si forniscono delle indicazioni su come possono essere rappresentati espandendo il presente modello concettuale.

3.6.1 Definizione di “componente”

Le modalità di organizzazione logica e formattazione grafica dei contenuti di un'edizione critica sono nate e si sono consolidate nel contesto delle edizioni critiche a stampa. Nelle edizioni digitali, tali modalità, pur venendo affiancate da nuove soluzioni apposite per l'ambiente digitale, sono state riproposte.³¹ Riguardo l'organizzazione logica delle edizioni critiche digitali, si noti come la TEI abbia proposto nel proprio vocabolario i medesimi componenti presenti nelle edizioni a stampa sotto forma di elementi *ad hoc*, permettendo così di strutturare il documento codificato in modo analogo alle edizioni a stampa. Si prenda come esempio l'apparato critico. Le singole voci d'apparato possono essere inserite nel corpo del testo critico, in corrispondenza di dove occorrono le variazioni testuali che descrivono, oppure possono essere presentate insieme in modo unitario al di fuori del testo critico, in un punto distinto del documento, in modo analogo alla maggior parte delle edizioni a stampa.³²

³¹ Per una galleria di modalità di presentazione dei materiali di corredo, in particolare dell'apparato critico, all'interno di edizioni critiche digitali si rimanda alla sezione 3.3 in Martignano 2017.

³² Le linee guida TEI prevedono diversi metodi per collegare un apparato critico al testo, di cui il primo riportato è chiamato “parallel segmentation method”, in quanto il testo, nel punto in cui è presente una variazione testuale, viene suddiviso per presentare in parallelo le diverse varianti (cfr. <https://tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPLK>).

La riflessione preliminare alla presentazione della terza parte del modello è che con il termine “componenti” si fa riferimento sia ai contenuti sia alle strutture in cui i contenuti vengono organizzati all’interno di un’edizione critica. Queste strutture, per quanto siano nate in seno all’edizione critica pubblicata in forma di libro stampato, permangono come riferimenti per allestire anche le edizioni digitali. Perciò ha senso che tali componenti vengano riproposti come entità logiche anche in un modello pensato per essere applicato in contesto digitale.

3.6.2 La generalizzazione dei principali componenti di un’edizione critica

Gli elementi che costituiscono un’edizione critica possono essere suddivisi in due macro categorie: da un lato il testo allestito dall’editore, ovvero il testo critico, e dall’altro lato i materiali di corredo o “strumenti complementari” che accompagnano il testo e che sono subordinati ad esso. Entrambe le categorie rientrano nel concetto più ampio di componente dell’edizione.

I tre concetti di testo edito, materiale di corredo e componente dell’edizione si prestano alla descrizione di diverse tipologie di edizione, non solo di un’edizione critica. Per questo motivo e per facilitare il riuso del presente modello concettuale, sono state definite tre classi native dedicate, rispettivamente `Edited Text`, `Accompanying Material` e `Scholarly Edition Component`. Le prime due sono sottoclassi di `Scholarly Edition Component`, mentre quest’ultima è descritta come una sottoclasse di `lrmo:F2 Expression`. Il modello LRMOO prevede, infatti, che un’istanza della classe `lrmo:F2 Expression` possa essere costituita da più componenti che a loro volta sono istanze di `lrmo:F2 Expression`. La proprietà `lrmo:R5 has component (is component of)`, che collega un’espressione a un suo componente, è così definita:

“This property associates an instance of an F2 Expression X with a structural component Y that conveys a part of the overall work realized by X, such as volumes, chapters, or sections” (LRMOO v. 0.9, p. 38).

In questo caso il componente non è una parte del testo allestito dall’editore, ma una parte dell’edizione scientifica, che è comunque una delle possibili realizzazioni dell’opera, in base a quanto illustrato in precedenza. Il rapporto tra `Scholarly Edition` e `Scholarly Edition Component` è formalizzato attraverso la proprietà `edition has component (is component of edition)`, sottoproprietà di `lrmo:R5 has component (is component of)`.

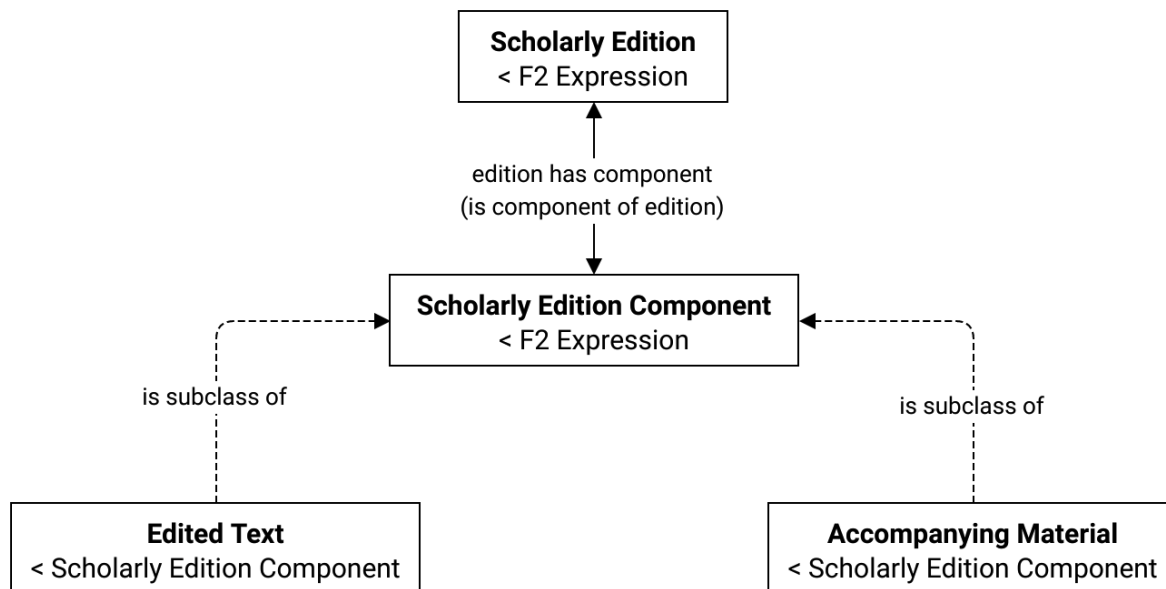


Figura 3.11 Schema dei principali componenti di un'edizione scientifica.

Il testo critico è, dunque, una sottoclasse di `Edited Text`. L'introduzione è una sottoclasse diretta di `Accompanying Material`, mentre l'apparato critico e il commento filologico sono sottoclassi indirette di `Accompanying Material`, come illustrato nel dettaglio alla sezione 3.6.7.

Questa soluzione rende il modello estremamente flessibile, consentendo di rappresentare le diverse modalità in cui un'edizione critica può essere allestita, descritte in Chiesa 2002:

“Essa [l'edizione critica] può consistere nella riproduzione dell'originale, se è conservato; o in un'ipotesi di ricostruzione dell'originale, se esso non è conservato; o ancora nella pubblicazione comparativa di testi diversi, ognuno dei quali gode della qualifica di originale o è comunque importante ad illustrare il processo di sviluppo testuale.”³³

Le ontologie esistenti sono state vagliate, ma nessuna offriva una soluzione adatta e flessibile quanto quella qui proposta.

Per quanto riguarda i materiali di corredo, nell'ontologia SE un concetto analogo è rappresentato dalla classe `se:Editorial Structure`, definita: “Scientific text structure depending on editorial purpose, e.g. an editing apparatus”. All'interno di questa classe sono comprese diverse classi, tra cui:

³³ Cfr. <https://lexiconse.uantwerpen.be/lexicon/editionCritical.html>.

- `se:Editorial Comment`, che rappresenta il commento che l'editore appone al testo critico;
- `se:Editing Apparatus`, definito “Editorial compositional structure in a scholarly edition to analyse text along certain criteria” e che rappresenta le diverse tipologie di apparato;
- `se:Editing Apparatus Entry`, che rappresenta la singola voce di un apparato;
- `se:Lemma`, ovvero la sequenza di parole del testo critico presentata all'inizio di una voce dell'apparato critico e che corrisponde con la lezione accolta, corretta o congetturata dall'editore;³⁴
- `se:Reading`, la lezione presentata in apparato.

Tuttavia, i concetti di lezione presentata nell'apparato (`se:Reading`) e di lezione stabilita dall'editore (`se:Lemma`) non si prestano a essere descritte come un materiale di corredo, in quanto rappresentano solo una parte di un particolare tipo di materiale di corredo, ovvero l'apparato critico. Inoltre, la definizione della classe `se:Editorial Structure` limita le modalità in cui il modello può essere espanso, poiché indica che il materiale di corredo è di natura testuale. Nell'ambito di un'edizione scientifica digitale il materiale di corredo deve poter essere anche un'immagine, un oggetto multimediale o un oggetto di altra natura. La definizione di `Accompanying Material` è più generica e flessibile.

La DoCO offre delle classi molto generiche che si rifanno alla struttura tradizionale del documento, quali `doco:Front Matter`,³⁵ `doco:Back Matter`,³⁶ `doco:Footnote`,³⁷ ecc. Trattandosi di un'ontologia che descrive i componenti di un documento in generale, i componenti sono stati modellati sulla base di come sono formattati graficamente e di dove sono collocati nel documento. Per formalizzare correttamente l'edizione critica occorrono, però, delle classi in grado di rappresentarne i diversi componenti in base alla funzione scientifica che ricoprono. Inoltre, i

³⁴ Cfr. definizione di `se:Lemma`: “Editorial structure in an apparatus entry with a copy of one or more words or phrases referring to its edition text, and as appearing in the original or a witness, or created by an editor, and judged by the editor to be authentic, accurate, and/or authoritative [, and possibly compared with variants referring to other witnesses]”.

³⁵ <https://sparontologies.github.io/doco/current/doco.html#d4e880>.

³⁶ <https://sparontologies.github.io/doco/current/doco.html#d4e262>.

³⁷ <https://sparontologies.github.io/doco/current/doco.html#d4e772>.

componenti non devono essere eccessivamente ancorati alla formattazione grafica delle edizioni a stampa, perché altrimenti il modello non si presterebbe alla rappresentazione di edizioni scientifiche digitali.

Per quanto riguarda il testo edito, la soluzione proposta coincide per certi versi con quella adottata nell'ontologia SE. La classe `se:Edition`, infatti, equivale a livello logico alla classe `Edited Text` e, come illustrato nella **Figura 3.3**, al suo interno racchiude altre tipologie di edizioni. La differenza tra l'ontologia SE e il presente modello risiede a monte nella definizione di edizione, come illustrato (vd. 3.5.3.1).

La DoCO, invece, ripresenta lo stesso problema sopra descritto anche per la descrizione del concetto di testo edito, dato che l'unica classe che offre, potenzialmente compatibile, è `doco:Body Matter`.³⁸

3.6.2.1 Il rapporto tra il testo edito e i materiali di corredo

In alcune edizioni critiche l'editore allestisce più testi per presentare l'opera e la sua tradizione testuale al meglio. Si pensi alle edizioni sinottiche, in cui l'editore presenta comparativamente due o più redazioni della medesima opera, oppure i casi analoghi all'edizione critica dei due poemetti in lingua d'oc, *Vita e passione di Santa Margherita d'Antiochia* (Lannutti 2012), in cui le edizioni dei due poemetti sono seguiti dall'edizione della fonte latina. La fonte latina è corredata da un apparato critico come le edizioni dei due poemetti. Per specificare che un materiale di corredo (`Accompanying Material`) pertiene a un determinato testo edito (`Edited Text`), si può usare facoltativamente la proprietà `accompanies (is accompanied by)`, che può essere definita come una sottoproprietà di `crm:P67 refers to (is referred to by)`.³⁹

³⁸ <https://sparontologies.github.io/doco/current/doco.html#d4e395>.

³⁹ “This property documents that an instance of E89 Propositional Object makes a statement about an instance of E1 CRM Entity. P67 refers to (is referred to by) has the P67.1 has type link to an instance of E55 Type. This is intended to allow a more detailed description of the type of reference. This differs from P129 is about (is subject of), which describes the primary subject or subjects of the instance of E89 Propositional Object” (CIDOC CRM v. 7.1.2, p. 155), http://cidoc-crm.org/cidoc-crm/7.1.2/P67_refers_to.

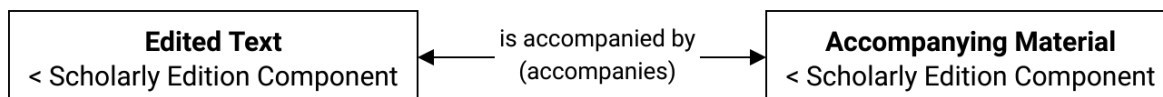


Figura 3.12 Il rapporto tra Edited Text e Accompanying Material.

3.6.3 La descrizione della *restitutio textus*

Il testo critico viene elaborato dall'editore sulla base della tradizione testuale e degli esiti delle precedenti fasi del lavoro critico, ovvero la *recensio* e la *collatio*. Per semplificare, si può dire che il testo critico sia una revisione dei testimoni, che derivi dai testimoni. Il modello LRM₀₀ permette di rappresentare le situazioni in cui un'istanza della classe `lrmo0:F2 Expression` è stata usata come base per produrre un'altra istanza di `lrmo0:F2 Expression` mediante la proprietà `lrmo0:R76 is derivative of (has derivative)`, definita nel seguente modo:

“This property associates an instance of F2 Expression with another instance of F2 Expression which was its source or one of its sources. [...] The property R76.1 has type of this property allows for specifying the kind of derivation, such as translation, revision, etc.” (LRM₀₀ v. 0.9, p. 52)

Per collegare il testo critico e più in generale il testo edito (Edited Text) al testimone (Witness) si propone una sottoproprietà di `lrmo0:R76 is derivative of (has derivative)`, denominata `is basis for (is based on)`. La proprietà collega un'istanza della classe `Critical Text` a una sola istanza della classe `Witness`, nel caso di un'edizione critica monotestimoniale, a più istanze di `Witness` nel caso di un'edizione pluritestimoniale.

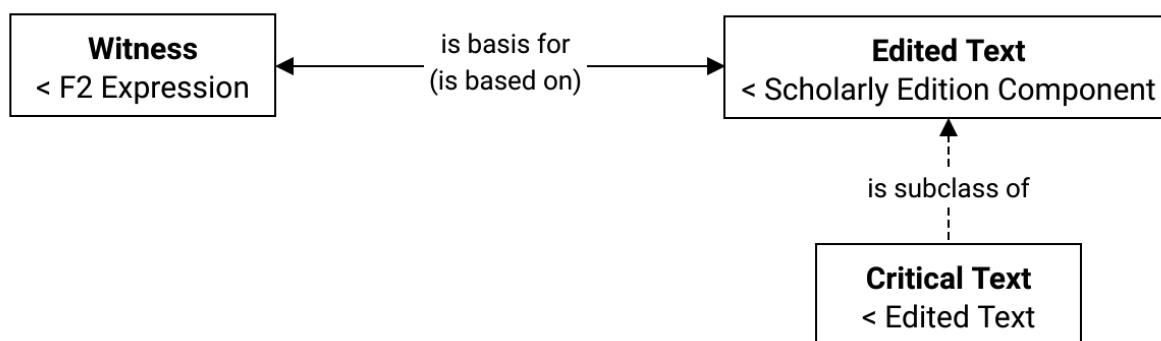


Figura 3.13 Schema del rapporto tra Witness ed Edited Text.

Nelle edizioni critiche pluritestimoniali avviene frequentemente che l'editore si serva di un manoscritto di base,⁴⁰ per rendere il testo critico il più possibile coerente e uniforme. Il manoscritto di base può essere scelto dall'editore sia in virtù di caratteristiche intrinseche (il grado di affidabilità o di leggibilità) sia per il posizionamento nello stemma. Per le stesse necessità di coerenza e uniformità, l'editore può prendere a riferimento il testo di un testimone, detto manoscritto di superficie,⁴¹ anche per ricostruire l'assetto grafico-fonetico del testo critico. Il manoscritto di superficie viene scelto in quanto considerato dall'editore più vicino da un punto di vista geografico e temporale all'originale e, quindi, rappresentativo della lingua in cui l'opera è stata prodotta.⁴²

Come esempio di utilizzo di un *manuscrit de surface* si riporta un estratto dell'introduzione dell'edizione critica del Libro della natura degli animali (Checchi 2020), nel quale l'editore segnala il testimone scelto e presenta le motivazioni della scelta (cfr. *ibid.*, p. 187):

“Si impiega Ch1 come manoscritto base di riferimento per le varianti formali (*manuscrit de surface*) [...]. La scelta è ricaduta su Ch1 in quanto testimone meno innovativo, più antico e pisano. Presentando inoltre una *mise en page* e una suddivisione in fascicoli molto vicina a quella dell'archetipo, è probabile che la sua *facies* linguistica sia la più prossima a quella del supposto autore pisano della sezione A, o quantomeno a quella dell'archetipo.”

La classe *Witness* ha, dunque, due sottoclassi, *Base Witness* e *Surface Witness*, che possono essere collegate esclusivamente alla classe *Critical Text* attraverso le proprietà *is mainly based on* (*is main basis for*) la prima e *is linguistically based on* (*is linguistic basis for*) la seconda. Entrambe sono sottoproprietà di *is based on* (*is basis for*).

⁴⁰ Il concetto di manoscritto di base potrebbe essere frainteso con quello di base di collazione, ovvero il testimone che l'editore utilizza in fase di collazione come riferimento rispetto al quale registrare le varianti presenti nel resto della tradizione. In questo contesto con manoscritto di base si intende il testimone che viene usato per preparare il testo critico e che viene corretto tramite uno o più testimoni detti di controllo (cfr. Leonardi 2022, p. 158).

⁴¹ Con manoscritto di superficie si intende il manoscritto da seguire per la forma del testo critico (cfr. Leonardi 2022, p. 164).

⁴² Nei casi in cui l'oggetto dell'edizione sia un'opera composta da più testi, si può scegliere il *manuscrit de surface* anche con l'obiettivo di garantire la maggiore uniformità possibile, preferendo quindi quel manoscritto che contiene il numero massimo di testi, idealmente tutti i testi.

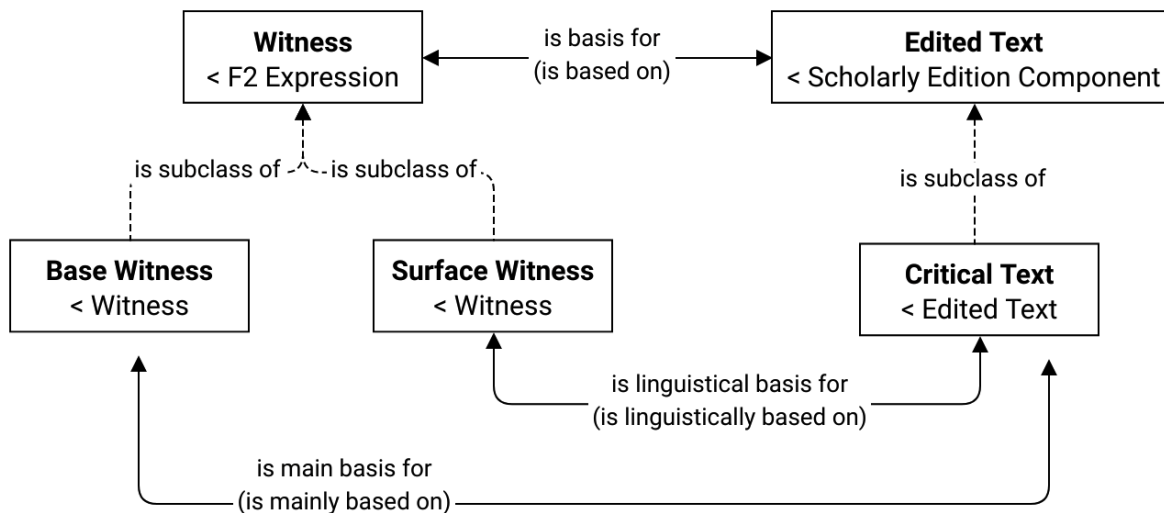


Figura 3.14 Schema del rapporto tra il testo critico e i testimoni.

La proprietà `is based on` (`is basis for`) si presta a essere estesa mediante la definizione di sottoproprietà che specifichino il rapporto tra un testimone e altre tipologie di testo edito. Nel caso in di una trascrizione diplomatica, per esempio, si potrebbe definire la proprietà `is diplomatic transcription of` per collegare l'istanza di `Witness` che rappresenta il testimone trascritto all'istanza della classe `Edited Text` o di una nuova sottoclasse *ad hoc* quale `Diplomatic Transcription`.

Le classi e le proprietà finora descritte delineano il rapporto tra il testo critico e la tradizione a livello macroscopico. Per descrivere nel dettaglio il lavoro filologico svolto dall'editore e i dati che rientrano nell'edizione critica, occorrono ulteriori classi e proprietà descritte nelle prossime sezioni.

3.6.4 La lezione

L'allestimento del testo critico richiede che l'editore intervenga su singoli luoghi del testimone (o dei molteplici testimoni) alla base dell'edizione critica e che nell'edizione segnali chiaramente il modo e il motivo in cui è intervenuto sul testo.

Il punto di partenza per descrivere a livello microscopico il rapporto tra il testo critico e la tradizione è il singolo luogo del testimone, detto lezione. La lezione può essere definita come ciò che si legge in un testimone.⁴³

Il concetto di lezione viene definito nell'ontologia SE e nella CAO. Quest'ultima descrive l'apparato critico, per cui la classe `cao:Reading`⁴⁴ rappresenta la lezione che l'editore offre all'interno di una voce dell'apparato critico ed è definita nel seguente modo:

“A reading is a **scholarly claim** about or **interpretation** of a **particular textual fragment contained in a witness**. Multiple readings may be given for the same textual fragment” (CAO, mia enfasi).

Anche nell'ontologia SE la classe `se:Reading`⁴⁵ rappresenta la lezione offerta nell'apparato critico, poiché è definita: “Copy of one or more words or phrases of an original or witness” ed è sottoclasse di `se:Editorial Structure`. Le definizioni di lezione offerte dalla CAO e dall'ontologia SE non coincidono, dunque, con quella precedentemente fornita: ciò che si legge in un testimone.

Nella descrizione dell'ontologia CAO, viene precisato che il concetto di lezione è rappresentato da due classi distinte:⁴⁶ la classe `frbroo:F23 Expression Fragment`,⁴⁷ che coincide con la lezione intesa come la porzione di testo contenuta nel testimone; la classe `cao:Reading` che invece è descritta come il risultato di un atto interpretativo, modellato attraverso la classe dell'ontologia

⁴³ La definizione presa principalmente a riferimento è quella di Leonardi 2022: “ciò che si legge nel manoscritto”. Altre definizioni analoghe rispetto a quella fornita nel testo sono quella presente in Chiesa 2002 (“le forme presenti nei testimoni (tecnicamente chiamate lezioni, ‘ciò che si legge’”) e quella contenuti in Stussi 1994 (“Con lezione di un determinato testimone si designa un passo del testo tramandato così come compare in tale testimone”).

⁴⁴ <https://fgiovannetti.github.io/cao/#Reading>.

⁴⁵ <https://github.com/nie-ine/e-editions/blob/master/ontology/scholarly-editing.ttl>.

⁴⁶ Cfr. CAO: “CAO uses the class `cao:Reading` to represent the concept of reading as interpretation, while it exploits the class `lrmoos` to represent the concept of reading as text. In fact, instances of the class `cao:Reading` are regarded as products of scholarly interpretation acts. A reading is generated by an operation of decoding and analysis of the source text that different scholars may interpret, or read, in different ways”

⁴⁷ L'ontologia CAO utilizza una versione precedente del modello LRMoo, denominata FRBRoo, all'interno della quale è prevista la classe F23 Expression Fragment per descrivere una porzione di testo. Questa classe è stata deprecata all'interno di LRMoo, che propone come soluzione alternativa l'utilizzo di `crm:E90 Symbolic Object`.

HiCO⁴⁸ `hico:Interpretation Act`,⁴⁹ che l'editore compie a partire dal testo del testimone, rappresentato da `frbroo:F2 Expression`, come si vede nella figura sotto.

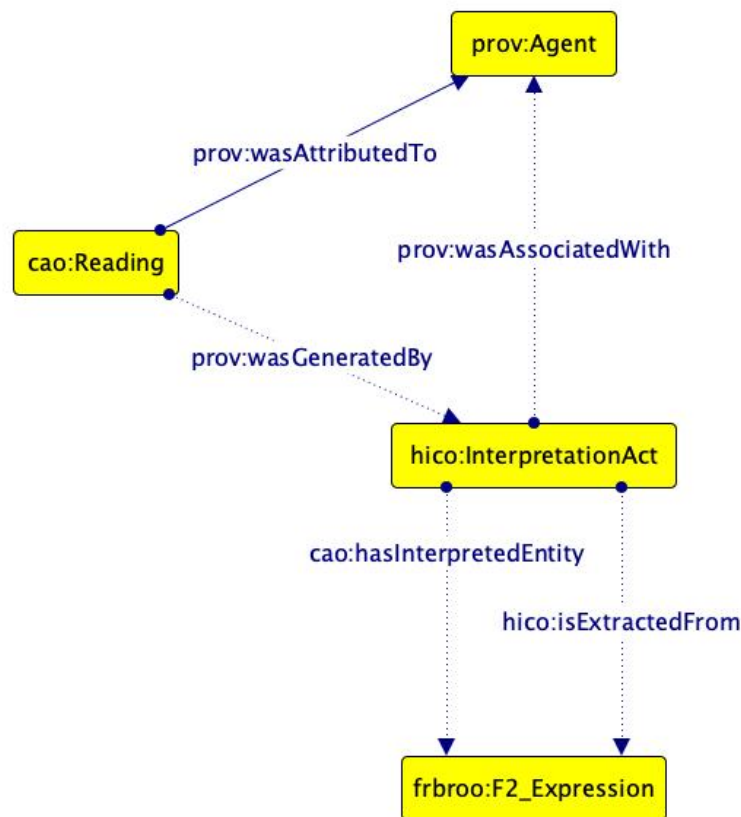


Figura 3.15 Grafo delle classi previste dalla CAO per descrivere il concetto di lezione (CAO).

Si offrono due esempi concreti di come le classi dell'ontologia CAO possano essere utilizzate per rappresentare delle lezioni.

Un esempio è ripreso dal manuale di Brambilla Ageno,⁵⁰ tratto dalla canzone *L'animo è turbato* di Neri de' Visdomini. La canzone è trasmessa da un solo manoscritto e al suo interno, il verso 69 figura così:

coloma contrariato

⁴⁸ Cfr. HiCO: "It allows to represent and reason on reliability of argumentations around attributions, by evaluating features such as motivations, types of cited sources or criteria, dates, relations with other claims (e.g. agreement/disagreement). Specifically, historical context regards events and situations that are part of the life-cycle of cultural heritage objects. For instance, being created by somebody, or being created at a certain time, are events related to an artefact that are claimed by an agent at a certain time, motivated with usage of primary sources, and recorded in a secondary source (e.g. a cataloguing record)."

⁴⁹ <https://marilenadaquino.github.io/hico/#d4e229>.

⁵⁰ Brambilla Ageno 1984, p. 123.

Da Contini è stato interpretato come:

*co l'om ha contrariato

Ma Brambilla Ageno propone un'interpretazione diversa:

c'ò lo ma[l] contrariato

Mediante le classi CAO, queste informazioni possono essere descritte come in figura.

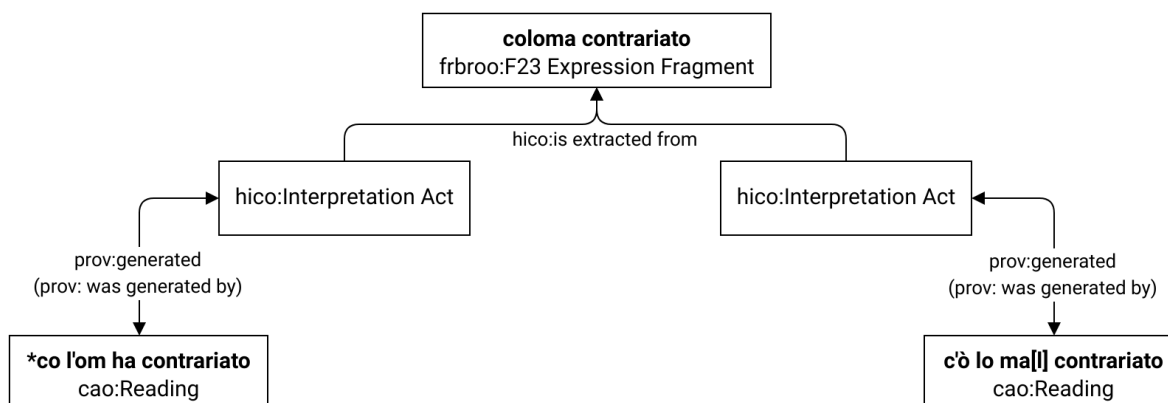


Figura 3.16 Esempio di applicazione della CAO per descrivere due diverse lezioni.

La soluzione proposta dalla CAO permette di descrivere i casi in cui due o più editori o copisti, hanno letto diversamente il medesimo passo contenuto all'interno di un testimone. Grazie all'inclusione della classe `prov:Agent` la CAO offre la possibilità di attribuire a un editore la responsabilità per l'interpretazione (l'istanza di `cao:Reading`). All'interno di un'edizione può capitare che l'editore riporti esplicitamente lezioni proposte da altri studiosi, per confutarle oppure per confermarle. Come esempio si riporta un verso tratto dall'edizione di un componimento trobadorico anonimo in lingua occitanica,⁵¹ curata da Luca Gatti e pubblicata nel contesto di Rialto.⁵²

Il verso 7 del componimento si presenta così all'interno dell'unico testimone che lo tramanda:

Bolaudir nouas oson.

Mentre nell'edizione critica è stato reso nel seguente modo:

vol audir novas o son,

⁵¹ [http://www.rialto.unina.it/An/461.209a\(Gatti\).trad.htm](http://www.rialto.unina.it/An/461.209a(Gatti).trad.htm).

⁵² Repertorio informatizzato dell'antica letteratura trobadorica e occitana. <http://www.rialto.unina.it/>.

Nella voce d'apparato associata al verso, l'editore illustra come la parola "son" sia stata oggetto di diverse interpretazioni: alcuni studiosi l'hanno interpretata come un verbo, traducendola in "sono", mentre l'editore propone un'altra interpretazione, intendendo che indichi una canzone:

"[...] L'intervento è ignorato da Blasi, che traduce il verso così: «vuol sentire dove **sono** notizie». Il senso conferito al passo, pur accettabile, non risulta particolarmente soddisfacente. [...] *Son* sarà qui da intendersi, più che nell'accezione di 'melodia', nel significato generico di '**canzone**'"

In questo caso a una sola istanza della classe `frbroo:F23 Expression Fragment` ("son"), che non pone particolari problemi da un punto di vista grafico, sono associate due istanze distinte della classe `cao:Reading`, una elaborata dall'editore e l'altra da un agente esterno.

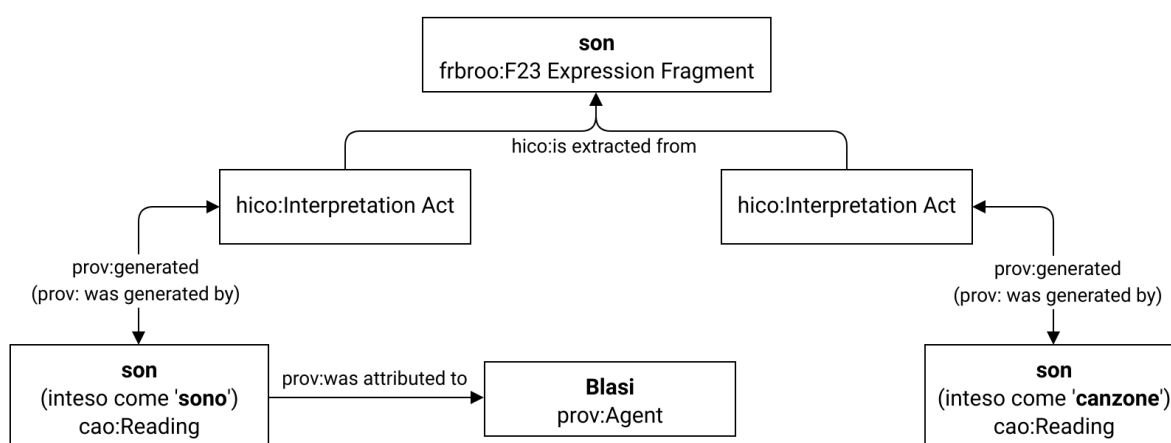


Figura 3.17 Esempio di utilizzo della CAO per attribuire una lezione a un editore esterno.

L'ontologia CAO offre un livello di dettaglio nella rappresentazione del concetto di lezione che è utile anche nel contesto del presente modello. Tuttavia, rimane il problema della definizione di `cao:Reading`. Per un filologo con lezione si intende appunto quanto si legge nel testimone, ovvero il concetto che nella CAO è rappresentato mediante la classe `frbroo:Expression Fragment`. Per una questione di coerenza tra il linguaggio filologico e la terminologia utilizzata nel presente modello la classe `cao:Reading` non è utilizzabile per descrivere il concetto di lezione, perché potrebbe provocare dei fraintendimenti. Inoltre, bisogna considerare che l'ontologia CAO è basata sulle linee guida TEI, ed è stata perciò sviluppata per rappresentare l'apparato critico all'interno di un'edizione critica digitale. Il concetto di lezione espresso da `cao:Reading` corrisponde all'elemento `<rdg>` delle linee guida TEI e serve più che altro come un'unità logica per la descrizione della variazione testuale.

In generale, nel linguaggio filologico lezione è un *umbrella-term* utilizzato per riferirsi a molteplici concetti leggermente differenti tra di loro. Può essere la sequenza di caratteri o parole presente in un testimone. Può essere l'interpretazione da parte di un editore di una certa sequenza di caratteri contenuta in un determinato testimone. Quando si parla di "lezione del copista" si fa riferimento a ciò che il copista ha letto nell'antigrafo così come a ciò che ha effettivamente trascritto nella propria copia. Con "lezione congetturale" si intende ciò che si dovrebbe leggere nell'originale o in un testimone perduto secondo l'editore. Nell'apparato critico una lezione comune a più testimoni corrisponde in realtà a sequenze di caratteri conservate in testimoni diversi e che non sono graficamente identiche. L'elenco potrebbe proseguire con i concetti di *lectio difficilior*, *lectio faciliior*, *lectio singularis* e altri, ma quanto riportato finora è sufficiente a dimostrare la mancanza di chiarezza rispetto alla definizione del concetto di lezione. L'ambiguità si genera tra la lezione intesa come la sequenza di caratteri o parole oggetto dell'interpretazione e la lezione intesa come frutto di un'interpretazione.

Occorre stabilire come si vuole usare l'etichetta lezione, se per definire la sequenza di caratteri presente nel testimone o la sua interpretazione (come nella CAO). Ci si potrebbe chiedere perché sia importante separare i due concetti. Ci sono due motivi interdipendenti. Prima di tutto, come già dimostrato negli esempi, occorre poter attribuire in modo chiaro interpretazioni diverse da parte di editori diversi, lasciando anche al lettore la possibilità di verificare il lavoro di interpretazione. Ciò è possibile distinguendo in modo chiaro tra l'oggetto dell'interpretazione e il risultato dell'interpretazione. Il secondo motivo dipende dal fatto che in ambito digitale, per poter codificare correttamente le informazioni, è necessario dividere i diversi livelli di analisi del testo, distinguendo la sequenza di caratteri conservata in una fonte dalla rispettiva lettura.⁵³

Una soluzione che risponde alle diverse problematiche enunciate consiste nella definizione di due classi native: come frutto dell'interpretazione la classe `Reading In Apparatus` equivalente a `cao:Reading`, così da poter riutilizzare la CAO in modo coerente con il dominio del presente modello; come oggetto dell'interpretazione la classe `Reading`, definita come ciò che può essere letto

⁵³ Cfr. "L'editore tende a leggere un manoscritto mescolando il punto di vista semantico con quello grafico. Questo è metodologicamente scorretto, dal punto di vista dell'informatica, almeno finché non sia dichiarato esplicitamente, e purché se ne traggano le debite conseguenze" (Orlandi 1990, p. 137).

in un testimone, rimarcando che si tratta di una sequenza di caratteri, parole o frasi contenuta in un testimone conservato o andato perduto.

La classe `Reading` può essere descritta come sottoclasse di `crm:E90 Symbolic Object`,⁵⁴ così definita:

“This class comprises identifiable symbols and any aggregation of symbols, such as characters [...]. An instance of E90 Symbolic Object may or may not have a specific meaning, for example an arbitrary character string.” (CIDOC CRM v. 7.1.2, p. 110).

Il modello CRM offre anche la classe `crm:E33 Linguistic Object`,⁵⁵ che descrive delle espressioni in linguaggio naturale. Questa classe, però, può essere usata solo per sequenze di parole di senso compiuto, mentre la porzione di testimone analizzata dall'editore può essere una sequenza di caratteri che, estrapolata dal proprio contesto o perché frutto di una deformazione dovuta ai copisti, non ha significato. Si predilige così l'utilizzo della classe `crm:E90 Symbolic Object`.

Il modello LRM₀₀ prevede che un'istanza di `lrmo0:F2 Expression` possa essere collegata a un proprio frammento, rappresentato da `crm:E90 Symbolic Object`, mediante la proprietà `lrmo0:R15 has fragment (is fragment of)`.⁵⁶ Per collegare `Witness` e `Reading` si propone una sottoproprietà di `lrmo0:R15 has fragment (is fragment of)`, denominata `has reading (is reading of)`.

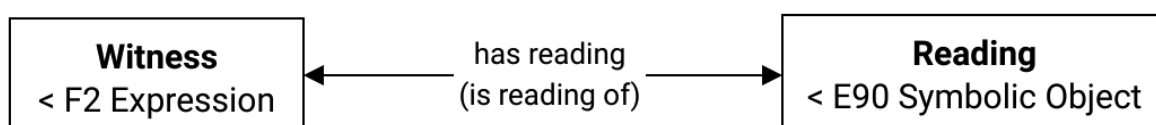


Figura 3.18 Schema del rapporto tra Witness e Reading.

3.6.4.1 Omissioni, lacune e passi illeggibili

Ci sono casi in cui un luogo del testimone non può essere letto, a causa dello stato di conservazione del testimone o per altre ragioni legate al modo in cui si presenta all'interno del proprio supporto fisico. In questo caso l'editore prova a leggere e interpretare il passo, ma, non riuscendovi, lo dichiara

⁵⁴ http://cidoc-crm.org/cidoc-crm/7.1.2/E90_Symbolic_Object.

⁵⁵ http://cidoc-crm.org/cidoc-crm/7.1.2/E33_Linguistic_Object.

⁵⁶ Cfr. “This property associates an instance of E90 Symbolic Object with an instance of F2 Expression of which it is a fragment. The fragment is not itself an instance of F2 Expression as it does not express any instance of F1 Work. When the fragment consists of intelligible words it is an instance of E33 Linguistic Object.” (LRM₀₀ v. 0.9, p. 41)

illeggibile. Se l'editore riesce comunque a intravedere alcuni caratteri o perlomeno ad avere un'idea del numero di caratteri e di parole mancanti, può fornire una lettura ipotetica, segnalando che si tratta di un passo incerto.

Si offre un esempio tratto dalla ballata *Amor, per ti sempre ardo*, in corso di edizione da parte di Vittoria Brancato nel contesto del progetto European Ars Nova. La ballata è attestata in forma frammentaria da un manoscritto, che in corrispondenza del v. 6, tra “pa” e “sse” presenta un buco.

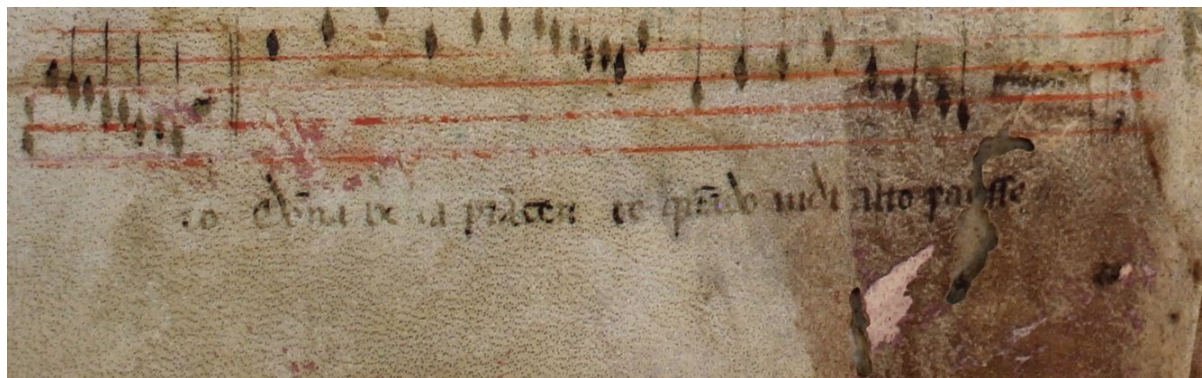


Figura 3.19 Esempio di manoscritto danneggiato che presenta dei buchi.

A causa del buco non è possibile leggere integralmente l'ultima parola del verso. Nel testo critico, la presenza del buco viene segnalata mediante dei segni diacritici.

quando vidi alto pa[...]sse

In altri casi in un testimone può mancare una porzione di testo. Può trattarsi di una lacuna meccanica, cioè quando una parte del manoscritto è stata fisicamente perduta, oppure di un'omissione volontaria o involontaria da parte del copista.

Si offre un ulteriore esempio, proposto nel manuale di Brambilla Ageno (1984), di un caso di “omissione sicura” tratto dall'autografo delle *Sposizioni* di Sacchetti (Laur. Ashb. 574). In un punto del testo si legge:

Perché non siamo giusti discesi da Dio giusto, come peccatori figliuoli d'Adamo peccatore.

Il senso della frase è incoerente, per cui l'editore può ipotizzare che una o più parole siano state involontariamente omesse e che la frase dovesse essere:

Perché non siamo giusti discesi da Dio giusto, [ma] come peccatori figliuoli d'Adamo peccatore.

Da un punto di vista concettuale, non siamo di fronte a vere e proprie lezioni (intese come ciò che può essere letto in un testimone), dato che la lettura è impedita, incerta o del tutto impossibile. L'editore rileva la presenza di lezioni parzialmente o del tutto illeggibili oppure mancanti all'interno del testimone, in corrispondenza di un particolare luogo del testo.

Per rappresentare su un piano più astratto i concetti di omissione, lacuna, passo illeggibile, ecc. occorrono delle classi a sé. Applicando il principio di economia, solitamente rispettato nello sviluppo delle ontologie, si propongono solo due classi native: `Gap` e `Unclear Passage`. Entrambe sono state ricalcate su due elementi delle linee guida TEI, rispettivamente `<gap>` e `<unclear>`, che si prestano a descrivere tutti i casi sopra delineati. Il primo, infatti, indica “a point where material has been omitted in a transcription, whether for editorial reasons described in the TEI header, as part of sampling practice, or because the material is illegible, invisible, or inaudible.”⁵⁷ Il concetto espresso da `<gap>` può essere adatto a rappresentare lacune e omissioni volontarie o involontarie. Per le lacune le linee guida TEI mettono a disposizione degli elementi vuoti, per segnalare in apparato l'inizio e la fine di una lacuna.⁵⁸ Tuttavia, i concetti di omissione e di lacuna possono essere considerati equivalenti.⁵⁹ L'elemento `<unclear>`, invece, contiene “a word, phrase, or passage which cannot be transcribed with certainty because it is illegible or inaudible in the source.”⁶⁰ La differenza rispetto al concetto espresso da `<gap>` è che, nel caso di un passo incerto, nel testimone si vede la presenza di una sequenza di caratteri, che però è illeggibile o che non può essere letta con certezza.

In che modo si rapportano `Gap` e `Unclear Passage` con il testimone? La prima classe può essere intesa come una sequenza di caratteri o parole che manca all'interno del testimone, la seconda come una sequenza di caratteri o parole presente nel testimone ma di lettura incerta. Entrambe le classi potrebbero essere rappresentate come dei particolari tipi di `crm:E90 Symbolic Object`, alla stessa stregua della classe `Reading`. Per questo motivo, si propone di formalizzare `Gap` e `Unclear Passage` come due sottoclassi di `Reading`, collegate a `Witness` dalla classe `has reading (is reading of)`.

⁵⁷ <https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-gap.html>.

⁵⁸ <https://tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPMI>.

⁵⁹ Si rimanda alla definizione di lacuna proposta in Leonardi 2022: “lacuna: omissione di una parte di testo. La lacuna è detta meccanica quando dipende dalla perdita fisica di una parte del manoscritto”.

⁶⁰ <https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-unclear.html>.

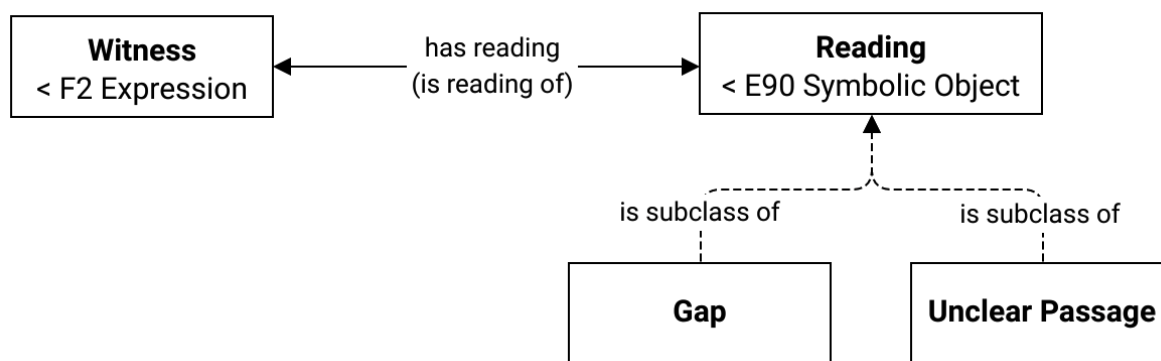


Figura 3.20 Schema delle classi *Gap* e *Unclear Passage* e del loro rapporto con *Reading* e *Witness*.

3.6.5 L'intervento critico: la descrizione della *restitutio textus* a livello microscopico

Sulla base delle lezioni e degli altri fenomeni testuali riscontrati nella tradizione testuale, l'editore stabilisce il testo critico, segnalando gli interventi effettuati e offrendo nell'apparato critico una panoramica sullo stato della tradizione testuale e le motivazioni dei propri interventi.

Per modellare questo lavoro filologico a livello microscopico si utilizza come punto di partenza la CAO, all'interno della quale la lezione offerta in apparato è il frutto di un atto interpretativo, rappresentato dalla classe `hico:Interpretation Act`, come già illustrato alla sezione 3.6.4. Espandendo la CAO, si può dire che l'intervento critico è un particolare tipo di atto interpretativo, mediante il quale l'editore stabilisce la porzione di testo critico e le lezioni da presentare nell'apparato critico.

La classe nativa `Critical Intervention` è sottoclasse di `hico:Interpretation Act`. A seconda che l'edizione sia monotestimoniale o pluritestimoniale, l'intervento critico è basato su una sola istanza di `Reading` (o di `Gap` o di `Unclear Passage`) o su più di un'istanza di tali classi. Per collegare l'intervento critico a `Reading`, si propone la proprietà `is carried out on`, sottoproprietà di `hico:is extracted from`.⁶¹ Nella CAO questa proprietà collega l'atto interpretativo (`hico:Interpretation Act`) direttamente al testimone (classe `frbroo:F2 Expression`) come si vede nella **Figura 3.15**. La soluzione della CAO rispecchia il fatto che l'editore, quando interviene sul testo, non analizza solo la singola porzione di testo problematica, ma tiene in conto anche il contesto più ampio in cui la lezione è inserita e in generale le caratteristiche

⁶¹ <https://marilenadaquino.github.io/hico/#d4e115>.

del testimone. Tuttavia, nel presente modello è sembrato opportuno collegare l'atto interpretativo alla singola lezione, per poter mettere più facilmente a confronto le lezioni dei diversi testimoni.

L'intervento critico richiede che l'editore presenti in apparato le lezioni e fornisca tutte le informazioni ad esse relative, che aiutino il lettore a comprendere i propri ragionamenti. La classe `Critical Intervention` è collegata alla classe `Reading In Apparatus` dalla proprietà `prepared by`, sottoproprietà di `prov:was generated by` usata nella HiCO.

Una porzione del testo critico viene stabilita dall'editore, questa porzione è rappresentata dalla classe nativa `Critical Text Passage`, che può essere modellata come una sottoclasse di `crm:E90 Symbolic Object`, dato che può corrispondere a una sequenza di parole di senso compiuto o a pochi caratteri. La proprietà `set by`, sottoproprietà di `prov:was generated by`, collega `Critical Text Passage` alla classe `Critical Intervention`. La porzione del testo critico è collegata al testo critico dalla proprietà `has passage (is passage of)`, sottoproprietà di `lrmo0:R15 has fragment (is fragment of)`.

Il grado zero dell'intervento critico è quello che mira ad adattare l'assetto grafico-fonetico del testo conservato nei testimoni all'uso moderno, affinché il testo risulti più facilmente comprensibile al lettore. Si può dire che il problema testuale da risolvere sia, banalmente, quello della leggibilità. Per mostrare concretamente in che cosa consista l'intervento critico dell'editore in casi simili, si offre un esempio tratto dalla già menzionata edizione digitale 'Petrarchiv'. La figura sotto riporta l'incipit del componimento che apre il Canzoniere di Petrarca, *Voi ch'ascoltate in rime sparse il suono*, così come figura all'interno del codice Vat. lat. 3195.

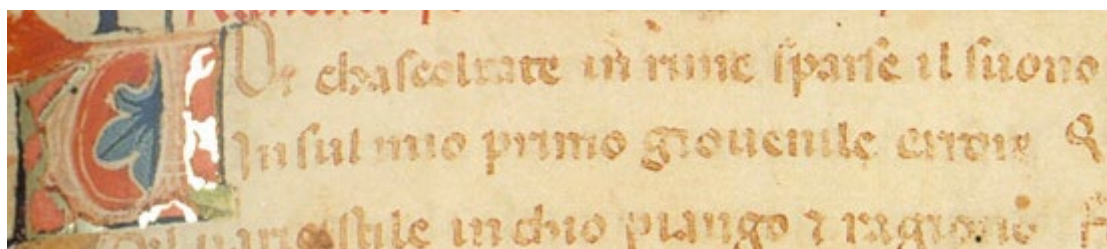


Figura 3.21 Incipit del Canzoniere di Petrarca nel codice Vat. lat. 3195.

La sequenza di caratteri "Uoi ch'ascoltate in rime sparse il suono" è la lezione del testimone. Nell'edizione digitale il verso è presentato al lettore nel testo critico in veste interpretativa: "Voi ch'ascoltate in rime sparse il suono". Dunque l'intervento critico è l'atto mediante il quale la lezione

contenuta nel codice Vat. lat. 3195 è stata interpretata dall'editore e presentata nel testo critico in una veste più leggibile. La pratica comune è che in un'edizione critica l'intero testo sia presentato in forma interpretativa e regolarizzata da un punto di vista grafico-fonetico, per cui molto spesso l'editore segnala direttamente nell'introduzione i criteri editoriali adottati, senza riportare la forma originale delle lezioni in apparato.

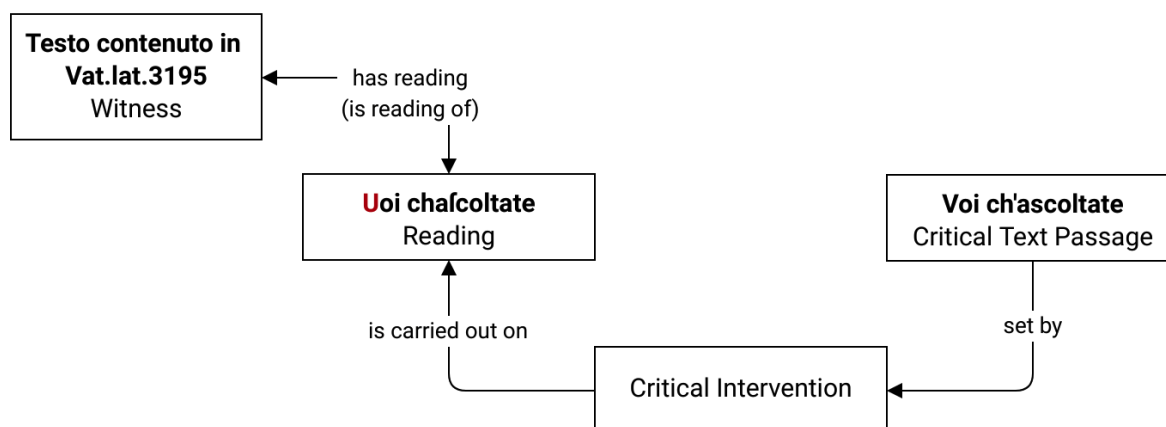


Figura 3.22 Esempio di come la regolarizzazione grafica di una lezione può essere formalizzata nel modello concettuale.

La classe *Reading In Apparatus* può essere utilizzata anche nelle edizioni monotestimoniali, quando l'editore riporta la lezione nella sua veste formale originaria, quando la lezione è stata corretta o modificata dall'editore o quando la lezione presenta delle peculiarità che è necessario segnalare al lettore.

Nel caso delle edizioni pluritestimoniali, l'editore è tenuto a presentare nell'apparato le lezioni scartate. Come esempio si propone l'edizione della *Commedia* curata da Petrocchi.⁶² All'inizio del verso 4 del primo canto dell'*Inferno* il codice Trivulziano 1080 riporta "Et", come si vede nell'immagine sotto.

⁶² Cfr. edizione della *Commedia* curata dal Petrocchi, pubblicata online dalla Società Dantesca Italiana (<https://danteonline.it/opere/>) e all'interno dell'edizione critica digitale della *Commedia* curata da Prue Shaw (<https://dantecomedia.it/index.html?ed=PET&canticle=1&canto=1>).

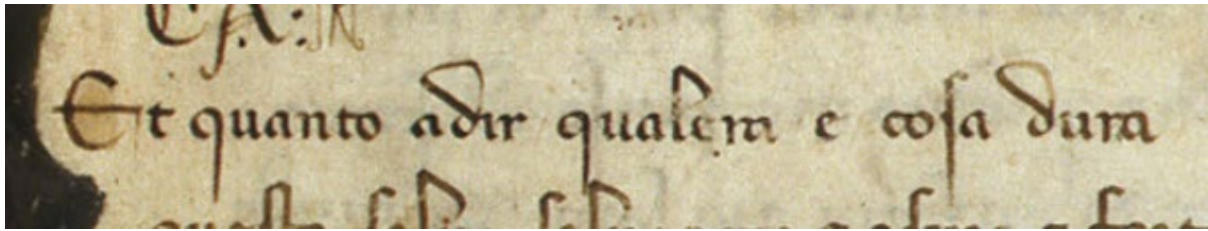


Figura 3.23 Verso 4 del Canto I dell’Inferno nel codice Trivulziano 1080.

Il codice Urbinates latino 366, invece, nel punto corrispondente riporta “Hai”.

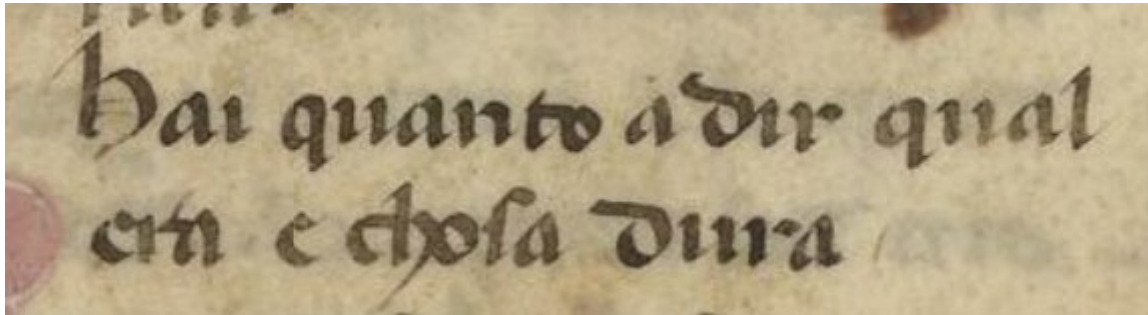


Figura 3.24 Verso 4 del Canto I dell’Inferno nel codice Urbinates latino 366.

Nell’edizione di Petrocchi il testo presenta la lezione del codice Urbinates, con una grafia leggermente diversa: “Ahi”. Questo caso può essere formalizzato mediante il presente modello come nella figura sotto.

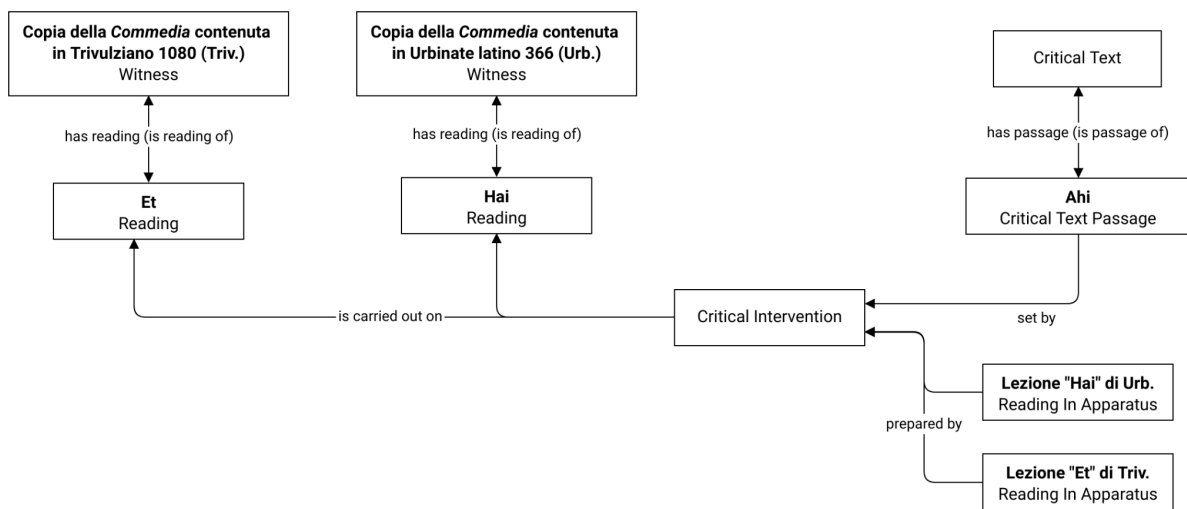


Figura 3.25 Esempio di come formalizzare un intervento critico con il modello concettuale.

Come illustrato nella sezione 3.6.4, è importante poter esplicitare chi ha effettuato un intervento, portando a una particolare interpretazione di una lezione o a proporre nel testo critico una correzione, una lezione o una congettura. Nella CAO l’atto interpretativo è collegato all’editore (rappresentato dalla generica classe `prov:Agent`) mediante la proprietà `prov:was associated`

with⁶³ (vd. **Figura 3.15**). La classe `Critical Intervention` è collegata alla classe `Editor` grazie alla proprietà nativa `performed by`, sottoclasse di `prov:was associated with`. In questo modo è possibile rappresentare i casi in cui un *équipe* di editori lavora alla medesima edizione o i casi in cui un editore riporta e confuta le soluzioni proposte in precedenza da altri editori. L'esempio illustrato nella **Figura 3.16**, nel presente modello concettuale viene, dunque, rappresentato come segue.

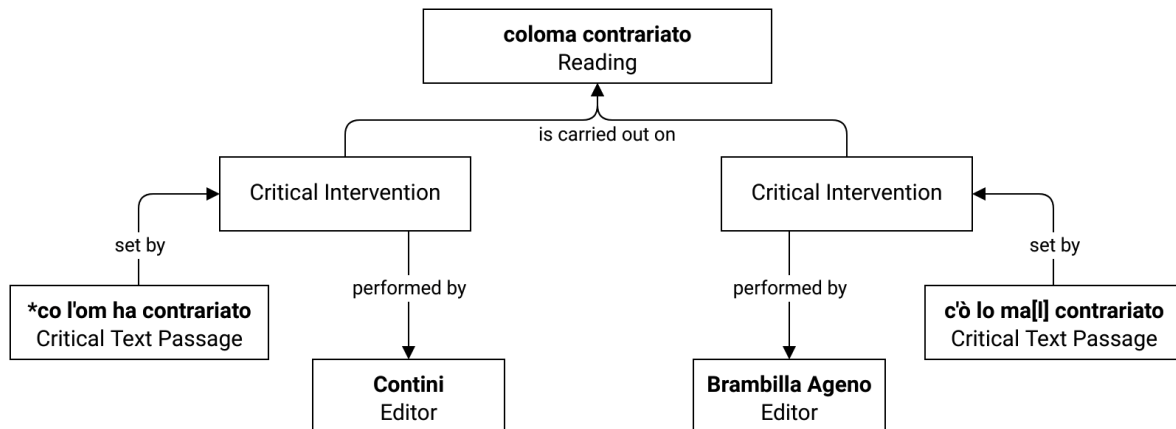


Figura 3.26 Esempio di come due diversi interventi critici sulla medesima lezione possono essere formalizzati con il modello concettuale.

Per rendere l'intervento critico da parte dell'editore il più possibile trasparente e verificabile, si possono riutilizzare due classi proposte nella HiCO: `hico: Interpretation Type`,⁶⁴ che permette di definire arbitrariamente il tipo di atto interpretativo effettuato, e `hico: Interpretation Criterion`,⁶⁵ definito: "The criterion or type of source used by an agent to support an interpretation act, such as bibliography, verbal communication, etc.". Nel presente modello si propongono due classi native, `Critical Intervention Type` e `Critical Intervention Criterion`, sottoclassi rispettivamente di `hico: Interpretation Type` e `hico: Interpretation Criterion`. La prima è collegata a `Critical Intervention` attraverso la proprietà `has intervention type`, sottoproprietà di `hico:has interpretation type`,⁶⁶

⁶³ "An activity association is an assignment of responsibility to an agent for an activity, indicating that the agent had a role in the activity. It further allows for a plan to be specified, which is the plan intended by the agent to achieve some goals in the context of this activity." (<https://www.w3.org/TR/prov-o/#wasAssociatedWith>).

⁶⁴ <https://marilenadaquino.github.io/hico/#d4e304>.

⁶⁵ <https://marilenadaquino.github.io/hico/#d4e291>.

⁶⁶ <https://marilenadaquino.github.io/hico/#d4e97>.

la seconda grazie alla proprietà `has intervention criterion`, sottoproprietà di `hico:has interpretation criterion`.⁶⁷

Le diverse modalità in cui l'editore stabilisce il passo del testo critico, ovvero i diversi tipi di intervento critico, possono essere definite attraverso altrettanti *named individual*⁶⁸ della classe `Critical Intervention Type`. Nel caso di un testo tramandato da più testimoni, nella tradizione della critica testuale si sono consolidate due principali modalità per stabilire il testo, individuate con dei termini latini affermatasi grazie al volume *Textkritik* di Paul Maas: *selectio* ed *emendatio*. La prima rappresenta la scelta che l'editore effettua tra le diverse lezioni attestate nella tradizione: se una delle lezioni viene ritenuta ascrivibile all'originale, viene accolta nel testo critico. La seconda rappresenta, invece, il caso in cui tra le lezioni trasmesse nessuna viene reputata ricevibile, per cui l'editore sana il luogo del testo. Entrando più nello specifico, l'editore può correggere tramite il ricorso a una lezione buona trasmessa da uno o più altri codici (*emendatio ope codicum*), basandosi sulle fonti dell'opera (*emendatio ex fontibus*) oppure in modo congetturale, cioè proponendo una soluzione non attestata in nessun testimone (*emendatio ope ingenii, divinatio*, 'congettura').⁶⁹ Nell'ontologia i concetti di *selectio*, *emendatio*, *emendatio ope codicum*, *emendatio ex fontibus* ed *emendatio ope ingenii* sono proposti come altrettanti *named individual* della classe `Critical Intervention Type`. A questi si aggiungono anche due tipi di intervento molto diffusi ovvero la correzione di un errore evidente e la regolarizzazione grafico-fonetica.

Si riporta un esempio di *selectio* tratto dal manuale di Brambilla Ageno (1984). In appendice l'autrice offre un'edizione critica di esempio di un testo tramandato da tre testimoni (P, V e L), commentando le scelte effettuate in fase di ricostruzione del testo. Al v. 42 dell'edizione si legge a testo:

che facemi ancosciare

Mentre nell'apparato critico:

42 che mi facie ancosciare V chemifaciangosciare L o facemi angosciare P.

⁶⁷ <https://marilenadaquino.github.io/hico/#d4e80>.

⁶⁸ Nel contesto di un'ontologia con il termine *individual* si indica un'istanza di una classe, che corrisponde a un oggetto o entità del dominio reale. Un *named individual* ha associato un identificativo, per essere riferibile dall'esterno anche da altre ontologie.

⁶⁹ Cfr. glossario disponibile in Leonardi 2022 (pp. 190-196).

Nella nota di commento, Brambilla Ageno motiva la propria scelta (*selectio*, appunto), in questo modo:

“La variante *mi faci(e)* di VL è indifferente, e solo la complessiva maggiore attendibilità di P ha fatto preferire la variante *facemi* [...]”

Questo esempio potrebbe essere formalizzato mediante l’ontologia come si osserva in figura.

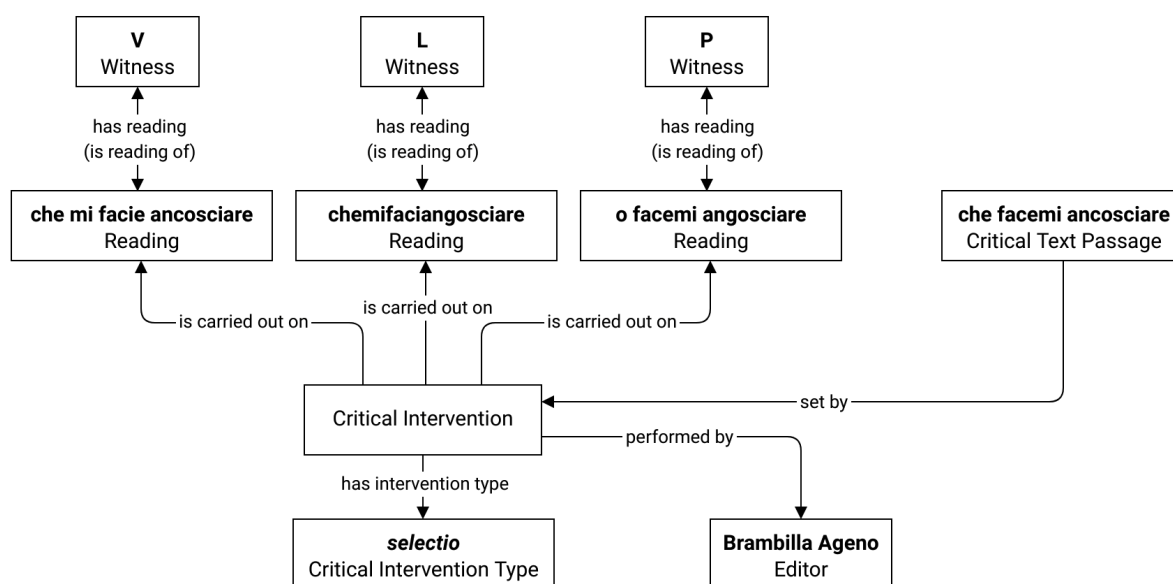


Figura 3.27 Esempio di *selectio* formalizzato con le classi del modello concettuale.

Per quanto riguarda i criteri su cui si basa un intervento critico, è possibile distinguere tra i criteri interni, “criteri di valutazione delle varianti in base alla loro plausibilità intrinseca” (Leonardi 2022), e criteri esterni, “criteri di valutazione delle varianti in base alla loro posizione nella genealogia della tradizione” (*ibid.*). In virtù di questa distinzione, si propongono due sottoclassi di *Critical Intervention Criterion* dedicate, rispettivamente *Internal Critical Intervention Criterion* ed *External Critical Intervention Criterion*.

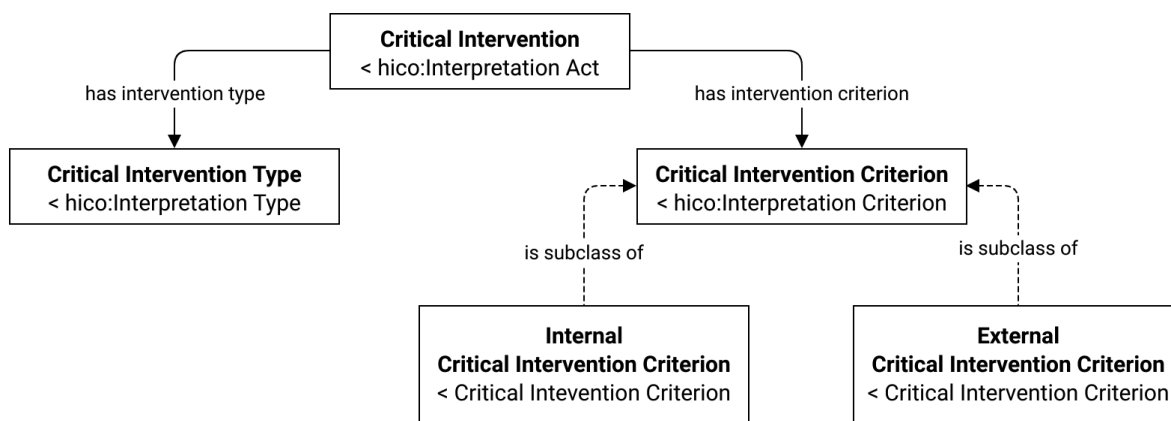


Figura 3.28 Schema delle classi per descrivere un intervento critico.

Per mostrare come la classe `Critical Intervention Criterion` possa essere usata, si propone un esempio tratto dal manuale di Brambilla Ageno (1984). Ai vv. 266-267 della frottola di Sacchetti *Chi drieto va* si legge:

Or pur lavora
ca la barba l'hai

Brambilla Ageno propone la suddivisione di “ca” in “c’a”, motivando come segue:

“ca per che, se è comune nei dialetti centrali e compare sporadicamente in autori toscani assai poco sorvegliati come Giovanni di Pagolo Morelli, sarebbe isolato nel fiorentino del Sacchetti, che non ne presenterebbe altro esempio in tutto l'autografo.” (Brambilla Ageno 1984, p. 33)

In altre parole, l'editrice è intervenuta sul testo secondo il criterio dell'*usus scribendi*, per rendere il testo coerente con la lingua dell'autore. Questo intervento critico può essere formalizzato come in figura.

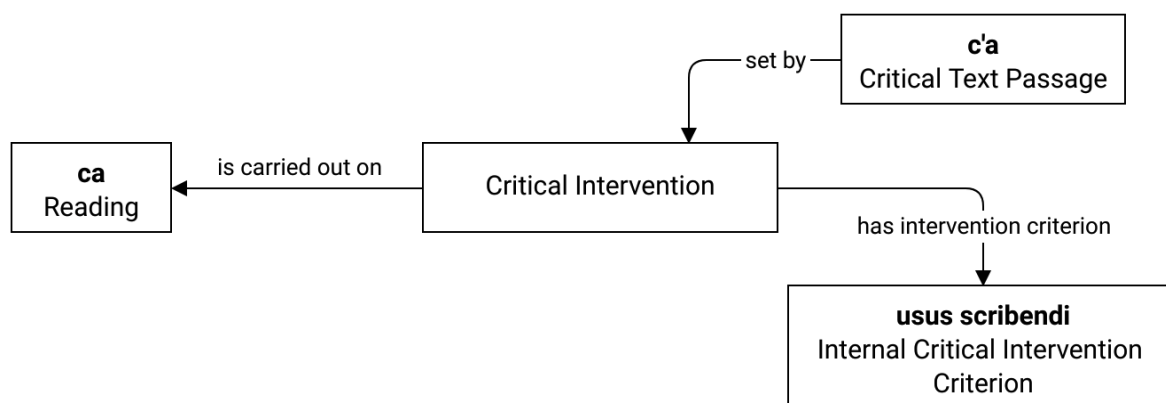


Figura 3.29 Esempio dell'utilizzo della classe `Critical Intervention Criterion`.

Un luogo del testo particolarmente problematico o caratterizzato da una notevole variazione testuale, può richiedere un intervento articolato da parte dell'editore, che dipende da molteplici ragioni. Per questo motivo un'istanza della classe `Critical Intervention` può essere collegata a più istanze delle classi `Critical Intervention Type` e `Critical Intervention Criterion`.

3.6.5.1 I segni diacritici

Nei casi in cui non sia possibile sanare un passo del testimone danneggiato, mancante o poco leggibile, l'editore può segnalare la presenza di questi fenomeni sia in apparato sia nel testo critico mediante dei segni diacritici.⁷⁰ La scelta dei segni diacritici e della formattazione grafica è arbitraria e dipende dall'uso nel campo disciplinare di cui si occupa l'editore e al limite da specifiche scelte personali.⁷¹ Ad ogni modo l'editore nell'introduzione spiega come ha allestito il testo critico, esplicitando in forma discorsiva o mediante una legenda apposita a cosa corrispondono le diverse caratterizzazioni grafiche.

Si riporta come esempio un estratto dell'introduzione dell'edizione del *Libro della natura degli animali* (Checchi 2020):

Nel testo critico si adottano infine i seguenti segni diacritici:

[lezione]	congetturale.
errore	imputabile al volgarizzatore o alla tradizione delle fonti latine.
†lezione†	incomprensibile

L'uso di segni diacritici si osserva anche nelle edizioni digitali, come, per esempio, la seconda edizione digitale della *Commedia* curata da Prue Shaw e pubblicata nel 2021.⁷² Nella parte dell'interfaccia in cui vengono visualizzate i testi editi dei manoscritti è presente una legenda, che esplicita

⁷⁰ Uno dei segni più utilizzati è l'obelò (†), detto anche *crux interpretum* o *crux desperationis*.

⁷¹ Cfr. Leonardi 2022, p. 168: "I criteri di trascrizione e di normalizzazione grafica suggeriscono spesso anche gli accorgimenti tipografici da utilizzare per rendere visibili gli interventi dell'editore sul testo critico, ma anche su questa tipologia di segni diacritici editoriali non c'è una norma unitaria. [...] Uno schema indicativo potrebbe essere il seguente: [] per le congetture dell'editore; [...] per le lacune; < > per le espunzioni dei copisti; <...> per le rasure dei copisti; † per i passi incomprensibili; corsivo per le porzioni di testo che non sono trasmesse dal manoscritto di superficie. Ma la casistica varia di edizione in edizione, ed è in genere esplicitata nell'introduzione".

⁷² <https://dantecommedia.it/>.

il significato dei segni diacritici e delle distinzioni per colore utilizzate. Per esempio: “[...] Unreadable text”, “\xx/ Interlinear insertion”.

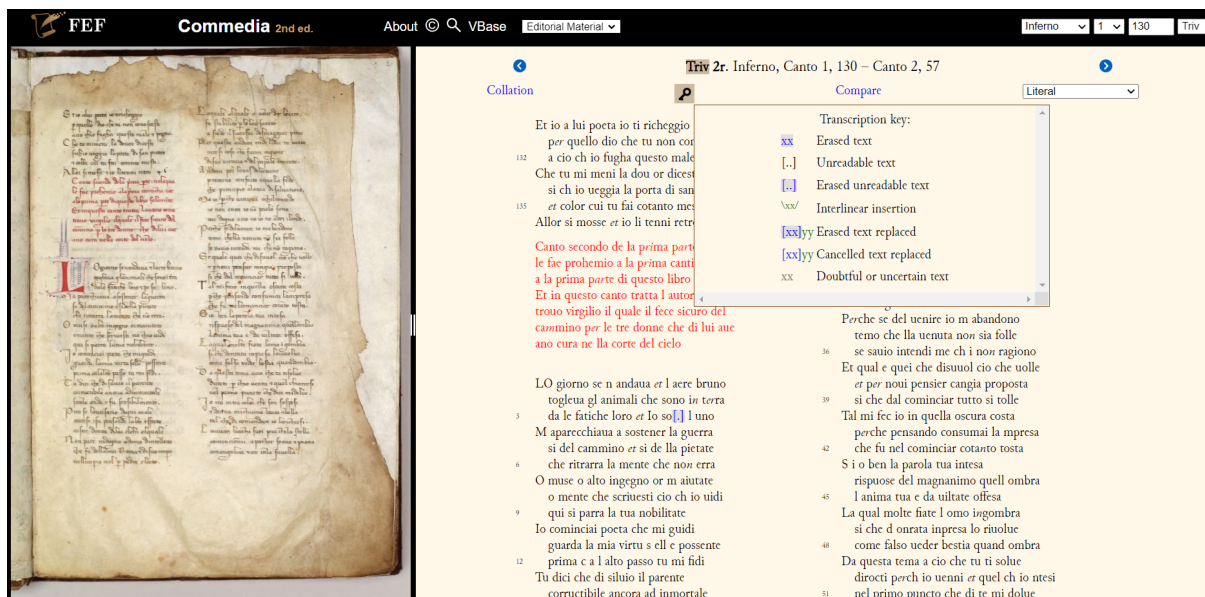


Figura 3.30 Schermata della seconda edizione critica digitale della *Commedia* curata da Prue Shaw.

Nell'ontologia Scholarly Editing è presente la classe `se:Editorial Symbol`, così definita: “symbol in a scholarly edition indicating a textual property of the original text.”⁷³ Nella presente ontologia una classe analoga a `se:Editorial Symbol` risulta essere superflua, in quanto il segno diacritico non è altro che una rappresentazione formale e abbreviata dei concetti espressi da altre classi, in particolare `Critical Intervention Type`, `Reading In Apparatus Type` (vd. sezione 3.6.6), `GapUnclear Passage`. Il segno diacritico è trattato, dunque, come una sequenza di caratteri che rientra nel contenuto delle classi `Critical Text Passage` o `Reading In Apparatus`. Il buco presente nel manoscritto alla Figura 3.19 e la rispettiva resa grafica della lacuna nel testo critico attraverso i caratteri “[...]”, potrebbero essere formalizzati come nella figura sotto.

⁷³ <https://github.com/nie-ine/e-editions/blob/master/ontology/scholarly-editing.ttl>.

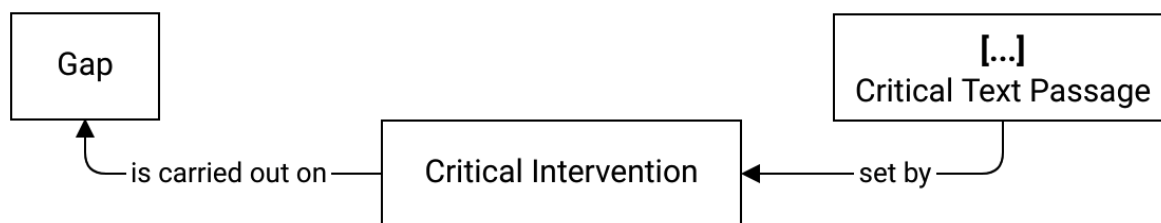


Figura 3.31 Esempio di come formalizzare una lacuna e la sua resa grafica nel testo critico.

Si propone un ulteriore esempio di omissione sanata per congettura, tratto dall'edizione critica del poemetto *Pacion de Santa Margarita* (Lannutti 2012), che è attestato in un solo manoscritto. Al verso 1355, la congettura è presentata tra parentesi quadre e in corsivo.

Anquaras mays [*mandta*] Jhesus

1355. *om.* mandta

La presenza dell'omissione deve essere stata evinta sia dall'analisi del senso del verso sia dall'individuazione di un'ipometria. Nella relativa nota di commento filologico viene proposta anche la possibile causa dell'omissione:

“1355. *mandta*: come al v. 1345. È possibile che l'omissione sia dovuta alla presenza di *mays*, che ha identico inizio.”

L'esempio sopra riportato può essere formalizzato mediante il modello come in figura:

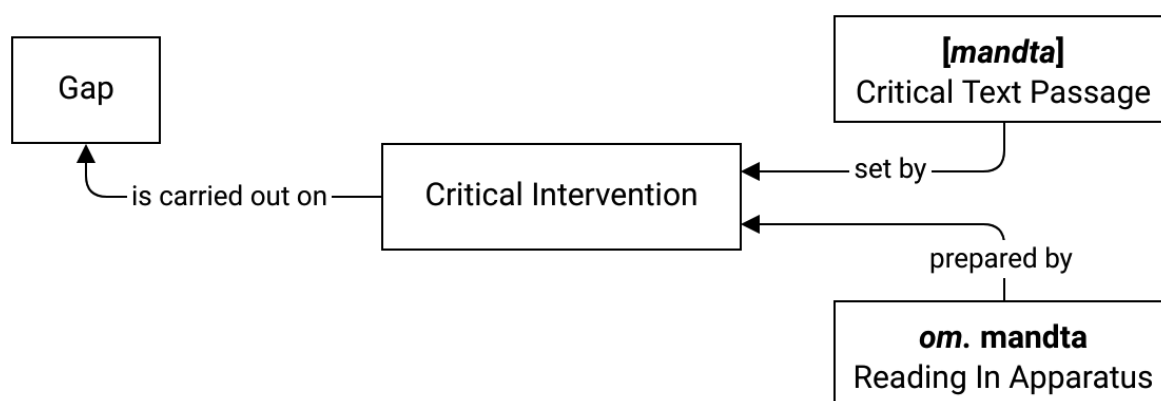


Figura 3.32 Esempio di come si formalizza un'omissione sanata per congettura.

3.6.6 La descrizione delle lezioni in apparato

Nella fase di analisi e interpretazione dei testimoni, l'editore valuta e classifica ciascuna lezione, *in primis* in base al modo in cui si presenta all'interno del proprio supporto fisico, per cui una lezione

può essere classificata come una cancellatura, un'aggiunta, una sostituzione, ecc. L'editore può stabilire se una lezione è un errore o un'innovazione, ovvero se non è ascrivibile all'originale, valutando la coerenza⁷⁴ della lezione rispetto al suo contesto da un punto di vista linguistico, semantico e, nel caso di componimenti poetici, anche da un punto di vista metrico. L'editore può servirsi di ulteriori parametri per valutare la qualità della lezione, tra cui l'*usus scribendi* dell'autore oppure facendo riferimento al contesto culturale in cui l'opera è stata realizzata, per esempio prendendo in considerazione i testi che l'autore potrebbe aver usato come fonti. Quando il testo è tramandato da più testimoni, l'editore classifica le lezioni anche sulla base dell'esito della collazione e dell'autorevolezza e posizione che ciascun testimone ricopre all'interno dello stemma.

Le diverse caratteristiche delle lezioni che emergono in fase di analisi guidano l'editore nell'individuazione delle lezioni non ascrivibili all'originale e per questo motivo l'editore si premura di segnalare le caratteristiche riscontrate al lettore all'interno dell'edizione critica, in particolare nell'apparato critico.

Come anticipato, il modello CAO permette di descrivere in modo dettagliato le caratteristiche delle lezioni all'interno dell'apparato critico. Si presentano qui le classi e le proprietà della CAO che sono state inserite nel modello, al fine di essere utilizzate unitamente alla classe `Reading In Apparatus`.

Le classi `cao:Reading Type`⁷⁵ e `cao:Reading Cause`⁷⁶ permettono di rappresentare rispettivamente l'etichetta (o più etichette) con cui l'editore ha classificato la lezione (per esempio: "variante formale" e "aggiunta") e la causa che, secondo l'editore, ha indotto il copista nell'introdurre un'innovazione (per esempio: "saut du même au même").⁷⁷ Le due classi sono state modellate rispettivamente sugli attributi `@type` e `@cause` dell'elemento `<rdg>`, che nel vocabolario TEI rappresenta la lezione all'interno di una voce dell'apparato critico, che possono essere associati agli elementi che rappresentano la lezione (`<rdg>`). Per una questione di chiarezza terminologica, nella presente

⁷⁴ Si rimanda al paragrafo dedicato alla "critica interna" in Contini 2014 (pp. 28 - 29).

⁷⁵ "This class comprises reading types. A reading type is the category for a variant reading (e.g. omission, conjecture, addition)" (<https://fgiovannetti.github.io/cao/#ReadingType>).

⁷⁶ "This class comprises reading causes. A reading cause is the cause for a variant reading (e.g. homeoteleuton, dittography, interpolation)" (<https://fgiovannetti.github.io/cao/#ReadingCause>).

⁷⁷ "The accidental skipping of some text in between two similar words or phrases; a particular type of haplography." (<https://fgiovannetti.github.io/cao/#saut-du-meme-au-meme>).

ontologia si propongono due classi native `Reading In Apparatus Type`, equivalente a `cao:Reading Type`, e `Reading In Apparatus Cause`, equivalente a `Reading Cause`. La classe `Reading In Apparatus` è collegata alla classe `Reading In Apparatus Type` mediante la proprietà `cao:has reading type` e alla classe `Reading In Apparatus Cause` grazie alla proprietà `cao:has reading cause`.

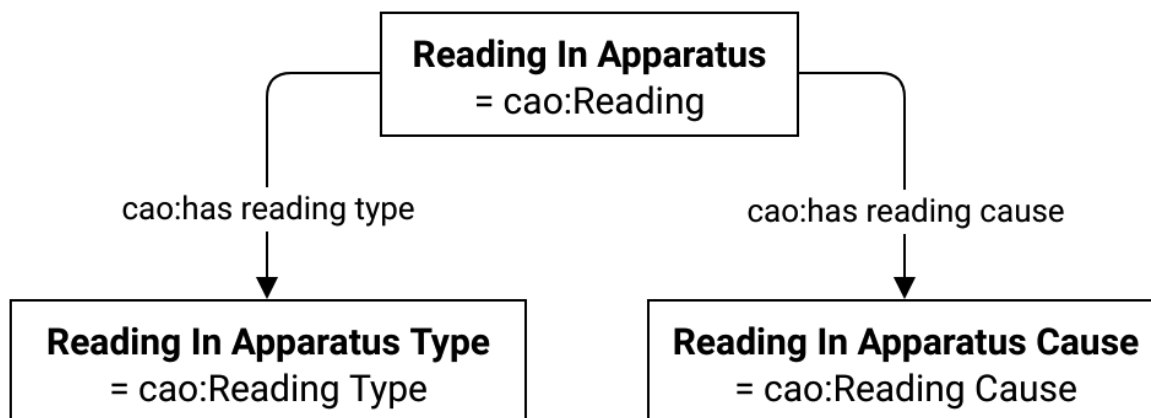


Figura 3.33 Schema di `Reading In Apparatus` e delle classi che ne rappresentano il tipo e la causa.

Per mostrare come queste classi possono essere utilizzate, si offre un esempio tratto dall'edizione critica di due poemetti in lingua d'oc del XIII secolo, *Vita e passione di Santa Margherita d'Antiochia*, curata da Maria Sofia Lannutti e pubblicata nel 2012. Nella voce d'apparato associata al verso 28 del primo poemetto, *Vida de Santa Margarita* (Lannutti 2012, p. 4), l'editrice descrive il modo in cui una porzione del verso si presenta in uno dei manoscritti:

moric sa mayre] in P scritto al di sopra di ~~no stardet~~ già al v. 27

L'esempio di sostituzione sopra illustrato può essere formalizzato come in figura.

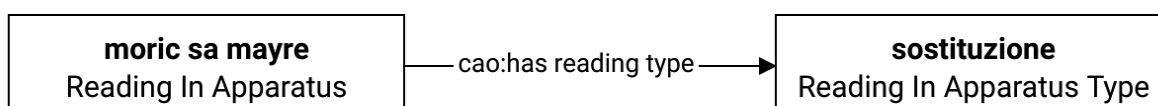


Figura 3.34 Esempio di istanza di `Reading In Apparatus` associata a un'istanza di `Reading In Apparatus Type`.

Nell'ontologia CAO sono proposti anche dei *named individual*⁷⁸ per le classi `cao:Reading Type` e `cao:Reading Cause`. I *named individual* disponibili per la classe `cao:Reading Cause` sono stati proposti in modo arbitrario dalla sua ideatrice:

“Some examples of values are:

- dittography (the unintentional repetition of a letter, word or phrase);
- haplography (the accidental omission of a letter, word or phrase; homeoarchy, homeoteleuton, saut du même au même are particular types of haplography);
- homeoarchy (the accidental skipping of a word or phrase having the same beginning; a polyptoton is a particular type of homeoarchy);
- polyptoton (the accidental skipping of a word or phrase forming a polyptoton, i.e. presenting the same root word);
- homeoteleuton (the accidental skipping of a word or phrase having the same ending);
- saut du même au même (the accidental skipping of some text in between two similar words or phrases);
- incorporation (the accidental incorporation of materials, such as marginalia)” (CAO).

I valori proposti risultano adatti agli scopi del presente modello, per cui sono stati inclusi anche nella presente ontologia, come *named individual* della classe `Reading In Apparatus Cause`.⁷⁹

I *named individual* della classe `cao:Reading Type`, invece, sono stati modellati in base alla classificazione delle lezioni proposta nell'ontologia LAWD (Linked Ancient World Data),⁸⁰ e nello specifico sono: aggiunta, cancellatura, congettura, correzione, omissione e trasposizione. Nel PLS le

⁷⁸ Nel contesto di un'ontologia con il termine *individual* si indica un'istanza di una classe, che corrisponde a un oggetto o entità del dominio reale e che ha associato un identificativo, per essere riferibile dall'esterno anche da altre ontologie, che può essere concreto, come per esempio un manoscritto, una persona, un animale, ecc., oppure astratto come un numero o una lettera. In altre parole, un *individual* corrisponde a un'istanza di una classe del dominio. Per esempio Con *named individual* si indica un *individual* cui viene dato un nome esplicito, così che possa essere riferito anche all'interno di altre ontologie.

⁷⁹ I valori proposti dalla CAO per la classe `cao:Reading Cause`, nel PLS vengono classificati in parte come tipologia di errore e in parte come causa dell'errore: “More specific terms are often used to **describe particular types of variants or errors** within these categories. Examples include: dittography (addition); haplography (omission); metathesis, anasyllabism, transposition; misreading, itacism (substitution). Because errors frequently arise due to misreading on the part of the copyist – that is an error in language processing that affects language production – textual scholars have also developed an elaborate vocabulary for **the triggers or precipitators for errors** (cf. copying of texts). Examples are anticipation, arrhythmia, homoeoarcton, homoeoteleuton, parablepsis, saut du même au même” (voce “types of error”, <https://wiki.helsinki.fi/display/stemmatology/Error%2C+types+of>, mia enfasi).

⁸⁰ L'ontologia LAWD, sviluppata nel contesto delle iniziative promosse dal programma Linked Ancient World Data Institute (<https://www.neh.gov/divisions/odh/institutes/linked-ancient-world-data-institute>), è stata ideata per colmare la distanza tra i dati utilizzati e pubblicati nei progetti che si concentrano sul mondo antico e i vocabolari standard dei Linked Data, come Dublin Core, Open Annotation Collaboration e CIDOC CRM.

principali categorie di errore sono, invece, aggiunta, omissione, trasposizione e sostituzione.⁸¹ Nel presente modello vengono accolti i valori utilizzati nella CAO, ma in una futura revisione del modello concettuale sarà necessario rivalutare le diverse proposte di classificazione degli errori. Inoltre, si segnala che i valori “omissione” e “congettura” devono essere utilizzati per rappresentare esclusivamente lezioni offerte in apparato che sono state classificate come omissione o come congettura. L’omissione in sé è rappresentata dalla classe *Gap*, presentata alla sezione 3.6.4.1. Per quanto riguarda le lezioni congetturali, le classi *Reading In Apparatus* e *Reading In Apparatus Type* devono essere utilizzate esclusivamente per rappresentare i casi di lezioni congetturate da altri editori esterni all’edizione, dal copista oppure dall’editore dell’edizione, senza però essere incluse nel testo critico. Mentre per le lezioni congetturali inserite nel testo critico, come nell’esempio di “[*mandta*]” riportato alla sezione 3.6.5.1, da un punto di vista concettuale è più opportuno utilizzare le classi *Critical Intervention Type* e *Critical Text Passage*, nel modo illustrato alla sezione 3.6.5.

Per rappresentare la valutazione della qualità della lezione rispetto a dei “criteri interni” (Leonardi 2022) di valutazione, si propongono altri due *named individual* nativi: “errore” e “innovazione”. Si precisa che l’errore è un’innovazione, ma si mantengono distinti i due concetti, rimandando alle definizioni offerte in Leonardi 2022 (pp. 192-193), per rappresentare i casi in cui una lezione, pur risultando congrua rispetto alla “coerenza grammaticale, semantica, metrica e narrativa del testo”, viene identificata come un’innovazione.

Per rappresentare i diversi casi individuabili dall’editore rispetto ai “criteri esterni” di valutazione si aggiungono anche: variante, variante adiafora, variante sostanziale, variante formale, errore congiuntivo, errore separativo, errore monogenetico, errore poligenetico, errore facilmente correggibile, diffrazione in presenza, diffrazione in assenza, lezione singolare, lezione caratteristica, banalizzazione e *lectio difficilior*.

⁸¹ Cfr. voce “types of error” del PLS: “all errors can be classified in four general categories that also comprise variants: addition, omission, transposition, and substitution” (<https://wiki.helsinki.fi/display/stemmatology/Error%2C+types+of>).

L'elenco dei *named individual* associati alle classi `cao:Reading Type` e `cao:Reading Cause` può essere ulteriormente esteso, contestualmente a possibili future espansioni del presente modello.

Nel caso in cui l'editore desideri attribuire la lezione a una mano scribale,⁸² si ripropone la soluzione della CAO, ovvero è possibile rappresentare la mano mediante la classe `prov:Agent`⁸³ e collegarla a `Reading In Apparatus` grazie alla proprietà `cao:has attributed hand`.

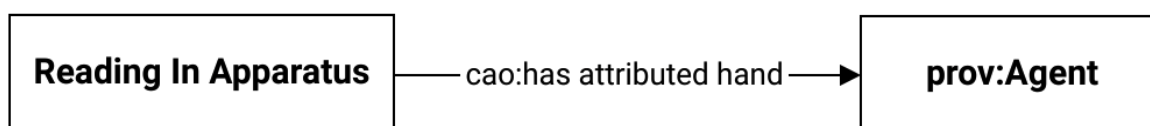


Figura 3.35 Schema del rapporto tra una lezione in apparato e la mano scribale cui è attribuita.

Se l'editore desidera rappresentare i rapporti di derivazione o altri tipi di legami tra le diverse lezioni presentate all'interno dell'apparato, il modello CAO mette a disposizione delle proprietà apposite, in parte native e in parte tratte dall'ontologia PROV-O:

- `cao:follows`,⁸⁴ modellata sull'attributo delle linee guida TEI `@varSeq`,⁸⁵ che consente di ordinare una sequenza di lezioni, in modo tale da esplicitare l'ordine temporale in cui sono apparse all'interno della tradizione;
- `cao:was corrected to`, “The property relates a corrected reading to the previous state before correction (also a reading)”;⁸⁶
- `prov:was revision of`, “A revision is a derivation for which the resulting entity is a revised version of some original. The implication here is that the resulting entity contains substantial content from the original. Revision is a particular case of derivation”;⁸⁷

⁸² Con “mano” si intende la persona che all'interno di un documento manoscritto ha copiato l'intero testo o porzioni di esso. Si utilizza il termine “mano”, in quanto spesso i copisti sono anonimi e non è possibile stabilire la loro identità.

⁸³ È stata vagliata la possibilità di creare una classe nativa `Copyist`, sottoclasse di `prov:Agent`. Considerando, però, che le mani che intervengono su un manoscritto non necessariamente sono quelle di un copista, si è preferita la soluzione della CAO.

⁸⁴ <https://fgiovannetti.github.io/cao/#follows>.

⁸⁵ https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-att.textCritical.html#tei_att.varSeq.

⁸⁶ <https://fgiovannetti.github.io/cao/#wasCorrectedTo>.

⁸⁷ <https://fgiovannetti.github.io/cao/#http://www.w3.org/ns/prov#wasRevisionOf>.

- `prov:was derived from`, che comprende la precedente proprietà in quanto è più generica: “A derivation is a transformation of an entity into another, an update of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity.”⁸⁸

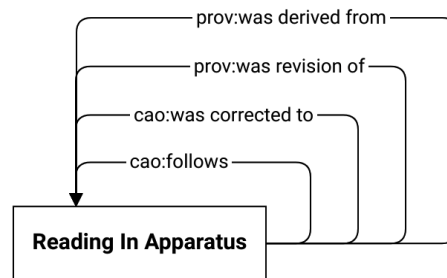


Figura 3.36 Schema delle proprietà riflessive di Reading In Apparatus.

Per mostrare come queste proprietà possono essere utilizzate, si riprende il caso di sostituzione tratto da Lannutti 2012 (sezione 3.6.6). Utilizzando la proprietà `cao:was corrected to`, le due lezioni attestate nel manoscritto P possono essere formalizzate come in figura.

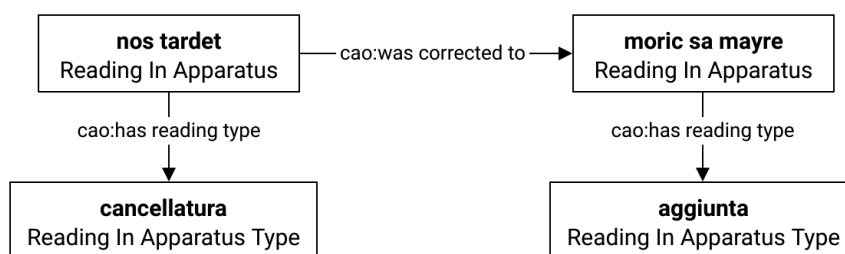


Figura 3.37 Esempio di come formalizzare il rapporto tra due lezioni in apparato correlate.

3.6.7 L'apparato critico

All'interno di un'edizione critica l'editore offre in calce al testo critico una serie di informazioni per rendere il lavoro svolto trasparente e per guidare il lettore nella comprensione del testo. Alcune di queste informazioni sono indispensabili, come le varianti sostanziali e formali. Altre, invece, vengono fornite se l'editore lo ritiene opportuno, tra cui ci sono:

- le note di commento, all'interno delle quali l'editore offre delle chiavi per interpretare il testo oppure illustra le motivazioni delle proprie scelte editoriali, in questo caso si tratta di note di commento filologico;

⁸⁸ <https://fgiovannetti.github.io/cao/#http://www.w3.org/ns/prov#wasDerivedFrom>.

- le fonti che, ad avviso dell'editore, sono state utilizzate dall'autore in fase di scrittura, con le eventuali citazioni e i riferimenti bibliografici;
- i passi paralleli, per guidare la comprensione del passo interessato a partire dall'interpretazione degli altri passi;
- le traduzioni o le parafrasi puntuali per alcuni luoghi del testo critico.

Per poter rappresentare a livello formale queste diverse tipologie di informazioni è stato necessario mettere a fuoco che cosa si intende con apparato critico e con fascia dell'apparato critico. Come spunto iniziale di riflessione si riportano due diverse definizioni.

In Leonardi 2022 il termine apparato è definito nel seguente modo: “settore dell'edizione in cui si registrano le differenze fra il testo critico e i manoscritti”.

In Chiesa 2002, invece, si legge:

“il testo critico che costituisce l'oggetto della pubblicazione è accompagnato da un **apparato critico** – di solito collocato al piede della pagina, più raramente al termine del testo critico – nel quale trovano posto le varianti che l'editore non ha ritenuto di accogliere, per ognuna delle quali vengono indicati i testimoni che la riportano. Ovviamente, a seconda degli scopi dell'edizione e della natura del testo pubblicato, **l'apparato può contenere materiali diversi**” (enfasi di chi scrive).

Con apparato critico si può intendere sia il solo apparato delle varianti, come nella definizione fornita da Leonardi, sia l'insieme di tutte le diverse tipologie di informazioni sopra elencate che vengono offerte insieme al testo critico, come nella definizione di Chiesa. Anche il concetto di fascia d'apparato è ambiguo, perché in generale indica delle suddivisioni all'interno dell'apparato critico, che nelle edizioni a stampa sono riconoscibili in quanto figurano a piè di pagina come fasce orizzontali di testo separate da spazi, da linee orizzontali o da altri segni grafici. A seconda che l'apparato critico corrisponda al solo apparato delle varianti o racchiuda diverse tipologie di informazioni, cambia anche il significato del termine fascia d'apparato. Nel primo caso è da intendersi come una suddivisione dell'apparato delle varianti in base a un determinato criterio, per esempio per distinguere le varianti formali da quelle sostanziali oppure per raggruppare le lezioni di diverse famiglie di testimoni. Nel secondo caso, la fascia d'apparato è una parte dell'apparato che raccoglie un particolare tipo di informazioni, come la fascia delle note di commento al testo, la fascia delle fonti, la fascia delle varianti, dei passi paralleli, ecc.

Di conseguenza, per formalizzare il concetto di apparato sono state vagliate due possibili soluzioni. La prima, basata sulla definizione di Leonardi, consiste nel creare una classe che rappresenti in generale il concetto di apparato. Dalla classe generica “Apparato” è possibile derivare delle classi che rappresentino i diversi tipi di apparato, in base all’obiettivo scientifico che soddisfano, per esempio la classe “Apparato critico” (inteso come apparato delle varianti), “Commento”, “Passi paralleli”, “Apparato delle fonti”, ecc. Infine la classe “Fascia d’apparato” rappresenta una suddivisione del solo “Apparato critico”.

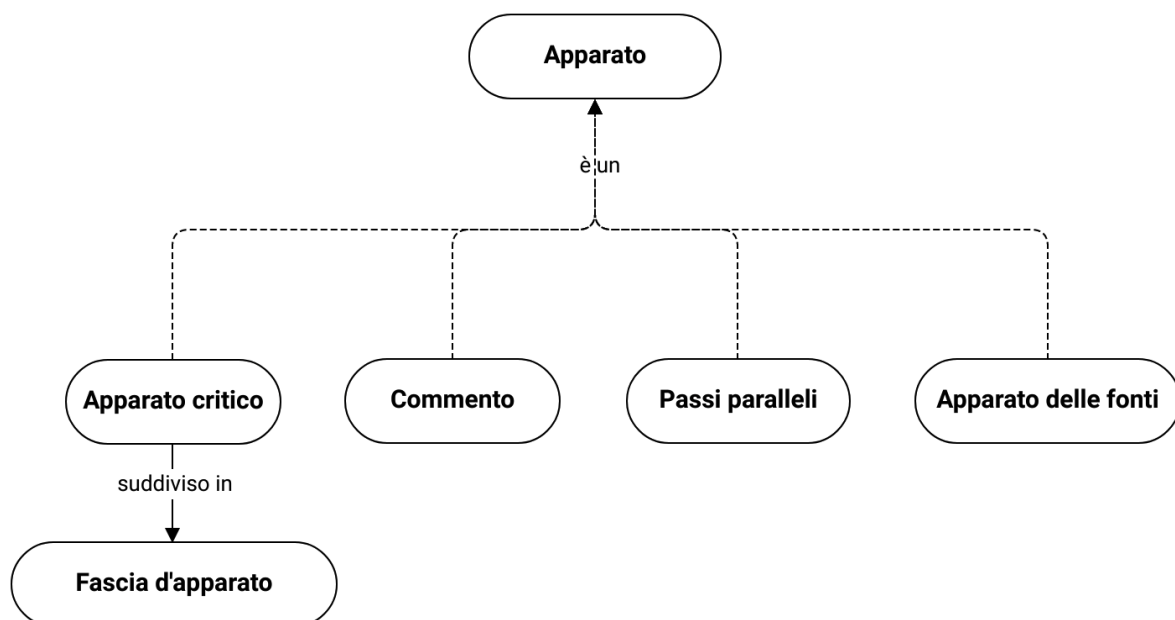


Figura 3.38 Schema della prima soluzione per rappresentare l'apparato critico.

La seconda soluzione è basata, invece, sulla definizione di Chiesa e prevede una classe “Apparato critico” che rappresenti l’insieme di tutte le annotazioni apposte al testo critico, una classe “Fascia d’apparato” che rappresenti le diverse suddivisioni interne all’apparato e che abbia una serie di sottoclassi per distinguere le tipologie di informazioni.

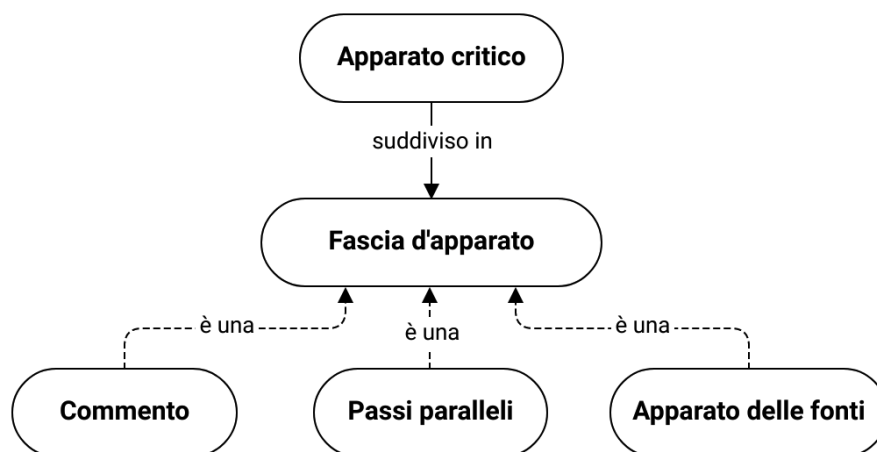


Figura 3.39 Schema della seconda soluzione per rappresentare l'apparato critico.

Nelle edizioni a stampa il distinguo tra “fascia come apparato a sé” (seconda soluzione) e “fascia come ulteriore suddivisione dell’apparato critico” (prima soluzione) è molto spesso sfumato. Nelle edizioni marcate in XML/TEI, invece, avendo la possibilità di strutturare l’edizione attraverso dei *tag* “parlanti”, che hanno un significato preciso, si tende maggiormente a distinguere tra i diversi tipi di informazione. Anche se l’elemento <app>, che rappresenta la voce d’apparato critico, consente di inserire al suo interno uno o più elementi <note>, per apporre delle note di diverso tipo al punto del testo in cui l’editore registra la variazione testuale.⁸⁹

Nel contesto del presente modello la prima soluzione è sembrata essere la più idonea, considerato anche che ricalca la soluzione adottata nell’ontologia SE. Secondo tale soluzione, le diverse tipologie di apparato sono rappresentate da altrettante classi, tra cui `se:ApparatusFontium` e `se:CriticalApparatus`, tutte definite come sottoclassi di `se:Apparatus`. Quest’ultima classe è descritta nel seguente modo:

“Editorial compositional structure in a scholarly edition **to analyse text along certain criteria**”
(mia enfasi).

Le voci di ciascun apparato sono sottoclassi di `se:ApparatusEntry`, la classe che rappresenta in maniera generica una voce d’apparato. La soluzione dell’ontologia SE è molto flessibile, in quanto

⁸⁹ Nella TEI la voce dell’apparato critico è rappresentata dall’elemento <app> (<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-app.html>), che può essere inserito all’interno del testo critico oppure presentato in un altro punto del documento all’interno dell’elemento <listApp> (<https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-listApp.html>). Una nota di commento è rappresentata dall’elemento <note>, che in modo analogo può essere inserito nel testo critico o in un altro punto del documento.

permette di rappresentare sia i casi in cui l'editore fornisce diverse tipologie di apparato distinte in modo netto tra di loro, sia i casi in cui l'apparato è uno solo e racchiude informazioni diverse. Nel primo caso si possono utilizzare le sottoclassi *ad hoc*, nel secondo caso direttamente `se:Apparatus` e `se:Apparatus Entry`.

Dato che l'ontologia SE non è più mantenuta, nella presente ontologia le classi `Apparatus` e `Apparatus Entry` sono state riproposte come classi native. La classe `Apparatus Entry` è collegata ad `Apparatus` mediante la proprietà `is entry of (has entry)`. Entrambe le classi, `Apparatus` e `Apparatus Entry` sono sottoclassi di `Accompanying Material`.

Per il momento, nella presente ontologia, sono state definite solo due sottoclassi di `Apparatus` e due di `Apparatus Entry`, per rappresentare l'apparato critico, inteso come apparato delle varianti (`Critical Apparatus`) e la voce dell'apparato critico (`Critical Apparatus Entry`) e il commento filologico (`Commentary`) e la singola nota di commento (`Note`). Nelle future espansioni del modello concettuale, potranno essere aggiunte ulteriori sottoclassi. Per collegare l'apparato critico a una sua voce si utilizza la proprietà `critical apparatus has entry (is entry of critical apparatus)`. Il commento filologico è legato a una sua nota dalla proprietà `commentary has note (is note of commentary)`. Entrambe le proprietà sono sottoproprietà di `has entry (is entry of)` che a sua volta può essere descritta come una sottoproprietà di `lrmo0:R5 has component (is component of)`.

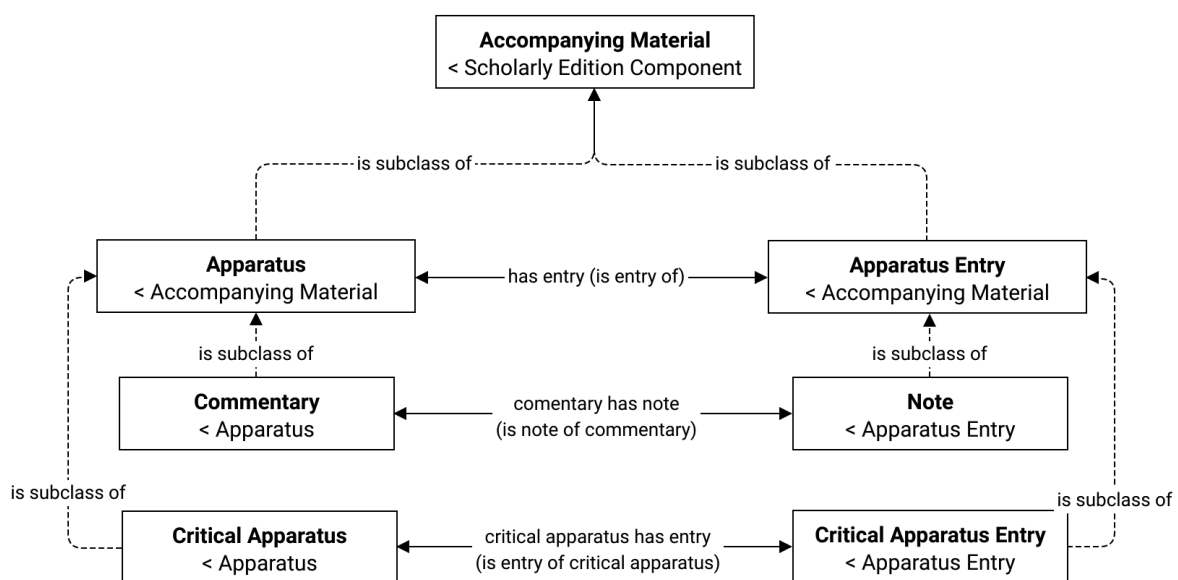


Figura 3.40 Schema delle diverse tipologie di apparato e delle rispettive voci.

Nella CAO l'apparato critico è definito come un insieme di annotazioni, mentre nell'ontologia SE "editorial compositional structure" (letteralmente 'struttura compositiva editoriale'). Entrambe le definizioni sono valide. Un apparato è un'unità logica, l'insieme delle annotazioni che l'editore appone al testo critico per un particolare obiettivo scientifico. Al contempo, l'apparato è un componente grafico dell'edizione. Tendenzialmente, in ambiente digitale, l'apparato non viene presentato come un componente grafico unitario separato dal testo critico, ma le singole voci dell'apparato vengono presentate all'interno del testo critico o vicino ad esso sotto forma di *pop-up*, pannelli laterali, finestre temporanee e simili.⁹⁰ Ci sono casi di edizioni scientifiche digitali in cui l'apparato costituisce un elemento grafico a sé dell'interfaccia, come per esempio in quelle che utilizzano il software di visualizzazione EVT (la seconda versione).

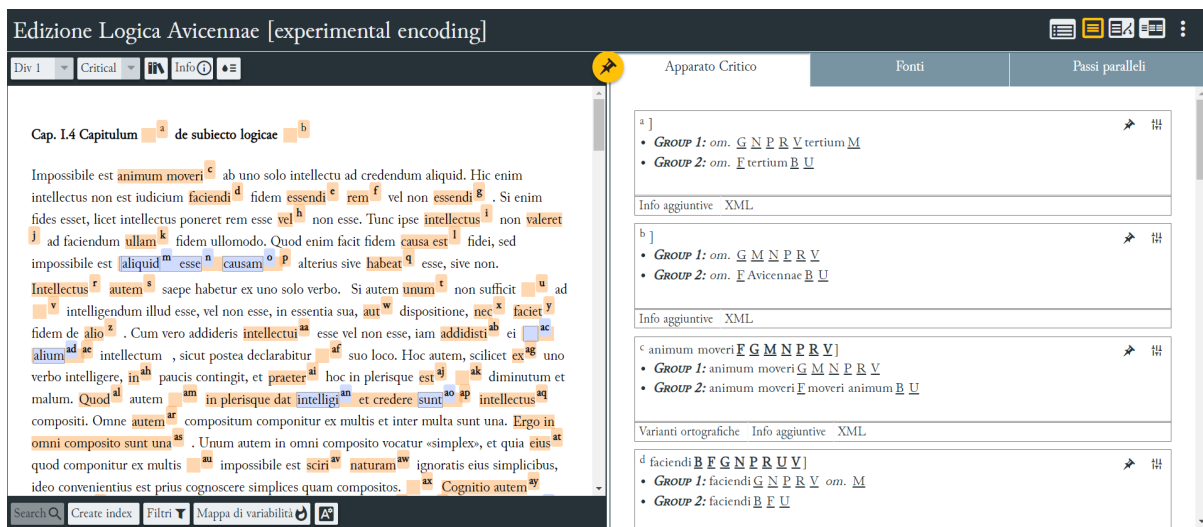


Figura 3.41 Schermata dell'interfaccia del software EVT 2.

Il presente modello concettuale permette di rappresentare formalmente entrambi i casi. Quando l'apparato non compare come elemento grafico all'interno dell'edizione, è sufficiente la classe *Apparatus Entry*, che è infatti sottoclasse diretta di *Accompanying Material*.

3.6.7.1 Il collegamento tra l'apparato e il testo edito

Un apparato è un componente che può rientrare in diverse tipologie di edizioni scientifiche. Affinché la classe *Apparatus* possa essere riutilizzata in modelli concettuali diversi, si propone di collegarla alla classe *Edited Text*, che rappresenta il testo edito. L'apparato critico e il commento

⁹⁰ Per un maggior numero di esempi di come viene presentato l'apparato critico all'interno di un'edizione critica digitale, si rimanda alla sezione 3.3 in Martignano 2017.

filologico, invece, sono due componenti tipici di un'edizione critica. Per questo motivo, `Critical Apparatus` e `Commentary` sono collegate alla classe `Critical Text`. L'apparato critico e il commento filologico potrebbero essere concetti utili anche nella definizione dei modelli concettuali di altre tipologie di edizioni, ma è preferibile non vincolare il riuso della classe `Edited Text`.

Per valutare come collegare i diversi apparati al testo edito o al testo critico, sono state nuovamente prese a riferimento le ontologie SE e CAO.

Nell'ontologia SE l'apparato, rappresentato dalla classe `se:Apparatus`, è collegato alla classe `se:Edition` mediante la proprietà `se:is_apparatus_of`. Questa relazione è utile nei casi in cui un'edizione contenga al proprio interno più testi allestiti dall'editore poiché permette di specificare a quale testo edito sia collegato l'apparato. Per questo motivo, si propone la proprietà denominata `is_apparatus_of (has_apparatus)`, sottoproprietà di `accompanies (is_accompanied_by)`, che collega le classi `Apparatus` ed `Edited Text`.

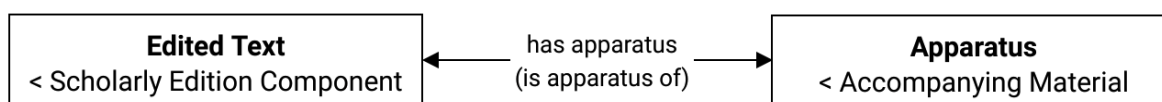


Figura 3.42 Schema del rapporto tra le classi Edited Text e Apparatus.

L'ontologia SE non offre delle proprietà per descrivere il rapporto tra il testo edito e le singole voci d'apparato, ma per completezza occorrono entrambe le prospettive, quella generica della proprietà `has apparatus (is apparatus of)` e quella puntuale per collegare la voce d'apparato al punto del testo edito cui è riferita.

Nella CAO una voce dell'apparato critico è collegata a un'istanza della classe `cao:Variation Location`, che rappresenta un punto del testo critico soggetto a variazione nella tradizione testuale,⁹¹ mediante la proprietà `oa:has_target`. La voce dell'apparato critico, infatti, nella CAO è rappresentata dalla classe `oa:Annotation` della Web Annotation Ontology (vd. 2.5.5.8):

“The creation of a new CAO scholarly annotation (or critical apparatus entry) is done through the Web Annotation Ontology. Each scholarly annotation has a body (the contents of the apparatus

⁹¹ “A location of variation is the locus in a collated text where textual variation is recognized to occur.” (CAO). <https://fgiovannetti.github.io/cao/#VariationLocation>.

entry) and a target (the location of the variation in the base text). Web Annotation allows the declaration of a date of creation and of a creator, making it possible to attach provenance information to each scholarly annotation” (CAO).

La classe `cao:Variation Location` è il *target* della voce dell’apparato critico, mentre il *body* è la classe `cao:Variation Unit`, modellata sull’elemento `<app>` delle linee guida TEI e definita:

“A unit of variation is a set of alternative readings that refer to the same locus in a collated text. A unit of variation features a maximum of [one] base reading and a minimum of [two] readings.”⁹²

La proprietà `oa:has body` unisce la classe `cao:Variation Unit` alla classe `oa:Annotation`. Infine, nella CAO è previsto che ad ogni voce dell’apparato critico sia possibile associare un’istanza della classe `prov:Agent`, così da poter specificare la persona responsabile per l’annotazione e anche la data di creazione dell’annotazione, come si vede in figura.

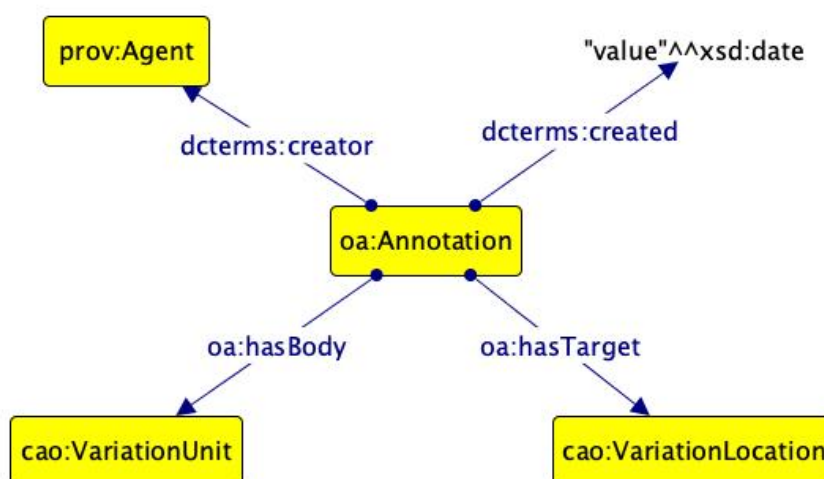


Figura 3.43 Schema del rapporto tra l’annotazione, l’unità di variazione e il luogo di variazione nell’ontologia CAO (CAO).

Come già accennato alla sezione 3.6.7, un’annotazione web è un collegamento, rappresentabile come un grafo orientato, che mette in relazione due diverse risorse web (entrambe con un IRI⁹³ associato): il contenuto dell’annotazione (o corpo, *body*, dell’annotazione) e l’elemento cui l’annotazione è riferita (*target*):

“An annotation is considered to be a set of connected resources, typically including a body and target, and conveys that the body is related to the target. The exact nature of this relationship

⁹² <https://fgiovannetti.github.io/cao/#VariationUnit>.

⁹³ IRI sta per Internationalized Resource Identifier, ed è un’estensione della specifica URI (Uniform Resource Identifier) che permette di utilizzare i caratteri Unicode, mentre URI richiede l’utilizzo esclusivo di caratteri ASCII.

changes according to the intention of the annotation, but the body is most frequently somehow “about” the target.” (<https://www.w3.org/TR/annotation-model/>, enfasi di chi scrive)

Il concetto di annotazione in sé si presta a descrivere tutte le voci di tutti gli apparati di un’edizione scientifica. Anche una voce d’apparato è costituita da un corpo, solitamente un testo di dimensione variabile, e dal collegamento al luogo nel testo edito cui è riferita. La classe `oa:Annotation` è specifica per la rappresentazione delle annotazioni web, che uniscono due informazioni collocate in due risorse web separate. Nel caso di un’edizione scientifica digitale, invece, le voci dell’apparato sono solitamente collocate all’interno della medesima risorsa in cui è situato il testo edito. L’ambiente digitale consente più facilmente il riuso e l’integrazione all’interno di un’edizione scientifica digitale delle informazioni poste in altre risorse web. Anche se questa pratica, ad avviso di chi scrive, non si è ancora consolidata nel campo della filologia digitale, può essere utile predisporre il modello in maniera tale da sfruttare appieno il potenziale di un’edizione scientifica digitale.⁹⁴ Inoltre, il *Web Annotation Data Model* prevede la descrizione di casi in cui il corpo dell’annotazione è un testo creato insieme all’annotazione:

“In many situations, the Body of the Annotation will be in a text format, and created at the same time as the Annotation without a separate IRI. In these cases, the Body’s text can be included as part of the Annotation to avoid having to interact with multiple systems. The Body may also have the features of External Web Resources, including especially the language of the text and the format that it is conveyed in.” (<https://www.w3.org/TR/annotation-model/#embedded-textual-body>)

La classe `se:Apparatus Entry` è definita, dunque, come equivalente della classe `oa:Annotation`.

Il *body* di una voce d’apparato è rappresentato dalla classe generica `Apparatus Entry Content`. Trattandosi di un contenuto di natura testuale, può essere equiparato alla classe `oa:Textual Body`.⁹⁵ La classe `Critical Apparatus Entry Content` è sottoclasse di `Apparatus`

⁹⁴ Un’altra considerazione a favore della separazione tra il testo annotato e la nota dell’editore su più documenti si trova in (Shillingsburg, 2014, p. 5): “A difficult but necessary principle for future development is that the digital representation of texts as transcriptions must not be mixed in the same files with the explanatory, analytical, and interpretive commentary. The reasons for separating these normally mixed elements need articulation, for the tradition of embedding markup in texts is strong. Intuitively we think of text and encoded enhancements as integrated, perhaps because when we invoke a database for images, transcriptions, and information about a text we want an appropriate integration in the rendered display on the screen. For several reasons, it is a mistake, however, to store these elements mixed together in a file. Very significant pitfalls can be avoided and advantages gained by maintaining archive content (images, transcriptions, and commentary) in separate storage units.”

⁹⁵ <https://www.w3.org/TR/annotation-vocab/#textualbody>.

Entry Content e rappresenta il contenuto di una voce dell'apparato critico. La proprietà `apparatus entry has content (is content of apparatus entry)` collega le classi `Apparatus Entry` e `Apparatus Entry Content`. Le classi `Critical Apparatus Entry` e `Critical Apparatus Entry Content`, invece, sono collegate da una sottoproprietà denominata `critical apparatus entry has content (is content of critical apparatus entry)`.

Nel presente modello concettuale, il *target* di una voce dell'apparato critico è una sequenza di caratteri o parole contenuta all'interno del testo critico, concetto che equivale alla classe `Critical Text Passage`, introdotta alla sezione 3.6.5. Nel caso di un apparato generico, invece, il *target* è una sequenza di caratteri o parole contenuta nel testo edito, rappresentata dalla classe nativa `Edited Text Passage`, definita come superclasse di `Critical Text Passage`. Per rappresentare il rapporto tra la voce dell'apparato e il testo, si può utilizzare sia la proprietà `oa:has target`, sia la proprietà `crm:P67 refers to (is referred to by)`. Nella presente ontologia, si propone una proprietà nativa, denominata `apparatus entry refers to (is referred to by apparatus entry)`, sottoproprietà di `crm:P67 refers to (is referred to by)`, in quanto `oa:has target` non ha una proprietà inversa. Nell'ambito del dominio descritto dal presente modello è importante che dalla voce dell'apparato si possa risalire al testo edito. Il collegamento tra una voce dell'apparato critico e `Critical Text Passage` è effettuato mediante `critical apparatus entry refers to (is referred to by critical apparatus entry)`, sottoproprietà di `apparatus entry refers to (is referred to by apparatus entry)`.

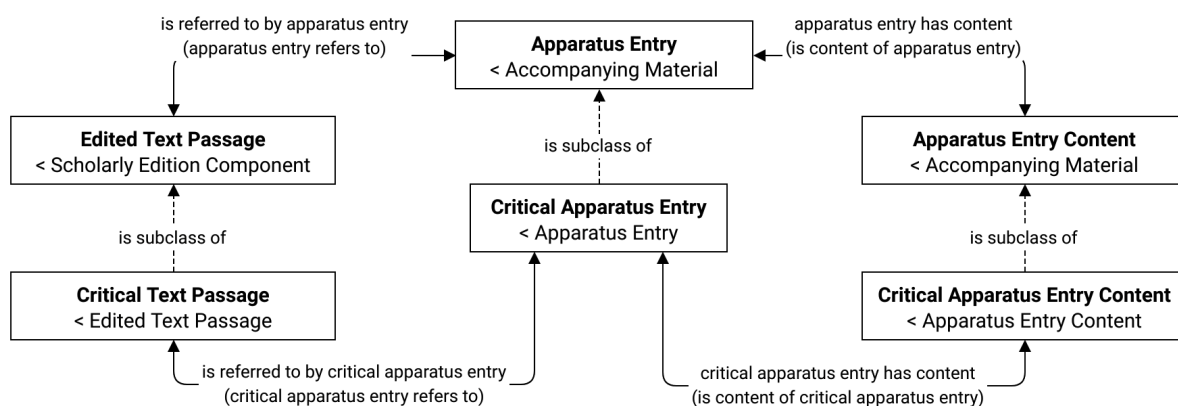


Figura 3.44 Schema delle classi utilizzate per rappresentare le voci d'apparato e le rispettive sottoclassi.

Il *Web Annotation Data Model* prevede che il punto preciso riferito dall'annotazione all'interno della risorsa web *target* sia individuato attraverso un selettore (*selector*):

“Many Annotations refer to part of a resource, rather than all of it, as the Target. We call that part of the resource a Segment (of Interest). A Selector is used to describe how to determine the Segment from within the Source resource. The nature of the Selector will be dependent on the type of resource, as the methods to describe Segments from various media-types will differ. Multiple Selectors can be given to describe the same Segment in different ways in order to maximize the chances that it will be discoverable later, and that the consuming user agent will be able to use at least one of the Selectors.” (<https://www.w3.org/TR/annotation-model/#selectors>)

Nella rispettiva ontologia il concetto di selettore è rappresentato dalla classe `oa:Selector`, che è superclasse di diversi tipi di selettori:

- `oa:Text Quote Selector`,⁹⁶ che descrive la porzione di testo copiandola;
- `oa:Text Position Selector`,⁹⁷ pensato per fornire il punto di inizio e di fine della porzione interessata sotto forma di indici numerici dei corrispondenti caratteri all'interno del documento web come *stream* (sequenza di dati);
- `oa:Svg Selector`,⁹⁸ utile per definire un'area all'interno di un'immagine.

Nella figura sotto si mostra come la classe `oa:Text Quote Selector` possa essere utilizzata.

⁹⁶ “The TextQuoteSelector describes a range of text by copying it, and including some of the text immediately before (a prefix) and after (a suffix) it to distinguish between multiple copies of the same sequence of characters.” (<https://www.w3.org/TR/annotation-vocab/#textquoteselector>).

⁹⁷ “The TextPositionSelector describes a range of text by recording the start and end positions of the selection in the stream. Position 0 would be immediately before the first character, position 1 would be immediately before the second character, and so on.” (<https://www.w3.org/TR/annotation-vocab/#textpositionselector>).

⁹⁸ “An SvgSelector defines an area through the use of the Scalable Vector Graphics (SVG) standard. This allows the user to select a non-rectangular area of the content, such as a circle or polygon by describing the region using SVG. The SVG may be either embedded within the Annotation or referenced as an External Resource.” (<https://www.w3.org/TR/annotation-vocab/#svgselector>).

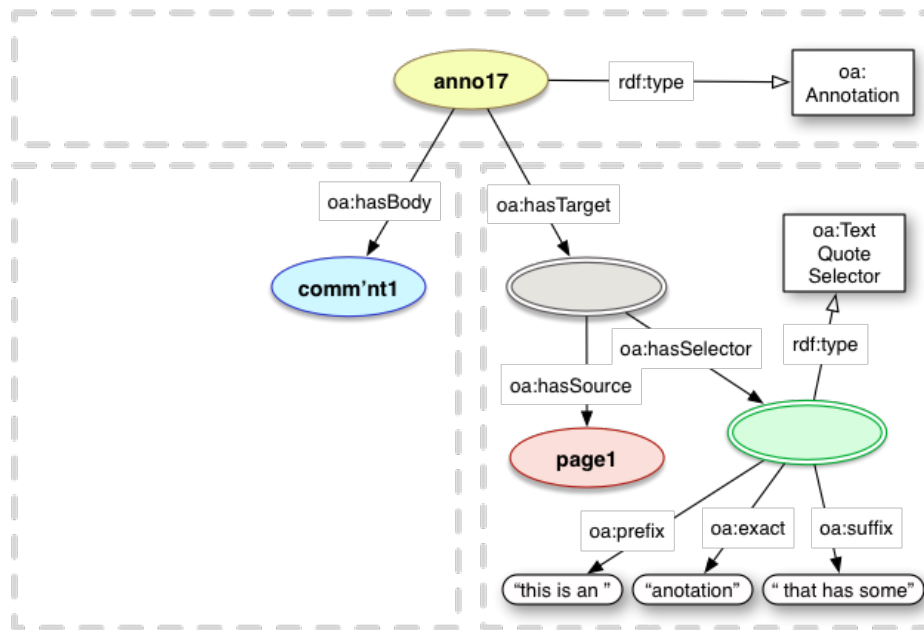


Figura 3.45 Schema che mostra come utilizzare la classe *oa:Text Quote Selector* (Web Annotation Vocabulary).

Nel presente modello, si propone la classe nativa *Passage Selector*, equivalente a *oa:Selector*, e collegata alla classe *Edited Text Passage* e *Critical Text Passage* mediante la proprietà *passage has selector* (*is selector of passage*).

Per inquadrare in termini più ampi che cos'è un selettore, si potrebbe far riferimento alla classe *crm:E41 Appellation*:

“This class comprises signs, either meaningful or not, or arrangements of signs following a specific syntax, that are used or can be used to refer to and identify a specific instance of some class or category within a certain context. Instances of *E41 Appellation* do not identify things by their meaning, even if they happen to have one, but instead by convention, tradition, or agreement. Instances of *E41 Appellation* are cultural constructs; as such, they have a context, a history, and a use in time and space by some group of users. A given instance of *E41 Appellation* can have alternative forms, i.e., other instances of *E41 Appellation* that are always regarded as equivalent independent from the thing it denotes.” (CIDOC CRM v. 7.1.2, p. 84)⁹⁹

Mentre la proprietà *has selector* (*is selector of passage*) può essere considerata come un particolare tipo della relazione espressa da *crm:P1 is identified by* (*identifies*):

⁹⁹ http://cidoc-crm.org/cidoc-crm/7.1.2/E41_Appellation.

“This property describes the naming or identification of any real-world item by a name or any other identifier.” (CIDOC CRM v. 7.1.2, p. 115)¹⁰⁰

Per determinare come l’edizione scientifica debba essere presentata nell’interfaccia, può essere utile descrivere formalmente come l’editore esplicita il collegamento tra la voce d’apparato e la porzione del testo edito o critico.

In un’edizione scientifica a stampa, il collegamento tra la voce d’apparato e il testo edito può essere esplicitato in modi diversi, spesso combinati tra di loro, ponendo prima della voce d’apparato:

- il numero della riga o del verso del testo edito in cui è contenuta la porzione di testo interessata;
- la copia, integrale o abbreviata, della porzione di testo interessata;
- una copia dell’esponente numerico o alfabetico con cui il collegamento a una voce d’apparato è segnalato nel testo edito subito dopo la porzione di testo interessata.

In un’edizione scientifica digitale, le modalità sopra elencate sono spesso riproposte all’interno dell’interfaccia grafica, anche se un’altra modalità molto diffusa in ambiente digitale consiste nel rendere interattiva la porzione del testo edito collegata a una voce dell’apparato, così che al *click* o al passaggio del cursore sulla porzione, venga mostrata la voce d’apparato direttamente all’interno o accanto al testo edito. A livello di codifica dei testi, il collegamento può essere sia puntuale, indicando in modo preciso l’inizio e la fine della porzione di testo coinvolta, sia più sommaria, segnalando l’elemento in cui la porzione è inserita, per esempio il verso, il paragrafo, ecc.

Nel presente modello si propongono due classi native, `Passage Quote Selector` e `Passage Position Selector`, ricalcate in parte su `oa:Text Quote Selector` e `oa:Text Position Selector`, sottoclassi di `Passage Selector`. La prima indica la citazione integrale o abbreviata della porzione interessata, la seconda invece un riferimento generico alla posizione della porzione nel testo, mediante la ripetizione dell’esponente o del numero dell’elemento in cui la por-

¹⁰⁰ http://cidoc-crm.org/cidoc-crm/7.1.2/P1_is_identified_by.

zione compare (paragrafo, riga, verso, ecc.). Se il selettore viene presentato all'interno della voce d'apparato, diventando un contenuto del corpo dell'annotazione, si può formalizzare questa relazione mediante la proprietà `selector is shown in`.

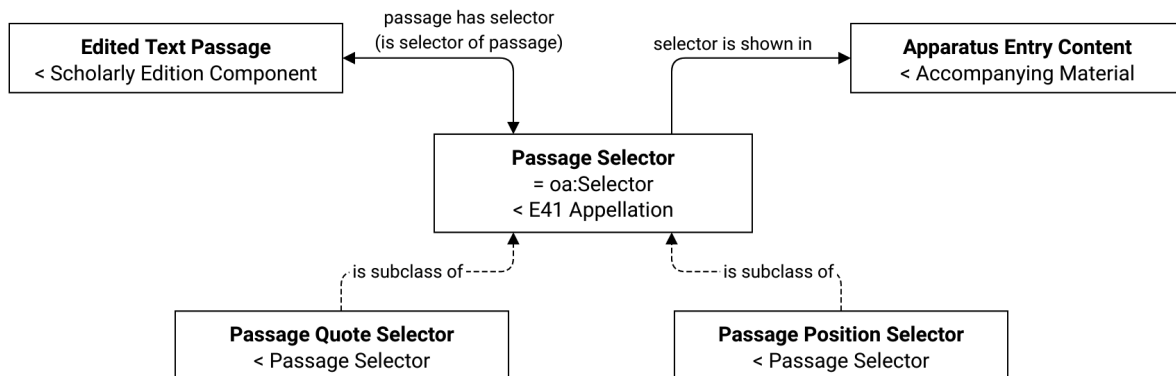


Figura 3.46 Schema che rappresenta le diverse tipologie di selettori di un luogo del testo edito e il rapporto con il contenuto di una voce d'apparato.

3.6.7.2 Il commento filologico

Come illustrato nella presente sezione, il commento filologico può essere descritto come un particolare tipo di apparato, concetto che a sua volta è stato definito come un componente strutturale dell'edizione e al contempo un insieme di annotazioni, secondo l'esempio rispettivamente dell'ontologia SE e della CAO.

Per completare la descrizione del commento filologico, occorre definire una classe nativa derivata da `Apparatus Entry Content`, che rappresenti il contenuto della nota di commento filologico, ovvero `Note Content`. Questa classe è collegata alle classi `Note` e `Critical Text Passage` in modo analogo alle classi descritte nella precedente sezione. In particolare, la proprietà `note has content (is content of note)` collega le classi `Note` e `Note Content`, mentre le classi `Note` e `Critical Text Passage` sono unite dalla proprietà `note refers to (is referred to by note)`.

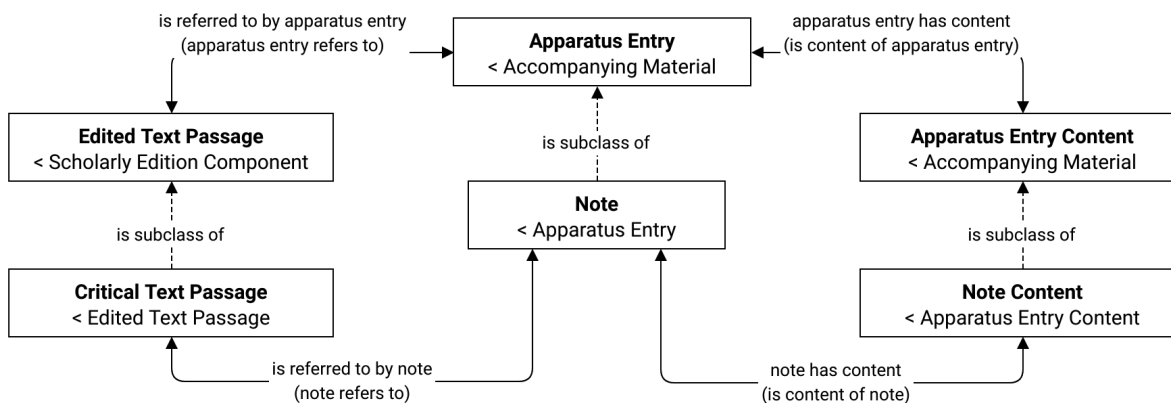


Figura 3.47 Schema delle classi e delle proprietà che rappresentano il commento filologico.

3.6.7.3 La fascia dell'apparato critico

Apparati di diverse tipologie possono essere suddivisi in fasce. Nella presente ontologia è stato formalizzato esclusivamente il concetto di fascia dell'apparato critico, inteso come apparato delle varianti. Questa scelta è dovuta al fatto che nelle edizioni critiche di opere dalla tradizione testuale molto vasta e complessa si tende molto spesso a suddividere in fasce l'apparato delle varianti, al fine di renderlo più leggibile, mentre per altre tipologie di apparati non si presenta tale necessità altrettanto frequentemente.

La fascia dell'apparato critico è rappresentata dalla classe nativa `Critical Apparatus Group` (definita come sottoclasse di `Accompanying Material`). Per indicare che un apparato critico è suddiviso in fasce, è necessario collegare la classe `Critical Apparatus Group` alla classe `Critical Apparatus` mediante la proprietà `critical apparatus has group`. Grazie alla proprietà `is part of critical apparatus group`, invece, si indica che una voce dell'apparato critico (`Critical Apparatus Entry`) rientra in una determinata fascia. Entrambe le proprietà descrivono il rapporto tra un elemento e i suoi componenti, per cui possono essere inquadrare come sottoproprietà di `lrmo0:R5 has component (is component of)`.

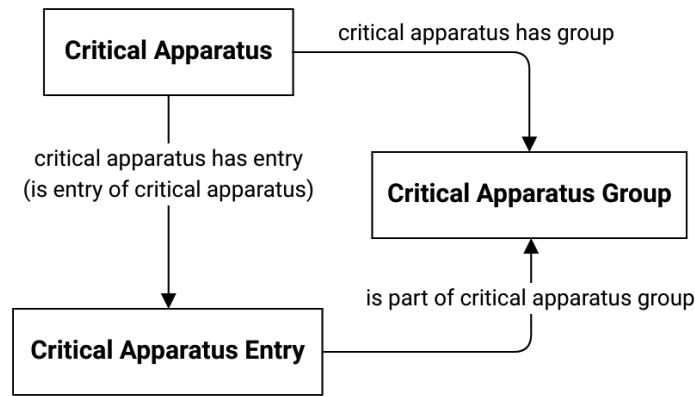


Figura 3.48 Schema delle classi e delle proprietà che descrivono il concetto di fascia dell'apparato critico.

Come esempio di edizione basata su un notevole numero di manoscritti, si prenda l'edizione delle *Rime* di Dante curata da De Robertis (Rime 2002). Nell'edizione del componimento *Le dolci rime d'amor ch'io solea*, al verso 127 (ibid., p. 78) sono associate tre diverse voci dell'apparato critico, ciascuna in una fascia distinta, riconoscibile anche grazie all'uso di una lettera maiuscola:

e sua persona acconcia di beltate

A 127. La sua Add2t b 1 M T3, A sua, z - As5, Sua g

B 127. adorna Add2 As6 [...]

C 127. adorna ch

La voce d'apparato nella prima fascia riporta le varianti alternative rispetto alla lezione scelta dall'editore e accolta nel testo critico ("e sua"). La voce della seconda fascia riporta una lezione alternativa alla pari rispetto al passo costituito, "adorna" e "acconcia", infatti, oltre a essere di un ugual numero di sillabe, sono semanticamente intercambiabili. Infine, trattandosi di un componimento che Dante ha inserito nel *Convivio*, la terza voce d'apparato riporta la lezione che potrebbe corrispondere a un mutamento subito dal testo all'atto della sua immissione all'interno del *Convivio*. Applicando la presente ontologia, l'esempio sopra riportato può essere formalizzato come in figura.

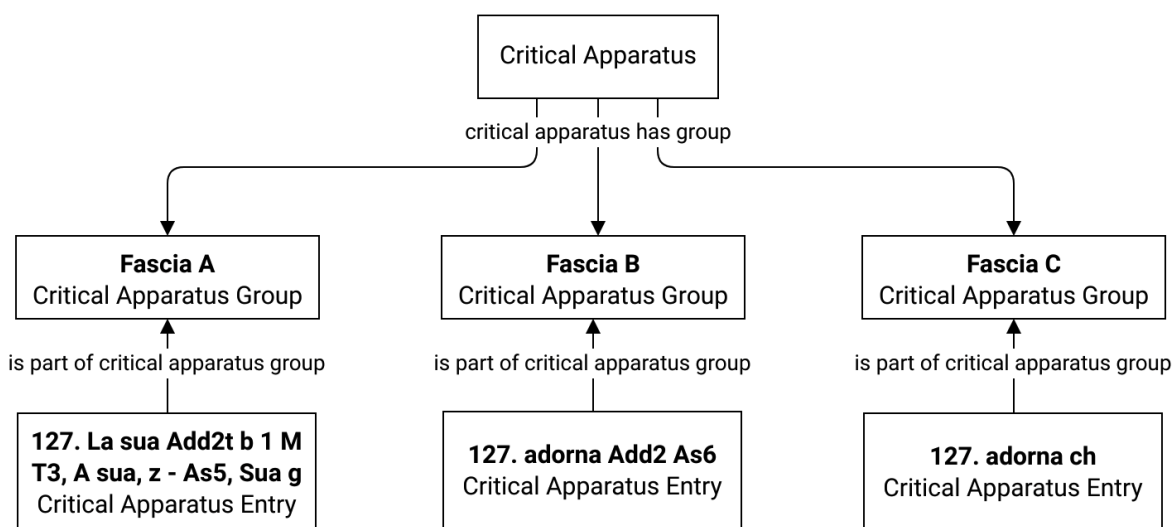


Figura 3.49 Esempio di come formalizzare tre diverse fasce dell'apparato critico.

3.6.7.4 I contenuti della voce dell'apparato critico

All'interno della voce dell'apparato critico i due contenuti principali sono la lezione scelta o proposta dall'editore e le sue varianti o, più in generale, le altre lezioni presenti nella tradizione dell'opera. La lezione scelta o proposta dall'editore, che nella tradizione anglosassone così come nelle linee guida TEI viene chiamato "lemma", viene presentata all'interno di una voce dell'apparato critico al duplice scopo di esplicitare il punto del testo cui la voce dell'apparato critico fa riferimento e di consentire al lettore di metterla più facilmente a confronto con le altre lezioni presenti nella tradizione.

A livello concettuale bisogna distinguere tra il passo vero e proprio, che è contenuto nel testo critico, e la sua copia all'interno della voce dell'apparato critico, ovvero il "lemma". Per formalizzarlo si potrebbe utilizzare la classe `Passage Selector` e, più precisamente, `Passage Quote Selector`, ma in questo modo si rischia di farlo sembrare un contenuto secondario, che può essere omesso dalla voce dell'apparato critico. In realtà, nella maggior parte delle edizioni critiche, anche digitali, il "lemma" è una parte fondamentale della voce dell'apparato.¹⁰¹

Nell'ontologia SE, il "lemma" è rappresentato dalle classi `se:Lemma` o `se:Witnessed Lemma`, a seconda che la lezione inserita nel testo critico sia presentata in apparato senza le sigle dei

¹⁰¹ Si rimanda nuovamente all'analisi delle interfacce grafiche delle edizioni scientifiche digitali contenuta nella sezione 3.3 di Martignano 2017.

testimoni in cui è attestata oppure insieme ad esse. Entrambe le classi sono sottoclassi di `se:Reading`. Nella CAO il “lemma” è rappresentato dalla classe `cao:Base Reading`, che è sottoclasse di `cao:Reading`.

Nel presente modello, dato che si riutilizza la classe `cao:Reading`, rinominata `Reading In Apparatus`, per rappresentare il concetto di lezione offerta in apparato, il lemma si presta a essere definito come una sottoclasse di `Reading In Apparatus` denominata `Base Reading In Apparatus` ed equivalente a `cao:Base Reading`.

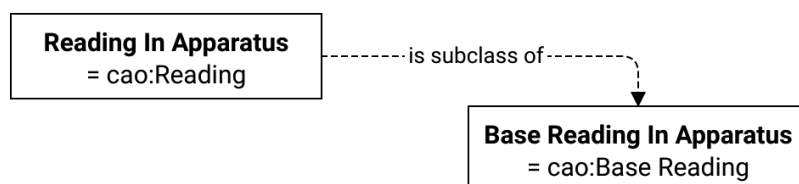


Figura 3.50 Schema del rapporto tra Reading In Apparatus e Base Reading In Apparatus.

Le classi `cao:Reading` e `cao:Base Reading` sono unite dalla proprietà `cao:has variant (is variant of)`.¹⁰² All’interno del presente modello sarebbe utile rappresentare altri tipi di relazione tra una lezione e la lezione scelta o proposta dall’editore. Una lezione, infatti, può essere una variante formale, che a livello semantico è equivalente al “lemma”, o una “variante adiafore”,¹⁰³ ovvero una variante di pari valore semantico e/o stemmatico rispetto alla variante messa a testo. Si propongono, dunque, due proprietà dedicate, entrambe sottoproprietà di `cao:has variant`:

- `has formal variant (is formal variant of)`, che unisce due lezioni semanticamente equivalenti ma diverse da un punto di vista formale;
- `has indifferent variant (is indifferent variant of)`, per collegare due varianti adiafore.

¹⁰² <https://fgiovannetti.github.io/cao/#hasVariant>.

¹⁰³ Cfr. definizione di variante adiafore fornita in Leonardi 2022: “indifferente; le varianti si definiscono adiafore quando non si trovano argomenti che rendono l’una preferibile all’altra”.

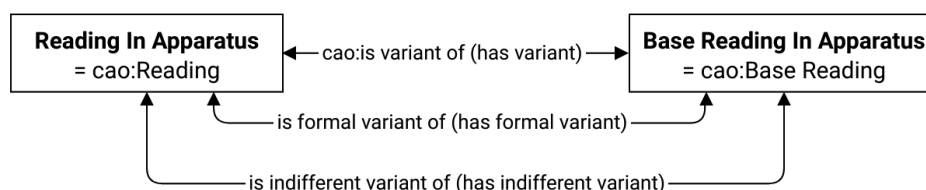


Figura 3.51 Schema delle proprietà che associano una lezione in apparato alla lezione accolta a testo mostrata nell'apparato critico.

3.6.7.4.1 Il collegamento al testimone

Per risalire al testimone o più testimoni che trasmettono la lezione a testo o le lezioni rifiutate, nelle voci d'apparato si è soliti utilizzare le sigle dei testimoni.¹⁰⁴

L'ontologia SE offre la classe `se:Siglum Reference`, che rappresenta il riferimento alla sigla di un testimone, posto accanto alla lezione o al "lemma" nell'apparato critico. La classe `se:Siglum Reference` è collegata alla lezione o al "lemma" mediante la proprietà `se:has siglum reference`.

La CAO prevede una soluzione più semplice e diretta: la classe `cao:Reading` è collegata direttamente alla classe `frbroo:F2 Expression` (che rappresenta il testimone) mediante la proprietà `cao:is witnessed by`.

A livello concettuale, la soluzione della CAO è sufficiente ad esprimere il rapporto tra una lezione presentata nell'apparato e il testimone o più testimoni in cui è attestata, a patto che la proprietà `cao:is witnessed by` consenta di collegare a una istanza di `cao:Reading` più istanze di `frbroo:F2 Expression`. Tuttavia, considerato l'obiettivo ultimo del presente modello, per stabilire correttamente come un'edizione critica debba essere visualizzata, la soluzione dell'ontologia SE è più completa e adatta.

Per questo motivo, nella presente ontologia la sigla del testimone è rappresentata dalla classe `Si-glum`, ricalcata su `se:Siglum`, come presentato nel dettaglio alla sezione dedicata all'[elenco dei testimoni](#). In modo analogo, la sigla del testimone presentata accanto a una lezione (`Reading In`

¹⁰⁴ Talvolta la sigla del testimone presenta dei caratteri in apice, che rimandano a un particolare aspetto del testimone. I numeri, per esempio, possono essere usati per distinguere le diverse mani che sono intervenute nel testimone. Nel caso di testimoni contenenti partiture musicali, accanto alla sigla è possibile indicare la voce che "canta" il testo contenente la lezione. Per esempio nella sigla "Sq^T", la "T" fa riferimento al *tenor*. Questi aspetti richiederebbero una formalizzazione dedicata, che per il momento è rimandata a un'eventuale espansione futura del modello.

Apparatus) o a un “lemma” (Base Reading In Apparatus) è rappresentata dalla classe Siglum Reference. Il collegamento tra quest’ultima e le prime due classi è rappresentato dalla proprietà `reading is witnessed by`, ritenuta maggiormente espressiva rispetto a `se:has siglum reference`.

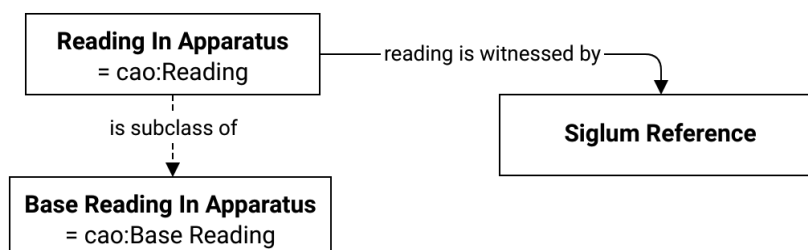


Figura 3.52 Schema del rapporto tra la lezione in apparato e il rimando alla sigla del testimone.

Un altro aspetto da considerare per descrivere il rapporto tra una lezione in apparato e le sigle dei relativi testimoni è la distinzione tra apparato positivo e apparato negativo:

“Secondo la terminologia tradizionale, si parla di apparato critico positivo quando esso riporta indicazione, oltre che dei testimoni che attestano la variante scartata, anche di quelli che attestano quella accolta; di apparato critico negativo quando sono indicati solo i testimoni della variante scartata, e quelli della variante accolta si ricavano per sottrazione” (Chiesa 2002).

Per rappresentare questo distinguo si potrebbero creare due sottoclassi *ad hoc* di `Critical Apparatus`. Per il momento la soluzione più economica, e quindi implementata nell’ontologia di esempio, è porsa quella di aggiungere alla classe `Critical Apparatus` due attributi: `isPositive` e `isNegative`, in modo tale da poter specificare questo aspetto. In una futura revisione del modello potrebbe sembrare più opportuno implementare la prima soluzione proposta.

3.6.7.4.2 Il rapporto tra la voce dell’apparato critico e le lezioni

Nella CAO il contenuto di una voce dell’apparato critico è rappresentato dalla classe `cao:Unit Variation`, che è ricalcata sull’elemento `<app>` delle linee guida TEI. Proprio come all’interno dell’elemento `<app>` è previsto che sia almeno un elemento `<rdg>` e al più un elemento `<lem>`, anche nella CAO un’istanza della `cao:Unit Variation` deve essere collegata ad almeno un’istanza di `cao:Reading` e al massimo a un’istanza di `cao:Base Reading`, rispettivamente mediante le proprietà `cao:has reading` e `cao:has base reading`, come illustrato in figura.

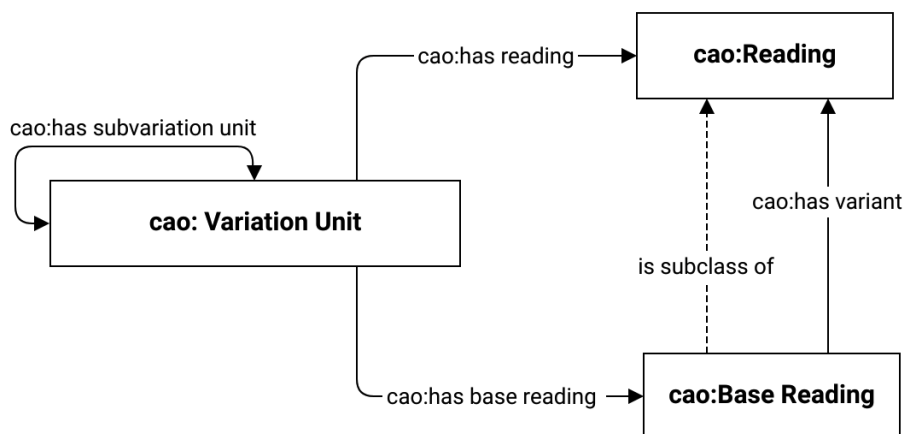


Figura 3.53 Schema del rapporto tra le classi *cao:Base Reading*, *cao:Reading* e *cao:Variation Unit*.

Nel presente modello, il contenuto della voce dell'apparato critico è rappresentato dalla classe *Critical Apparatus Entry Content*. Definendo quest'ultima come la superclasse di *Reading In Apparatus*, sia la lezione sia il "lemma" sono formalizzati come contenuti della voce dell'apparato critico. Le proprietà *cao:has reading* e *cao:has base reading* possono essere utilizzate anche nel presente modello, per poter porre dei vincoli sul numero massimo e minimo di istanze rispettivamente delle classi *Reading In Apparatus* e *Base Reading In Apparatus* previste in una voce dell'apparato critico.

3.6.7.4.3 Le sottovarianti

In una voce dell'apparato critico le lezioni potrebbero essere raggruppate in base alle famiglie cui i rispettivi testimoni appartengono oppure secondo altri criteri scientifici che a discrezione dell'editore facilitano la consultazione delle varianti.

La CAO prevede che all'interno di un'istanza della classe *cao:Unit Variation* sia possibile definire altre istanze della medesima classe, che rappresentano delle sottovariazioni, collegate tra di loro con la proprietà *cao:has subvariation unit*, come si vede nella **Figura 3.52**. Questa proprietà ha lo scopo di rappresentare dei possibili annidamenti dell'elemento `<app>` in altri elementi `<app>`, come previsto nelle linee guida TEI. Si offre un esempio.¹⁰⁵

```

<app n="a1" type="substantive">
  <rdg wit="#E1 #Hg #Ha4">
    <app n="a2" type="orthographic">

```

¹⁰⁵ L'esempio è tratto dalla versione online delle linee guida TEI: <https://www.tei-c.org/release/doc/tei-p5-doc/en/html/examples-app.html>.

```

        <lem wit="#E1 #Hg">Experience</lem>
        <rdg wit="#Ha4">Experiens</rdg>
    </app>
</rdg>
<rdg wit="#Cp #Ld1 #La">
    <app n="a3" type="orthographic">
        <lem wit="#Cp #Ld1">Experiment</lem>
        <rdg wit="#La">Ex<g ref="#per"/>iment</rdg>
    </app>
</rdg>
<rdg wit="#Ra2">
    <app n="a4" type="orthographic">
        <lem resp="#ed2013">Eriment</lem>
        <rdg wit="#Ra2">Eryment</rdg>
    </app>
</rdg>
</app>

```

Nel presente modello, la definizione di “sottounità di variazione”, proposta nella CAO, è stata sostituita dalla definizione del concetto di “gruppo di lezioni”, calcolato sull’elemento TEI <rdg-Grp>. ¹⁰⁶ La codifica TEI sopra riportata può essere modificata utilizzando <rdgGrp> al posto di <app> come segue:

```

<app type="substantive">
    <rdgGrp type="subvariants">
        <lem wit="#E1 #Hg">Experience</lem>
        <rdg wit="#Ha4">Experiens</rdg>
    </rdgGrp>
    <rdgGrp type="subvariants">
        <lem wit="#Cp #Ld1">Experiment</lem>
        <rdg wit="#La">Ex<g ref="#per"/>iment</rdg>
    </rdgGrp>
    <rdgGrp type="subvariants">
        <lem resp="#ed2013">Eriment</lem>
        <rdg wit="#Ra2">Eryment</rdg>
    </rdgGrp>
</app>

```

Il vantaggio di utilizzare il concetto di “gruppo di lezioni” rispetto a quello di “sottounità di variazione” è che non si creano fraintendimenti con il concetto di voce dell’apparato in sé. La voce

¹⁰⁶ “<rdgGrp> (reading group) within a textual variation, groups two or more readings perceived to have a genetic relationship or other affinity.” (<https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-rdgGrp.html>).

dell'apparato critico rimane una sola e al suo interno l'editore può organizzare le lezioni nel modo che ritiene più opportuno.

Il concetto di gruppo di lezioni è rappresentato dalla classe `Reading Group`, sottoclasse di `Critical Apparatus Entry Content` e collegata a `Reading In Apparatus` attraverso la proprietà `is part of reading group`.

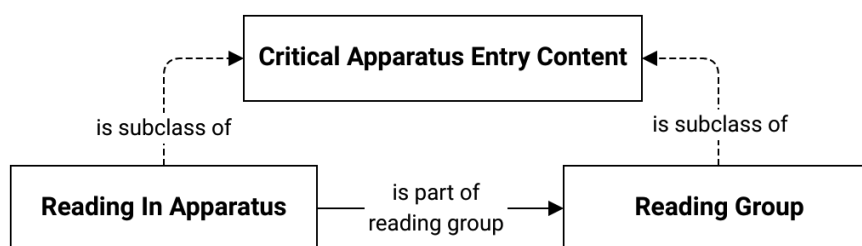


Figura 3.54 Schema delle classi e delle proprietà disponibili per rappresentare un gruppo di lezioni.

In molte edizioni critiche a stampa, specie in quelle basate su una vasta tradizione testuale, la variante che riunisce due o più testimoni viene presentata una sola volta, senza presentare le forme originarie presenti nei singoli testimoni. Come si legge, per esempio, nei criteri di edizione all'interno dell'introduzione all'edizione delle Rime di Dante, curata da De Robertis: "La variante che riunisce due o più testimoni viene data in lezione standard, quella di un singolo testimone in trascrizione diplomatica." Questi casi sono rappresentabili formalmente nel modello grazie alla distinzione tra `Reading` e `Reading In Apparatus`. Per cui la prima parte dell'esempio di apparato codificato in XML/TEI sopra riportato, potrebbe essere espressa come si vede in figura.

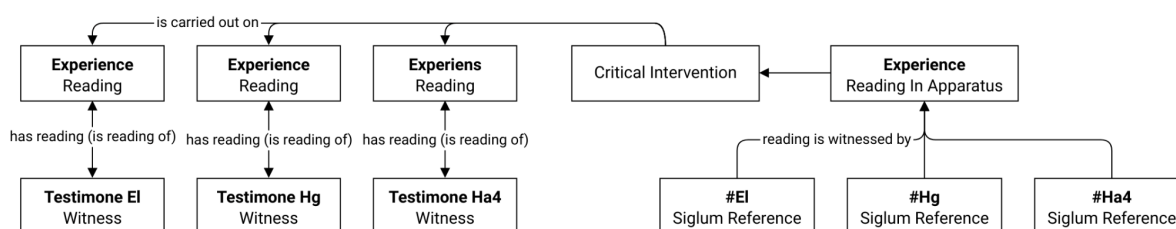


Figura 3.55 Esempio di come formalizzare il rapporto tra Reading e Reading In Apparatus.

Qualora l'editore volesse presentare all'interno dell'apparato anche le varianti formali, queste potrebbero essere formalizzate sempre come istanze della classe `Reading In Apparatus`, e poste eventualmente in una fascia dell'apparato critico a parte o raggruppate mediante `Reading Group` all'interno della voce dell'apparato critico. La differenza tra `Critical Apparatus Group` e

Reading Group è che le fasce d'apparato sono le stesse trasversalmente in tutto l'apparato critico. I gruppi di lezioni, invece, possono variare di voce in voce.

3.6.7.4.4 Le note alle lezioni

All'interno di una voce dell'apparato critico possono esserci diversi tipi di note. Alcune sono incentrate su una particolare lezione e al loro interno l'editore offre chiarimenti su come la lezione si presenta all'interno del testimone o su altre caratteristiche della variante. Altri tipi di note possono riguardare l'intera voce dell'apparato critico e l'editore se ne serve per motivare le proprie scelte. Nel campo della filologia preso a riferimento come dominio del presente modello, la consuetudine è quella di separare questo secondo tipo di note dall'apparato critico, inserendole nel commento filologico, dato che spesso sono piuttosto lunghe. In questo modo l'apparato critico consente una consultazione più agevole delle sole varianti.

Per una questione di completezza e di piena compatibilità con la CAO,¹⁰⁷ nel presente modello si offre la classe `Critical Apparatus Entry Note` per rappresentare qualsiasi nota inserita all'interno di una voce dell'apparato critico e che può essere riferita all'intera voce, a una lezione, al "lemma" o a un gruppo di lezioni, grazie alla proprietà `entry content has note` che la unisce direttamente a un'istanza di `Critical Apparatus Entry Content`.

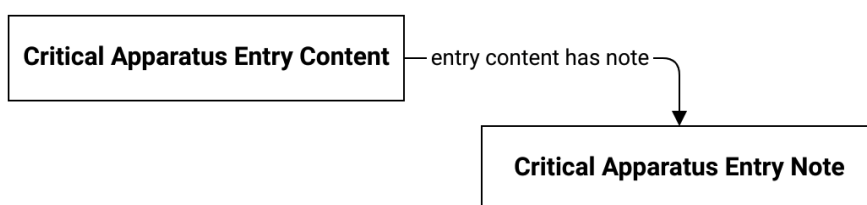


Figura 3.56 Schema della classe *Critical Apparatus Entry Content Note*.

3.6.8 L'introduzione

In un'edizione critica il testo è introdotto al lettore da un altro testo di dimensione variabile, che possiamo definire banalmente "introduzione", all'interno del quale l'editore inquadra l'opera edita e la sua tradizione da un punto di vista storico, linguistico, letterario e culturale, illustrando inoltre

¹⁰⁷ "The CRM property `crm:P3_has_note` links a reading or apparatus entry to a scholarly note. The range of the property `crm:P3_has_note` is an `rdfs:Literal`. It is possible to use `crm:P2_has_type` on the note to further characterize the type of annotation (a possible value is, e.g., 'philological')" (CAO).

come è stata allestita l'edizione. L'introduzione può essere suddivisa in parti distinte, specie se molto lunga e articolata. Nel modello è rappresentata, dunque, dalla classe `Introduction`, mentre una sua parte dalla classe `Introduction Part`, collegate fra di loro dalla proprietà `introduction has part (is part of introduction)`. `Introduction` è sottoclasse di `Accompanying Material`, mentre la proprietà `introduction has part (is part of introduction)` è sottoproprietà di `lrmo0:R5 has component (is component of)`.

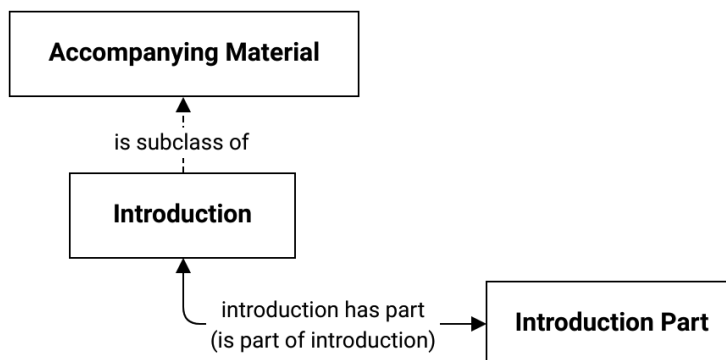


Figura 3.57 Schema delle classi per descrivere il concetto di introduzione e delle sue parti.

Tra le parti in cui è suddivisa un'introduzione possono esserci anche l'elenco dei testimoni (o 'testimoniale') e lo *stemma codicum* (o un'altra rappresentazione grafica dei rapporti tra i testimoni). La distinzione tra l'introduzione e questi componenti non è sempre netta, in quanto possono essere sia presentati a parte sia inseriti in forma più o meno discorsiva all'interno dell'introduzione. È parso opportuno predisporre nel modello delle classi dedicate, illustrate nel dettaglio oltre. Tuttavia, possono essere descritte anche come parti dell'introduzione.

3.6.8.1 L'elenco dei testimoni

L'allestimento del testo critico può essere stato condotto su una selezione più ristretta di testimoni o un solo testimone, ma il testimoniale presenta tutti i testimoni analizzati dall'editore. Per ogni testimone, oltre alla segnatura completa, vengono fornite la sigla ed eventualmente una descrizione. La descrizione può essere più o meno approfondita e trattare diversi aspetti del testimone: lo stato di conservazione, l'aspetto materiale, l'analisi paleografica, ecc.

L'elenco dei testimoni è rappresentato dalla classe `Witnesses List`, il cui nome è stato ricalcato sull'elemento `<listWit>` del vocabolario TEI.¹⁰⁸

La classe `Witnesses List` è collegata alla classe `Witness`, definita nella sezione del modello dedicata ai concetti base, tramite la proprietà `witness is presented in`. Le istanze di `Witness` sono le unità logiche che corrispondono alle entrate dell'elenco dei testimoni.

Come anticipato, la classe `Siglum` (che corrisponde a quella dell'ontologia SE `se:Siglum`) rappresenta la sigla del testimone che funge da identificativo. La classe `Witness` è collegata a `Siglum` mediante la proprietà `witness is identified by`. Mentre la classe `Siglum Reference` è collegata alla classe `Siglum` grazie alla proprietà `refers to siglum`.

Le informazioni riguardanti un testimone possono essere formalizzate molto dettagliatamente servendosi di vocabolari già esistenti, senza dover elaborare nuove classi:

“There are, indeed, two main ways to describe the witnesses involved in the reconstruction of a critical text: describing each witness within the model in detail or relying on external descriptions, such as those provided by museums, cultural institutions, and other datasets (e.g. the British Museum, Pleiades, GeoNames, Worldcat, VIAF, DBpedia). CRM-FRBR is fully equipped for the description of manuscripts as museum artefacts as well as for publications” (CAO).

Nel presente modello l'obiettivo è rappresentare le informazioni sul testimone così come sono fornite dall'editore all'interno dell'edizione critica, ovvero in forma più discorsiva che catalogafica e accompagnate, eventualmente, da commenti dell'editore. A questo scopo è stata creata la classe nativa `Witness Description`, collegata alla classe `Witness` mediante la proprietà `witness is described by`.

¹⁰⁸ Cfr. <https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-listWit.html>.

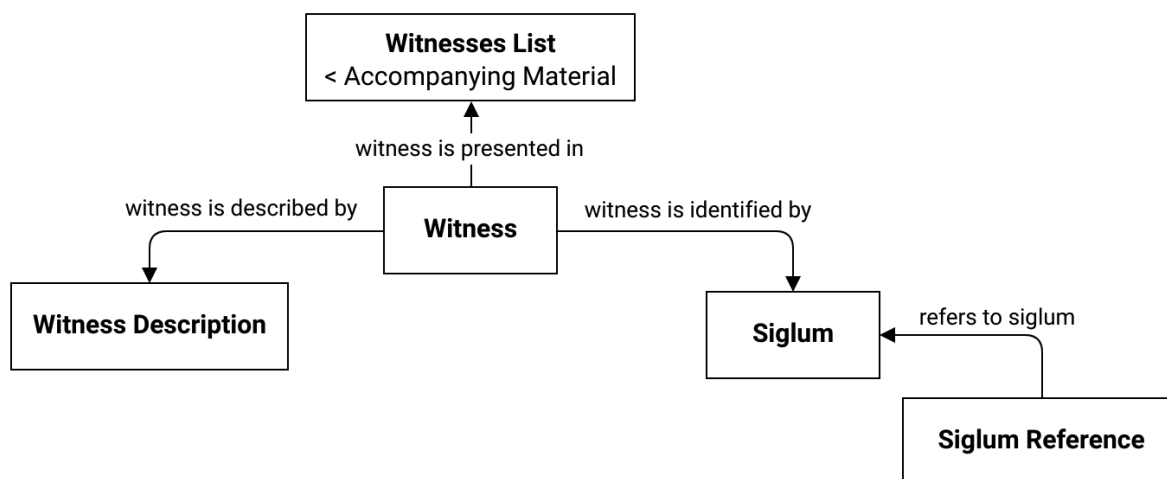


Figura 3.58 Schema delle classi e delle proprietà disponibili per formalizzare l'elenco dei testimoni.

3.6.8.2 Stemma Codicum

In diverse edizioni critiche, l'editore, per presentare in forma schematica i rapporti tra le varianti o i rapporti genealogici (o di altra natura) individuati tra i diversi testimoni, si serve di grafici. Tra questo tipo di grafici, uno dei più utilizzati è lo *stemma codicum*, un grafo ad albero che rappresenta i rapporti genealogici tra i diversi testimoni della tradizione di un'opera. L'editore utilizza lo *stemma* e vi fa riferimento sia per motivare gli interventi effettuati in fase di allestimento del testo critico sia per raggruppare in fasce le entrate dell'apparato critico o le lezioni all'interno di una voce dell'apparato critico. Come anticipato, molto spesso lo *stemma* è parte integrante dell'introduzione.

Altri tipi di rappresentazioni grafiche, utilizzate nelle edizioni critiche che applicano i metodi della filogenesi e della cladistica, sono i cladogrammi, degli schemi che rappresentano sia le relazioni tra i testimoni sia la loro reciproca distanza, relativamente a una singola variante o a una porzione di testo (cfr. Leonardi 2022, p. 129).

Un esempio di applicazione di questo tipo di grafi si trova nell'edizione critica digitale della *Commedia* a cura di Prue Shaw, in cui è possibile visualizzare la variazione testuale in forma di cladogrammi, come si vede in figura.

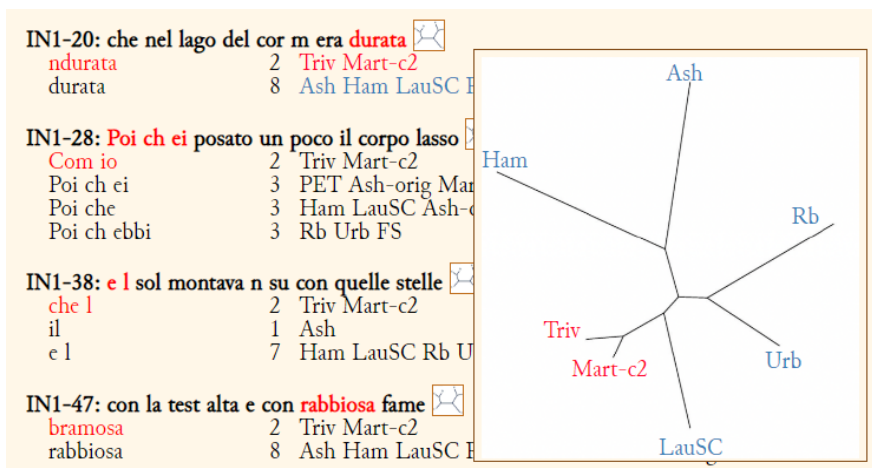


Figura 3.59 Esempio di cladogramma tratto dalla seconda edizione digitale della *Commedia* curata da Prue Shaw.

Lo *stemma codicum*, così come il cladogramma dell'esempio, possono essere formalizzati in modi diversi. L'ideale sarebbe definire delle classi e delle proprietà per descrivere direttamente i rapporti reciproci tra i testimoni o le varianti che sono rappresentati nel grafo, per esempio attraverso delle classi tipo "Archetipo" e "Subarchetipo" e delle proprietà quali "è figlio di" e "è fratello di". Nel contesto del presente modello concettuale, è sufficiente rappresentare lo *stemma codicum* (e altre tipologie di grafo) come dei grafici che fanno parte dell'edizione. Ciò non toglie che eventuali espansioni future del modello descrivano nel dettaglio i rapporti presentati nello *stemma codicum* o in altri grafici, consentendo l'implementazione di grafici navigabili e interattivi.

Tra le ontologie analizzate, le classi che rappresentano il concetto di immagine sono:

- `doco:Figure`, "A communication object comprising one or more graphics, drawings, images, or other visual representations";¹⁰⁹
- `crm:E36 Visual Item`, "This class comprises the intellectual or conceptual aspects of recognisable marks, images and other visual works. **This class does not intend to describe the idiosyncratic characteristics of an individual physical embodiment of a visual item, but the underlying prototype.** For example, a mark such as the ICOM logo is generally considered to be the same logo when used on any number of publications. The size, orientation and colour may change, but the logo remains uniquely identifiable. The same is

¹⁰⁹ <https://sparontologies.github.io/doco/current/doco.html#d4e685>.

true of images that are reproduced many times. This means that visual items are independent of their physical support.” (CIDOC CRM v. 7.1.2, p. 82)

La seconda classe è parsa più adatta a rappresentare il concetto di *stemma codicum* e di altri grafici affini, in quanto permette di descrivere i grafici su un piano più astratto, che prescinde dalle loro realizzazioni grafiche.

Per il momento, il presente modello concettuale include solo il concetto di *stemma codicum*, rappresentato dalla classe nativa *Stemma Codicum*, definita come una sottoclasse di *crm:E36 Visual Item*.

In eventuali espansioni del modello, ulteriori tipologie di grafici possono essere facilmente aggiunte, definendole come sottoclassi di *crm:E36 Visual Item* e di *Scholarly Edition Component* o di *Accompanying Material*, a seconda che ricoprano un ruolo secondario o centrale nell’edizione.

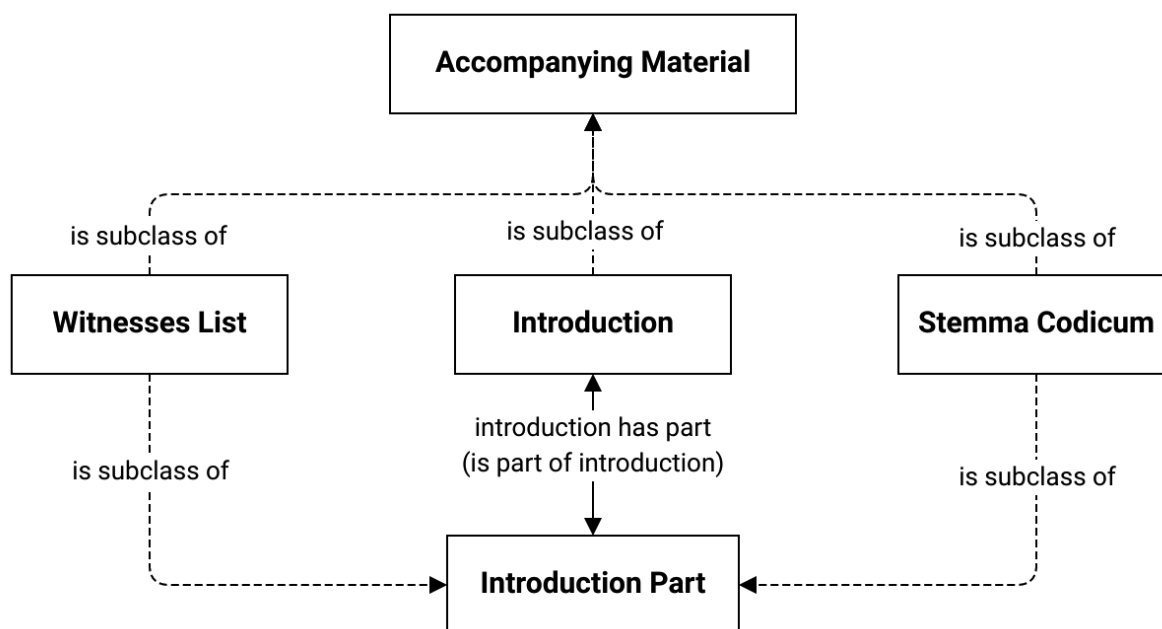


Figura 3.60 Schema dei rapporti tra le classi *Witnesses List*, *Stemma Codicum* e *Introduction*.

3.7 La suddivisione in moduli

Come specificato nel corso della descrizione, l’ontologia realizzata è suddivisa a livello logico in tre parti distinte che adempiono a funzioni diverse. La prima parte, dedicata ai concetti di base, è stata concepita come punto di partenza per lo sviluppo non solo della presente ontologia ma anche degli

altri modelli concettuali. Molti dei concetti in questa parte dell'ontologia, infatti, grazie anche all'utilizzo di LRM_{OO}, sono stati formalizzati in modo generico e facilmente adattabile alla descrizione di altre tipologie di edizioni. La terza parte, incentrata sui componenti di un'edizione critica digitale, funge da base per lo sviluppo di uno strumento di visualizzazione. La seconda parte, invece, è ulteriormente suddivisibile in due sezioni. Da un lato descrive alcuni dei componenti di un'edizione critica digitale che sono frutto del lavoro di interpretazione e ricostruzione da parte dell'editore, cioè le lezioni in apparato (e tutte le relative classi) e i passi del testo edito. Dall'altro lato, questa parte dell'ontologia, che rende trasparente e verificabile il lavoro svolto dall'editore, può essere applicata nel campo delle edizioni semantiche digitali (vd. 2.5.1). L'ontologia, nonostante a livello logico sia suddivisa in moduli, in termini concreti è stata sviluppata come un'unica ontologia, quindi un unico documento OWL. In questa fase iniziale, si è ritenuto opportuno procedere in questo modo al fine di facilitare la revisione dell'ontologia, in particolare, da un punto di vista dell'organizzazione gerarchica e della dipendenza da ontologie esterne. Tuttavia, l'ontologia, essendo anche di notevoli dimensioni, risulterebbe più usabile per i tre diversi scopi sopra descritti, se venisse suddivisa in tre moduli distinti, ovvero in tre diverse ontologie collegate tra loro. Ciascuna parte potrebbe essere ulteriormente sviluppata e modificata in modo autonomo,¹¹⁰ favorendo anche il riuso del modello concettuale. Per questo motivo, una volta messa a punto, l'ontologia dovrà essere “modularizzata” (Doran et al. 2007). Le ontologie risultanti saranno pubblicate nel medesimo *repository* GitHub e messe a disposizione attraverso lo stesso dominio di quella attuale.

¹¹⁰ Per maggiori informazioni sui vantaggi della modularizzazione di un'ontologia e di come possa essere effettuata si rimanda a (Doran et al. 2007).

Capitolo 4

Dal modello concettuale di esempio alla progettazione di uno strumento di visualizzazione

Nel capitolo 2 è stata descritta la strategia di modellizzazione che si propone come soluzione al problema dell'inadeguatezza degli strumenti informatici per la visualizzazione di edizioni scientifiche digitali rispetto alle esigenze scientifiche degli studiosi. Tale strategia prevede a monte la realizzazione di una rete di modelli editoriali in forma di modelli concettuali, ovvero dei modelli che descrivono da un punto di vista astratto le diverse tipologie di edizione, inquadrare secondo i rispettivi approcci filologici. I diversi modelli editoriali / concettuali, formalizzati come ontologie nel linguaggio OWL 2, hanno il duplice scopo di facilitare la comunicazione tra i filologi (gli “esperti del dominio”) e gli sviluppatori informatici e di fornire un’indicazione dei requisiti di base degli strumenti informatici, partendo dalla definizione dei componenti di un particolare tipo di edizione scientifica digitale e dei rapporti tra i componenti.

La seconda operazione prevista nella strategia di modellizzazione è che, una volta stabilito un modello concettuale, quest’ultimo venga utilizzato, secondo i principi del *domain-driven design* (Evans 2004), per progettare in termini concreti la struttura di uno strumento informatico e le sue funzionalità. L’idea è cioè di derivare dal modello concettuale (“domain model”) dei modelli di tipo implementativo, che nel capitolo 2 abbiamo definito “logical system models” e “implementation models” (vd. 2.3). Questa operazione di derivazione andrebbe condotta dopo avere testato e messo a punto il modello concettuale di base, e coinvolgendo gli sviluppatori informatici, così da stabilire insieme a quest’ultimi le tecnologie da utilizzare per sviluppare lo strumento.

La derivazione dal modello concettuale al modello implementativo, può essere facilitata “traducendo” prima il modello concettuale in UML, un linguaggio di modellizzazione di sistemi informatici molto noto e utilizzato nel campo dell’ingegneria del software, come già accennato nel capitolo 3. La traduzione da OWL a UML è un passaggio delicato, che deve essere svolta dagli sviluppatori informatici insieme ai filologi (o esperti di dominio), per garantire che la “semantica” del modello, ciò che il modello “significa” o “esprime”, non venga modificata. Idealmente la strategia di modellizzazione dovrebbe elaborare non solo i modelli concettuali, ma anche dei morfismi, ovvero delle indicazioni precise di come i modelli debbano essere tradotti in UML. In questo modo i modelli implementativi potrebbero essere derivati direttamente da un modello concettuale in UML, anziché dall’ontologia in OWL, che comunque rispecchia il modello editoriale espresso dall’ontologia.

Questo capitolo, dunque, dopo alcune premesse teoriche sulla programmazione orientata agli oggetti e sul linguaggio di modellizzazione UML, presenta:

- delle indicazioni di massima su come tradurre il modello concettuale di esempio dal linguaggio OWL a UML;
- degli spunti di riflessione su come progettare uno strumento informatico, analizzando da un punto di vista implementativo i componenti di un’edizione scientifica digitale definiti nel modello concettuale di esempio e i problemi che pongono.

L’obiettivo di questa parte del lavoro di ricerca non è quello di preparare dei diagrammi UML completi e pronti all’uso, ma di mettere alla prova l’espressività e l’efficacia di un’ontologia per determinare la progettazione di uno strumento informatico.

4.1 Uno sviluppo *object-oriented* di strumenti di visualizzazione complessi

Gli argomenti esposti in questo capitolo necessitano di una premessa tecnica riguardo la natura degli strumenti informatici per cui la strategia di modellizzazione è stata ideata.

Come illustrato nel capitolo 1 (1.3.4), ci sono diverse tipologie di programmi e software che rientrano nella categoria di “strumento di visualizzazione”. Alcuni di questi li abbiamo definiti di “grado zero”, perché nella sostanza sono dei semplici fogli di stile XSLT che convertono un’edizione digitale

in un formato per la pubblicazione sul web (come HTML) o a stampa (per esempio PDF). Appena sopra il “grado zero” ci sono gli strumenti come TEI Boilerplate e CETEIcean, che convertono l’edizione in un formato idoneo alla pubblicazione nel web e offrono un’infrastruttura mediante la quale è possibile sviluppare delle funzionalità *ad hoc*. Questo secondo tipo di strumenti per poter essere usato richiede un certo livello di competenze informatiche. Il terzo e ultimo livello è quello di strumenti come EVT, TEI Publisher e Versioning Machine, pronti all’uso sia per pubblicare che per fruire dell’edizione, in quanto offrono un insieme di funzionalità rivolte all’utente finale, il lettore.

La strategia di modellizzazione è stata concepita per realizzare strumenti dell’ultimo tipo, che si rivolgono, quindi, all’utente-editore e all’utente-lettore, e che permettono di visualizzare un’edizione scientifica digitale in un’interfaccia grafica elaborata e dotata di molte funzionalità. Questa scelta dipende da due ragioni principali. In primo luogo, per realizzare gli strumenti dei primi due livelli, mettere in atto la strategia di modellizzazione proposta risulta essere troppo oneroso e ridondante (fermo restando che i modelli concettuali sarebbero dei riferimenti teorici utili anche per questo tipo di strumenti). A maggior ragione se si considera che gli strumenti dei primi due livelli sono spesso utilizzati dagli sviluppatori per realizzare gli strumenti dell’ultimo livello. In secondo luogo, nell’ultimo livello rientra buona parte degli strumenti di visualizzazione realizzati *ad hoc* per un particolare progetto di edizione. Lo scopo ultimo della strategia di modellizzazione proposta è quello di favorire la realizzazione di strumenti “prêt-à-porter” personalizzabili, innovativi e sofisticati, disincentivando la produzione di software personalizzati. In altre parole, la strategia di modellizzazione mira a essere un “meccanismo grazie al quale i prodotti di ricerca più avanzati si acquisiscono, si stabilizzano e diventano usufruibili da un numero elevato di ricercatori, dando allo stesso tempo un ruolo anche etico all’esistenza di una ricerca scientifica digitale costosa ed elitaria” (Mancinelli e Pierazzo 2020).

I software di visualizzazione sono complessi, in quanto non devono eseguire un solo compito, come la conversione di un documento da un formato all’altro, ma offrire all’utente-lettore tante funzionalità diverse per poter analizzare e studiare i contenuti di un’edizione scientifica.

Nel campo dell’ingegneria del software ci sono diversi paradigmi di programmazione. Il più vantaggioso da adottare nel campo della filologia digitale è quello “a oggetti” o “object-oriented programming” (OOP), come sostenuto in (Del Grosso et al. 2017). Questo paradigma di

programmazione facilita, infatti, lo sviluppo e la manutenzione di applicativi complessi e favorisce il riuso dei componenti.

La programmazione orientata agli oggetti è un paradigma basato sul concetto di “oggetto”. Un oggetto è una porzione di codice costituita da variabili (o attributi o proprietà), che costituiscono i dati dell’oggetto, e da operazioni (o metodi) che l’oggetto può effettuare. Più precisamente, nell’OOP una classe rappresenta l’insieme di tutti i possibili oggetti che possiedono caratteristiche comuni, mentre un oggetto è un’istanza di una classe.

Qui emerge come il paradigma a oggetti faciliti il passaggio dal modello concettuale alla progettazione di come lo strumento di visualizzazione debba essere implementato. Il modello concettuale individua e descrive i componenti che costituiscono un tipo di ESD (cioè un modello editoriale), in altre parole gli oggetti del dominio applicativo dello strumento di visualizzazione. Dato che lo strumento di visualizzazione deve presentare all’utente l’ESD, consentendogli di interagire con essa, i suoi componenti possono essere a livello implementativo altrettanti oggetti.

“Many objects in a design come from the analysis model. [...] The abstractions that emerge during design are key to making a flexible design” (Gamma et al. 1994, p. 11).

4.2 UML

UML (Unified Modeling Language) è un linguaggio standard per la modellizzazione visiva utilizzato per progettare, descrivere e analizzare sistemi informatici.

Nel concreto, i modelli UML sono dei diagrammi che possono essere usati in modo analogo alla planimetria di un edificio: indicano come l’applicativo debba essere realizzato, prima che si inizi a scrivere il codice.

UML fu creato negli anni ’90 da tre sviluppatori informatici, Grady Booch, Ivar Jacobson e Jim Rumbaugh. Nel 1997 la versione 1.1 di UML divenne uno standard adottato dall’Object Management Group, organizzazione che tuttora lo sviluppa e lo mantiene. La versione più recente di UML, nonché quella presa a riferimento nel presente lavoro di ricerca, è la 2.5.1,¹ pubblicata nel 2017.

¹ Il documento di specifica della versione 2.5.1 di UML è disponibile online (<https://www.omg.org/spec/UML/>) ed è identificato nel presente elaborato dalla seguente sigla bibliografica “UML v. 2.5.1”.

Nel campo dell'ingegneria del software esistono altri linguaggi oltre a UML per modellizzare e progettare l'architettura di un software. Tra i più noti si citano C4 Model,² ArchiMate³ e SysML.⁴ Rispetto a questi linguaggi, UML può risultare più complesso da utilizzare e da apprendere, in quanto è molto generico e consente di descrivere un sistema da diversi punti di vista e a diversi livelli di astrazione. Se il sistema da modellare non è particolarmente complesso, l'utilizzo di UML potrebbe risultare eccessivamente dispendioso anche in termini di tempo. D'altro canto se ci si attiene a pochi punti di vista e soprattutto al giusto livello di astrazione UML è semplice da usare. UML è sicuramente il linguaggio più noto e utilizzato nel campo dell'ingegneria del software, offre un'ampia gamma di notazioni standard ed è un linguaggio molto espressivo e potente. Inoltre, permette di utilizzare anche solo alcune parti specifiche della propria notazione e gli strumenti dedicati consentono di generare automaticamente il codice a partire dai diagrammi, velocizzando lo sviluppo del codice.⁵ Per questi motivi, UML è sembrato lo strumento di modellizzazione più adatto per la strategia di modellizzazione.

Come prodotto finale del presente lavoro di ricerca sarebbe stato possibile scrivere direttamente del codice, realizzando per esempio una libreria di componenti pronti all'uso. Tuttavia, i diagrammi UML presentano il vantaggio di risultare più facilmente comprensibili anche agli esperti di dominio, facilitando, in questa fase della ricerca, la discussione tra filologi e sviluppatori informatici su come lo strumento informatico debba essere sviluppato.

4.2.1 Cenni teorici su UML e la programmazione a oggetti

Un modello UML è costituito da tre categorie principali di elementi, ciascuno dei quali può essere utilizzato per descrivere un tipo di oggetto all'interno del sistema informatico che viene modellato: i "classificatori", gli "eventi" e i "comportamenti".

² <https://c4model.com/>.

³ <https://publications.opengroup.org/standards/archimate/c226>.

⁴ <https://sysml.org/>.

⁵ Cfr. <https://icepanel.medium.com/7-modelling-languages-for-software-architecture-2ed3a9f75343>.

I “classificatori” descrivono un insieme di oggetti. Un oggetto è associato ad altri oggetti mediante delle relazioni. Inoltre, un oggetto è caratterizzato da uno “stato”, che identifica i valori delle proprietà dell’oggetto.

Gli “eventi” descrivono un insieme di possibili fenomeni che possono accadere all’interno di un sistema e che comportano delle conseguenze.

Un “comportamento” descrive un insieme di possibili azioni che (in un certo periodo di tempo) possono generare altri eventi oppure che rispondono a un evento accaduto. Tra gli eventi generati da un comportamento o che scatenano un comportamento ci sono anche la modifica allo stato degli oggetti.⁶

Per comprendere meglio i concetti enunciati, si immagini come esempio di strumento informatico un registro elettronico da utilizzare in una scuola. Il registro elettronico deve gestire i profili dei diversi utenti, che possono essere suddivisi in due principali categorie: la categoria “Alunno” e la categoria “Professore”. Queste due categorie di enti sono rappresentate da due classi distinte. In UML: “A Class classifies a set of objects and specifies the features that characterize the structure and behavior of those objects” (UML v. 2.5.1, p. 263). Una classe è rappresentata in UML da un rettangolo. Le due classi “Alunno” e “Professore” in UML si rappresentano come segue.



Figura 4.61 Class diagram delle classi “Alunno” e “Professore”.

Ogni classe è caratterizzata da degli attributi. Per esempio la classe Alunno come attributi di base ha la “classe”, che è un numero intero tra 1 e 5, e la “sezione”, che è una lettera dell’alfabeto. Quando nel registro elettronico si deve creare un profilo utente per un nuovo studente “Mario Rossi”, che è stato inserito nella classe 1^A, il sistema informatico deve creare un nuovo oggetto della classe “Alunno” avente come classe “1” e come sezione “A”.

⁶ Cfr. UML v. 2.5.1, pp. 54 - 55.

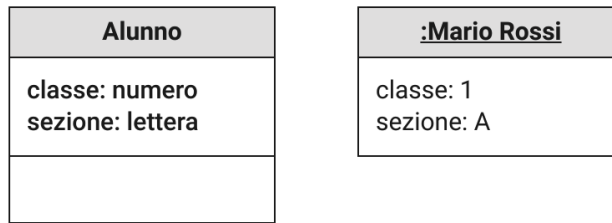


Figura 4.62 Class diagram della classe “Alunno” e del suo oggetto “Mario Rossi”.

Lo “stato” dell’oggetto “Mario Rossi” è determinato dagli attributi “classe” e “sezione” e dai loro rispettivi valori. Nel momento in cui per Mario Rossi finisce il primo anno scolastico e viene promosso alla classe seconda, nel sistema informatico occorre aggiornare i dati, modificando lo stato dell’oggetto “Mario Rossi”: il valore dell’attributo “classe” da “1” deve passare a “2”. L’evento è la promozione dalla prima classe alla seconda dell’oggetto “Mario Rossi”. Il comportamento è l’aggiornamento del valore dell’attributo “classe” all’interno dell’oggetto “Mario Rossi”.

Il passaggio da una classe a un’altra, così come da una sezione a un’altra è un evento che coinvolge tutti gli oggetti della classe “Alunno”. Dunque, nella classe “Alunno” sono specificati dei metodi, delle operazioni predefinite, che consentono di intervenire sullo stato dell’oggetto, in particolare “cambiaClasse” metodo che richiede il nuovo valore da assegnare all’attributo classe e “cambiaSezione” che analogamente richiede di specificare il nuovo valore dell’attributo sezione.

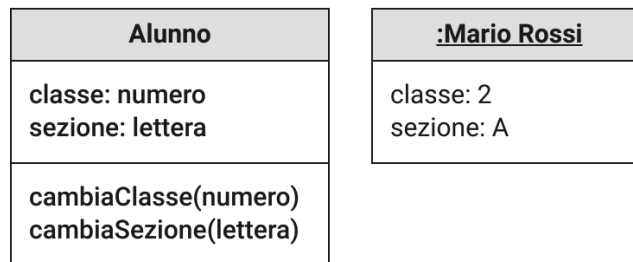


Figura 4.63 Class diagram della classe “Alunno” con i relativi metodi e dell’oggetto “Mario Rossi”.

Quando si vuole modificare la classe dell’oggetto “Mario Rossi”, a livello di codice è sufficiente “invocare” il metodo “cambiaClasse”, comune a tutti gli oggetti della classe “Alunno”, sull’oggetto “Mario Rossi” passando il nuovo valore.

Gli oggetti possono interagire tra di loro e possono modificare l’uno lo stato dell’altro. Immaginiamo un “evento”: il professore di matematica “Pico De Paperis” mette un voto all’alunno “Mario Rossi”. In questo caso come si comportano i due oggetti? Immaginiamo che la classe “Professore” abbia il metodo “mettiVoto”. Questo metodo riceve in input il voto e il nome dell’alunno che il

professore inserisce o seleziona nell'interfaccia grafica del registro elettronico. Il metodo chiama l'oggetto che corrisponde al nome dell'alunno selezionato dal professore. La classe "Alunno" ha, invece, il metodo "riceviVoto", che prende il voto e lo salva insieme agli altri voti dell'alunno.

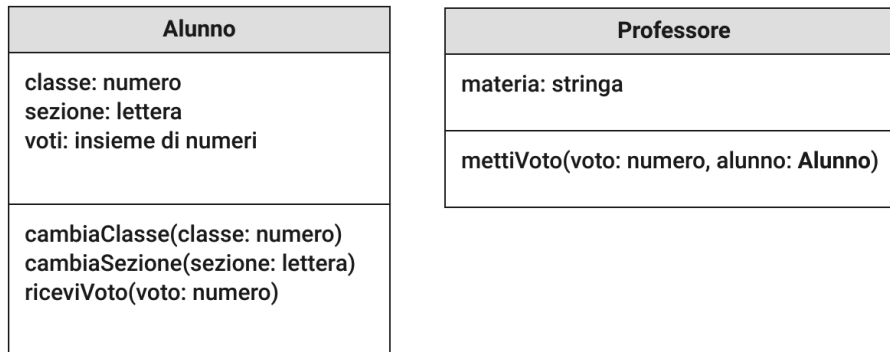


Figura 4.64 Class diagram completo delle classi "Alunno" e "Professore".

Nella figura sopra, si vede come l'operazione "mettiVoto" della classe "Professore" richiami la classe "Alunno". Le figure finora presentate sono dei "class diagram",⁷ ovvero dei diagrammi UML che presentano le classi nelle quali è organizzato il codice di uno strumento informatico. Una classe è rappresentata da un rettangolo, all'interno del quale nell'intestazione è riportato il nome della classe, mentre all'interno è suddiviso in due parti, quella in alto dedicata alle proprietà (o attributi) e quella in basso dedicata ai metodi (o operazioni) che caratterizzano la classe. Degli attributi si specifica la tipologia di dato (o di dati) che rappresentano, specificandola dopo i due punti (":"). Per i metodi, dopo i rispettivi nomi, si specificano tra parentesi i dati che necessitano in input e le loro rispettive tipologie (es. "(voto: numero)"). Nel caso in cui i metodi restituiscano un dato in output, se ne può specificare la tipologia, mettendo due punti (":") dopo la parentesi chiusa e poi il tipo di dato (es. "mettiVoto(voto: numero): boolean", dove il valore booleano può indicare se il voto è stato messo correttamente oppure no).

⁷ Cfr. "Class diagram is UML structure diagram which shows structure of the designed system at the level of classes and interfaces, shows their features, constraints and relationships - associations, generalizations, dependencies, etc. Some common types of class diagrams are: domain model diagram, diagram of implementation classes. Object diagram could be considered as instance level class diagram which shows instance specifications of classes and interfaces (objects), slots with value specifications, and links (instances of association)" (<https://www.uml-diagrams.org/class-diagrams-overview.html>). Esempi di "class diagram": <https://www.uml-diagrams.org/examples/hospital-domain-diagram.html?context=cls-examples>.

L'aggiunta del voto può essere rappresentata in UML sotto diversi punti di vista e in forma di altri tipi di diagrammi. Per esempio, per descrivere l'insieme di operazioni che il professore deve effettuare per mettere un voto sul registro elettronico, si può realizzare un "activity diagram",⁸ che rappresenta la sequenza delle operazioni che si verificano all'interno di un sistema per svolgere una certa attività.

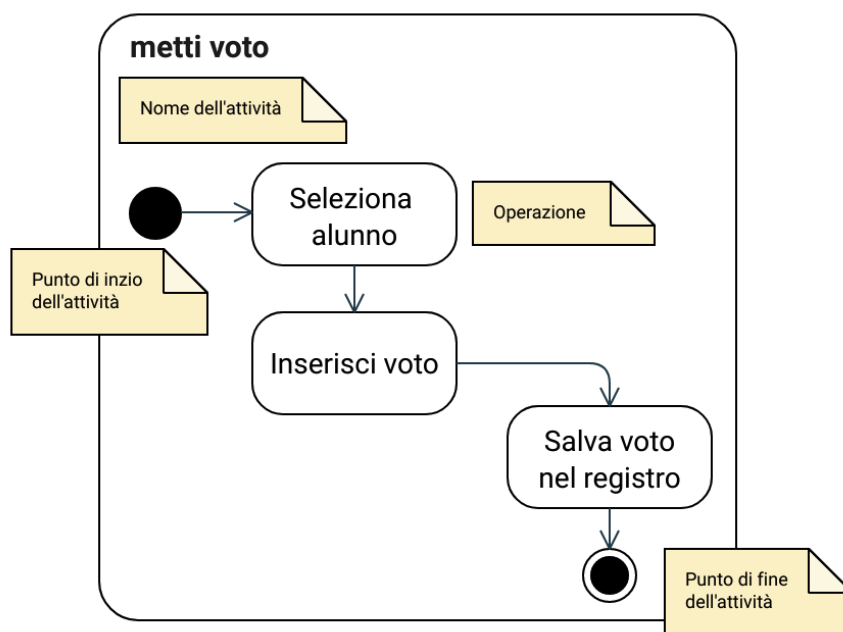


Figura 4.65 Activity diagram dell'operazione "metti voto".

UML permette di realizzare diverse tipologie di modelli e di rispettivi diagrammi,⁹ fornendo un'ampia gamma di elementi grafici.

"A UML model consists of elements such as packages, classes, and associations. The corresponding UML diagrams are graphical representations of parts of the UML model. UML diagrams contain graphical elements (nodes connected by paths) that represent elements in the UML model" (UML v. 2.5.1, p. 725).

⁸ Cfr. "Activity diagram is UML behavior diagram which shows flow of control or object flow with emphasis on the sequence and conditions of the flow. The actions coordinated by activity models can be initiated because other actions finish executing, because objects and data become available, or because some events external to the flow occur. The following nodes and edges are typically drawn on UML activity diagrams: activity, partition, action, object, control, activity edge" (<https://www.uml-diagrams.org/activity-diagrams.html>). Esempi di *activity diagram*: <https://www.uml-diagrams.org/activity-diagrams-examples.html>.

⁹ Le diverse tipologie di diagramma previste in UML sono: activity diagram; class diagram; communication diagram; component diagram; composite structure diagram; deployment diagram; interaction overview diagram; object diagram; package diagram; profile diagram; state machine diagram; sequence diagram; timing diagram; use case diagram (cfr. UML v. 2.5.1, p. 727).

Per distinguere le tipologie di diagrammi si fa riferimento alla tipologia di elemento maggiormente rappresentato al loro interno. Per esempio, un diagramma è detto “class diagram” quando rappresenta prevalentemente delle classi e la loro struttura.¹⁰

In generale, si possono distinguere due principali categorie di modelli realizzati in UML:

- i modelli strutturali, che rappresentano un sistema informatico e gli elementi al suo interno sincronicamente, quindi come sono in un determinato momento temporale;
- i modelli comportamentali, che invece rappresentano un sistema informatico e gli elementi al suo interno diacronicamente, ovvero come cambiano e interagiscono nel corso del tempo.¹¹

È possibile realizzare modelli UML che rappresentino un sistema informatico secondo entrambe le prospettive, quella sincronica e quella diacronica. Tuttavia, per praticità e chiarezza, è preferibile realizzare dei modelli di dimensioni contenute e che adottino una particolare prospettiva non solo temporale ma anche sul modo di descrivere il sistema informatico.

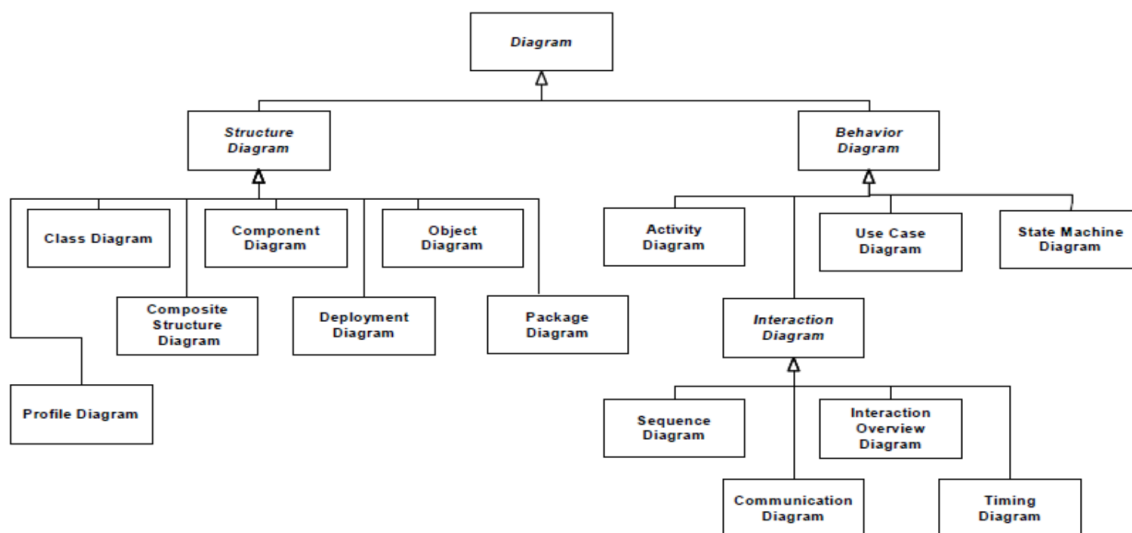


Figura 4.66 Tassonomia dei diagrammi strutturali e comportamentali (UML v. 2.5.1, p. 685).

¹⁰ Cfr. “The diagram contents area contains the graphical symbols; the primary graphical symbols define the type of the diagram (e.g., a class diagram is a diagram where the primary symbols in the contents area are class symbols).” (UML v. 2.5.1, p. 725).

¹¹ Cfr. “The general division of UML modeling constructs into two semantic categories: Structural Semantics defines the meaning of UML structural model elements about individuals in the domain being modeled, which may be true at some specific point in time. [...] Behavioral Semantics defines the meaning of UML behavioral model elements that make statements about how individuals in the domain being modeled change over time.” (UML v. 2.5.1, p. 55).

Questa sezione rappresenta solo un'infarinatura dei concetti alla base di UML e della programmazione a oggetti. Nel corso del resto del capitolo, verranno introdotti al bisogno gli ulteriori concetti utilizzati.

4.3 La “traduzione” dell'ontologia in *class diagram*

Il modello concettuale di esempio può essere suddiviso in tre parti distinte: i concetti basilari, per inquadrare in termini astratti il concetto di edizione scientifica e nello specifico di edizione critica; i concetti per descrivere nel dettaglio l'allestimento del testo critico e le informazioni che l'editore può includere in un'edizione per descrivere il lavoro filologico svolto; infine i componenti che costituiscono l'edizione scientifica, i diversi contenuti del prodotto editoriale finito. Quest'ultima prospettiva è quella da adottare per proseguire il lavoro di modellizzazione da un punto di vista implementativo, facendo coincidere i componenti definiti nell'ontologia con i componenti che costituiscono l'interfaccia grafica dello strumento di visualizzazione.

Nel corso del capitolo 3, per illustrare i concetti e le relazioni del dominio sono stati adoperati degli schemi grafici. Tali schemi grafici non seguono una notazione standard, ma sono stati ideati *ad hoc*, sulla base dei diagrammi UML abbozzati con l'*editor* grafico utilizzato per sviluppare l'ontologia. Come già illustrato (3.2), infatti, il software OWLGrEd permette di realizzare ontologie OWL in forma di diagrammi UML.

Per esplicitare le differenze tra gli schemi da me proposti e i diagrammi generati dall'*editor* OWLGrEd, si prenda ad esempio la porzione dell'ontologia dedicata ai concetti di lettore (classe `Reader`), pubblicazione dell'edizione scientifica (`Scholarly Edition Publication`), edizione scientifica ed edizione critica. Il relativo schema da me realizzato si presenta come in figura.

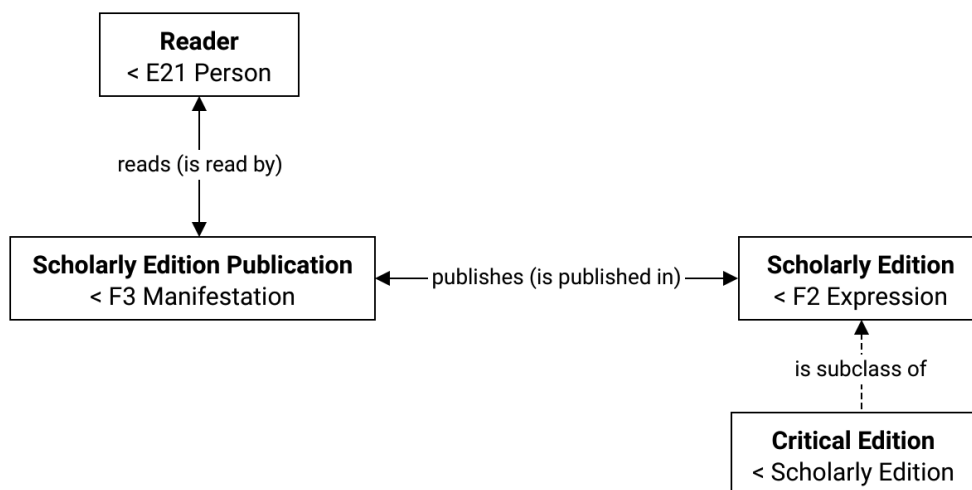


Figura 4.67 Schema che rappresenta la parte dell'ontologia di esempio che descrive il rapporto tra l'edizione scientifica e il lettore.

Nel software OWLGrEd la stessa porzione dell'ontologia è presentata in modo molto simile: si può osservare, infatti, come in entrambe le figure le classi dell'ontologia siano formattate graficamente come dei rettangoli, cioè con la notazione di "Classe" del linguaggio UML.

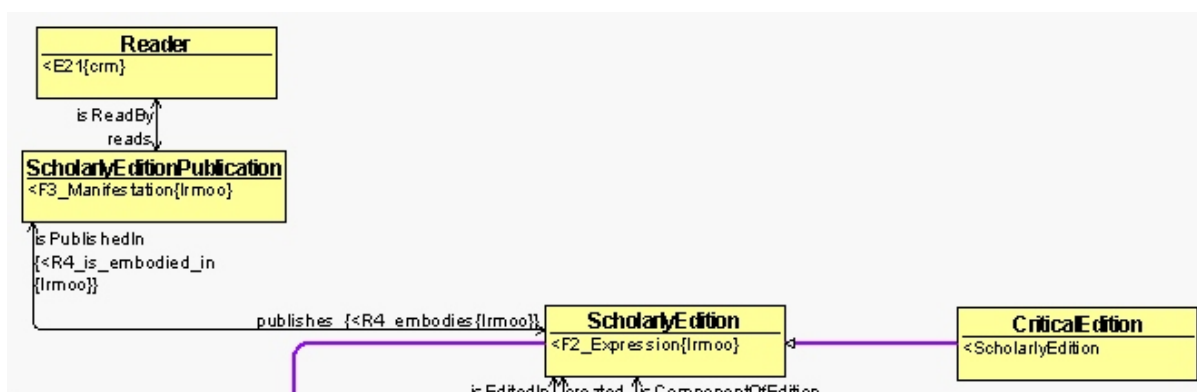


Figura 4.68 Diagramma UML generato con OWLGrEd della parte dell'ontologia di esempio che descrive il rapporto tra l'edizione scientifica e il lettore.

Grazie all'utilizzo del software OWLGrEd è stato possibile non solo scrivere l'ontologia in OWL ma anche abbozzare dei diagrammi UML che descrivono il dominio con una notazione più familiare agli sviluppatori informatici. Si tratta solo di bozze, in quanto alcuni dei rapporti rappresentati nell'ontologia con delle proprietà definite *ad hoc*, in UML possono essere rese in forma più sintetica utilizzando degli elementi equivalenti da un punto di vista concettuale. Per esempio, la proprietà `introduction has part (is part of introduction)`, che associa la classe `Introduction` alla classe `Introduction Part`, a livello concettuale stabilisce che un'introduzione è

costituita da un insieme di parti. In UML questo rapporto può essere rappresentato con il concetto di “composizione”, un particolare tipo di associazione tra elementi (per i dettagli si rimanda a 4.3.2).

Il primo passo per guidare lo sviluppo di uno strumento informatico, dunque, consiste nel “tradurre” i componenti di un’edizione scientifica digitale, descritti nel modello concettuale, in *class diagram* UML.¹² In questo modo si facilita la comprensione del modello concettuale da parte di uno sviluppatore informatico e, al contempo, si offre una base sulla quale poter avviare una progettazione più dettagliata da un punto di vista implementativo. Nella prossima sezione, si illustra come gli stessi *class diagram* possono essere utilizzati come base per specificare la struttura e le funzionalità di uno strumento di visualizzazione.

In questa sezione non si presenta una rassegna dettagliata di *class diagram* derivati dal modello concettuale, ma si indicano delle strategie di massima per realizzare la “traduzione”.

4.3.1 Scholarly Edition Component e i rapporti di parentela

Il concetto di “componente dell’edizione scientifica” che nel modello concettuale è rappresentato dalla classe `Scholarly Edition Component`, è il punto di partenza per la modellizzazione in UML del sistema informatico, in quanto rappresenta ogni contenuto di un’edizione scientifica: il testo edito, l’apparato critico, ma anche la riproduzione fotografica di un testimone, uno stemma codicum, ecc. La classe ontologica `Scholarly Edition Component` comprende oggetti di natura diversa, accomunati dall’essere parte di un’edizione scientifica e dal poter essere considerati un tipo di `lrmoo:F2 Expression`, cioè una forma di espressione.

A livello implementativo è opportuno mantenere i rapporti di parentela definiti tra le classi del modello concettuale, in modo tale da sfruttare il meccanismo dell’ereditarietà (in inglese *inheritance*). Riproponendo i medesimi rapporti “classe-sottoclasse” in fase di implementazione, è possibile riutilizzare le operazioni delle superclassi: l’ereditarietà è, infatti, uno dei meccanismi più utilizzati nella programmazione a oggetti per riusare codice ed è detta anche *white-box reuse*. Le sottoclassi possono estendere le proprie superclassi aggiungendo attributi e operazioni. Inoltre, nel caso

¹² I diagrammi UML proposti nel presente elaborato sono stati realizzati con l’applicativo web per la progettazione grafica Moqups (<https://moqups.com/it/>), in quanto consente di ottenere dei diagrammi con una risoluzione migliore rispetto al software OWLGrEd.

in cui una sottoclasse necessita di modificare il modo in cui un'operazione è stata implementata è possibile effettuare un *override*,¹³ cioè la sottoclasse sovrascrive l'operazione ereditata dalla superclasse, definendo un'operazione omonima e che prevede gli stessi parametri (in linguaggio tecnico si dice che abbia la stessa firma o *signature*), ma con un'implementazione diversa.

Per esempio la classe `Apparatus Entry` definisce un'operazione (chiamata ipoteticamente "refersTo") per collegarsi al proprio *target*, ovvero un'istanza della classe `Edited Text Passage`. La classe `Critical Apparatus Entry`, essendo sottoclasse di `Apparatus Entry`, potrà riprendere la stessa operazione "refersTo" e modificarla affinché recuperi un'istanza di `Critical Text Passage`.

L'utilizzo dell'ereditarietà come meccanismo per il riuso di codice presenta dei limiti. In particolare, rende più difficile testare il codice e modificare separatamente la sottoclasse e la superclasse. Se delle modifiche al modello dovessero richiedere di cambiare l'implementazione di una superclasse, di conseguenza anche la sottoclasse andrebbe aggiornata. Per questo motivo, è preferibile servirsi il più possibile di un altro meccanismo per il riuso di codice, chiamato composizione, presentato nella prossima sezione.

La classe `Scholarly Edition Component`, tecnicamente, non serve per creare degli oggetti che diventino poi parte dell'interfaccia grafica, ma a definire delle caratteristiche comuni alle diverse tipologie di componenti. Allo stesso modo, la classe `Accompanying Material` serve a stabilire che cosa caratterizza i diversi tipi di materiali che accompagnano un testo edito. Per questo motivo, può essere più opportuno implementare `Scholarly Edition Component` e `Accompanying Material` non come classi ma come classi astratte oppure come interfacce. Una classe astratta è una classe che non può essere istanziata, dalla quale cioè non è possibile derivare oggetti, e che viene utilizzata per rappresentare delle superclassi. Un'interfaccia (in inglese *interface*) è sempre un tipo di "classifier" che definisce una serie di operazioni. Le classi che realizzano l'interfaccia devono riportare e implementare le operazioni dichiarate nell'interfaccia.

"An Interface is a kind of Classifier that represents a declaration of a set of public Features and obligations that together constitute a coherent service. An Interface specifies a contract; any instance of a Classifier that realizes the Interface shall fulfill that contract. The obligations associated with an

¹³ Si precisa che non tutti i linguaggi *object-oriented* prevedono il meccanismo dell'*override*.

Interface are in the form of constraints (such as pre- and postconditions) or protocol specifications, which may impose ordering restrictions on interactions through the Interface. Interfaces may not be instantiated. Instead, an Interface specification is implemented or realized by a BehavioredClassifier, which means that the BehavioredClassifier presents a public facade that conforms to the Interface specification” (UML v. 2.5.1, p. 213, mia enfasi)

Le principali differenze tra una classe astratta e un’interfaccia, in un linguaggio di programmazione come Java, sono:

- le classi astratte possono avere attributi, costanti (attributi il cui valore non può essere modificato), metodi definiti (che indicano concretamente le operazioni da svolgere) e non definiti, mentre le interfacce possono avere solo costanti e metodi non definiti;
- i metodi e le costanti di un’interfaccia devono essere pubbliche (ovvero visibili da tutte le altre classi), mentre una classe astratta permette di definire metodi e attributi (o costanti) con visibilità diverse, in modo tale da nasconderle alle altre classi;
- una classe può estendere una sola classe (astratta o concreta), mentre può estendere più interfacce senza limiti.

Quest’ultimo aspetto è particolarmente rilevante, perché a livello di ontologia è possibile che una classe abbia più superclassi. Nel modello concettuale di esempio *Stemma Codicume Witnesses List* sono sottoclassi sia di *Introduction Part* sia di *Accompanying Material*. Per questo motivo, si propone di tradurre *Scholarly Edition Component* e *Accompanying Material* come interfacce piuttosto che come classi astratte, ma la scelta tra le due implementazioni dipende molto dalle caratteristiche dei metodi da associare alle due classi (ovvero le loro funzionalità) e da come conviene strutturare e riusare il codice, oltre che dalle caratteristiche del linguaggio di programmazione scelto. I concetti alla base di un’edizione scientifica nel modello concettuale, in UML possono essere modellati come in figura.

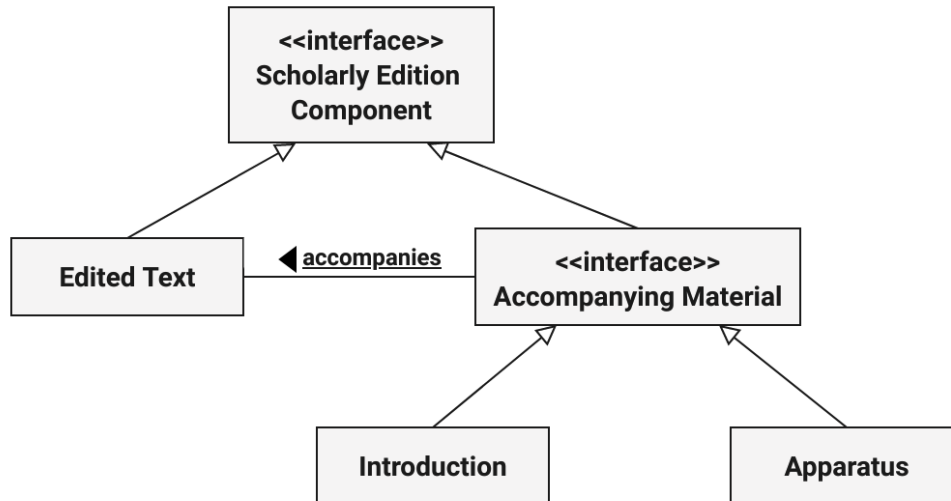


Figura 4.69 Class diagram dei principali componenti alla base di un'edizione scientifica digitale.

Nella notazione UML un'interfaccia è rappresentata come una classe, ovvero come un rettangolo, all'interno del quale ci sono un'intestazione con il nome dell'interfaccia, preceduto però dalla dicitura “<<interface>>”. I collegamenti in UML sono rappresentati dall'elemento “Association”:

“An Association classifies a set of links, each of which is an instance of the Association (UML v. 2.5.1, p. 261)

Un'associazione è rappresentata in UML da una freccia. Diverse tipologie di associazione, sono rese con frecce diverse. La freccia con la punta bianca, per esempio, indica un tipo di associazione tra classi chiamato “generalizzazione”,¹⁴ che significa banalmente che una classe è sottoclasse di un'altra. La proprietà “accompanies (is accompanied by)” dell'ontologia è stata trasformata in un'associazione equivalente. Il triangolino accanto all'etichetta dell'associazione indica il senso di lettura, quindi la relazione si legge: “Accompanying Material accompanies Edited Text”.

4.3.2 Il rapporto “parte-intero”

Nel modello concettuale di esempio ci sono diverse proprietà che esprimono una relazione parte-intero tra due classi, per esempio tra l'introduzione (“Introduction”) e le sue parti (“Introduction Part”), tra il testo edito (“Edited Text”) e un luogo al suo interno (“Edited Text Passage”), tra l'ap-

¹⁴ Cfr. “A Generalization is a taxonomic relationship between a more general Classifier and a more specific Classifier. Each instance of the specific Classifier is also an instance of the general Classifier. The specific Classifier inherits the features of the more general Classifier. A Generalization is owned by the specific Classifier.” (UML v. 2.5.1, p. 180).

parato (“Apparatus”) e una voce d’apparato (“Apparatus Entry”), ecc. Nella notazione UML il rapporto “parte-intero”, quindi tra un elemento e un altro elemento che ne costituisce una parte, è rappresentato attraverso un collegamento chiamato “aggregazione”:

“Sometimes a Property is used to model circumstances in which one instance is used to group together a set of instances; this is called aggregation. To represent such circumstances, a Property has an aggregation property, of type AggregationKind; the instance representing the whole group is classified by the owner of the Property, and the instances representing the grouped individuals are classified by the type of the Property. AggregationKind is an enumeration with the following literal values:

- **none** Indicates that the Property has no aggregation semantics.
- **shared** Indicates that the Property has shared aggregation semantics. Precise semantics of shared aggregation varies by application area and modeler.
- **composite** Indicates that the Property is aggregated compositely, i.e., the composite object has responsibility for the existence and storage of the composed objects.

Composite aggregation is a strong form of aggregation that requires a part object be included in at most one composite object at a time. If a composite object is deleted, all of its part instances that are objects are deleted with it.” (UML v. 2.5.1, p. 154)

Prima di stabilire come tradurre questi rapporti parte-intero dell’ontologia di esempio, occorre interrogarsi su che cosa un’aggregazione o una composizione comportino a livello di organizzazione del codice.

L’aggregazione e la composizione sono come l’ereditarietà dei meccanismi per riusare codice. L’ereditarietà è detta *white-box reuse*, in quanto una classe può riusare i metodi definiti nella propria superclasse, implementandoli o modificandoli a piacere tramite l’*override*. L’aggregazione e la composizione sono dette, invece, *black-box reuse* perché consentono a una classe di riutilizzare solo i metodi pubblici di un’altra classe, senza sapere come sono stati implementati e senza poterli modificare. Inoltre il meccanismo di ereditarietà fa sì che quando viene invocato un metodo di una superclasse, le sottoclassi possano rispondere, per cui l’operazione invocata è quella definita nella superclasse, ma viene eseguita secondo l’implementazione (quindi l’insieme concreto di azioni) della sottoclasse che ha risposto alla chiamata.

La differenza principale tra aggregazione e composizione riguarda il livello di dipendenza reciproca tra le due classi. Un’aggregazione tra due classi implica che una classe costituisce l’altra, ma

non necessita dell'altra classe per esistere. Riprendendo l'esempio del registro elettronico, si può dire che il rapporto tra la classe "Alunno" e la classe "Gruppo Classe" sia un'aggregazione: l'alunno può essere liberamente spostato da un gruppo classe a un altro, al contempo, se un gruppo classe viene cancellato, gli alunni possono essere assegnati a dei nuovi gruppi classe. In una composizione, invece, una classe costituisce un'altra classe e da essa dipende per esistere.

Per spiegare meglio il concetto di composizione, si propone un esempio tratto dall'ontologia. Si pensi al rapporto tra il testo edito (`Edited Text`) e i luoghi testuali contenuti al suo interno, rappresentati da `Edited Text Passage`, così come il rapporto tra `Critical Text` e `Critical Text Passage`. Un luogo testuale è parte integrante del testo e non può essere presentato senza di esso, perché perderebbe di senso. Dunque, in questo caso il rapporto tra il testo e il luogo del testo è traducibile come una composizione.

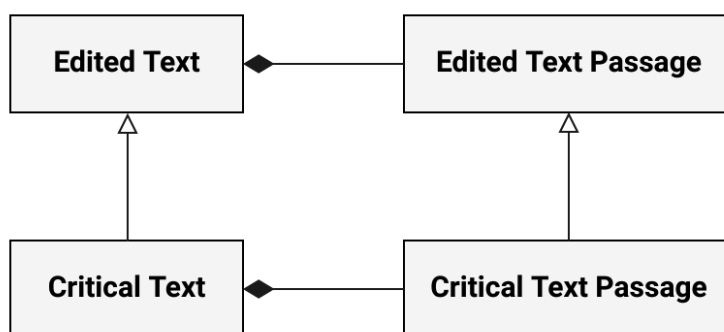


Figura 4.70 Class diagram che rappresenta il rapporto di composizione tra il testo edito e un passo del testo edito così come tra le loro rispettive sottoclassi.

Nella notazione UML la composizione è rappresentata da una linea che unisce i due elementi e che nel verso dell'elemento che rappresenta l'intero presenta un rombo nero.

Proseguendo con gli altri rapporti parte-intero presenti nel modello si può dire che il rapporto tra l'introduzione (`Introduction`) e le sue parti (`Introduction Part`) può essere tradotto come una composizione. `Witnesses List` e `Stemma Codicum` possono essere implementate indipendentemente da `Introduction` oppure come sottoclassi di `Introduction Part`. In questo secondo caso, il rapporto con l'introduzione è nuovamente traducibile come una composizione.

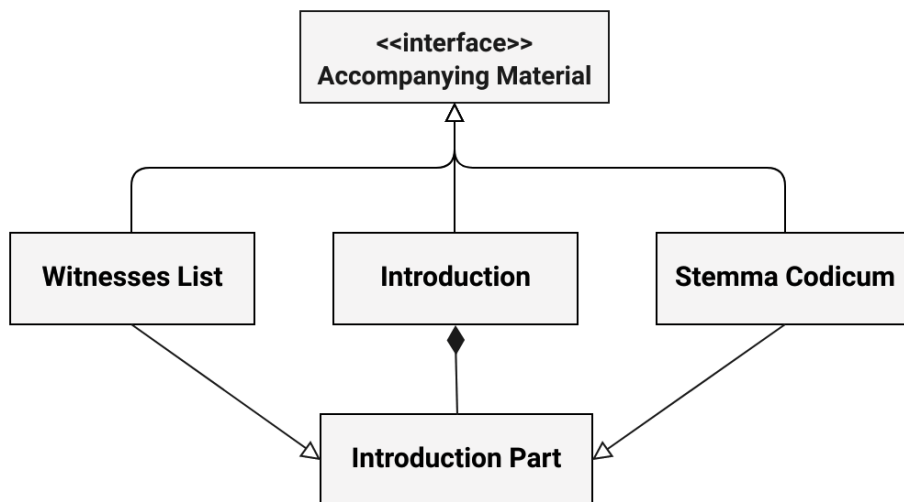


Figura 4.71 Class diagram che rappresenta i rapporti tra l'introduzione, le sue parti, la lista dei testimoni e lo stemma codicum.

Per tradurre in UML il rapporto tra un apparato e le sue voci, rappresentato dalla proprietà `has entry` (`is entry of`) occorrono alcune considerazioni. Nella definizione di composizione sopra riportata si legge:

“Composite aggregation is a strong form of aggregation that requires a part object be included in at most one composite object at a time. If a composite object is deleted, all of its part instances that are objects are deleted with it”.

Nel caso di un'edizione scientifica digitale, molto spesso (per non dire quasi sempre), le voci d'apparato sono presentate all'interno del testo critico in forma di finestra *pop-up* o pannello a scomparsa. A livello logico ogni voce d'apparato fa parte di un solo apparato e questo vale anche per tutte le sottoclassi di `Apparatus Entry` e di `Apparatus`, quindi le note di commento filologico, le voci dell'apparato critico, le voci dell'apparato delle fonti, ecc. Però a livello implementativo non conviene che il rapporto tra un apparato e le sue voci sia fortemente vincolante. Se le voci d'apparato sono presentate e gestite nell'apparato, allora la proprietà `has entry` (`is entry of`) può essere tradotta come una composizione, altrimenti conviene utilizzare un'aggregazione. Nella notazione UML l'aggregazione è rappresentata con una freccia uguale a quella della composizione, ma con il rombo bianco.

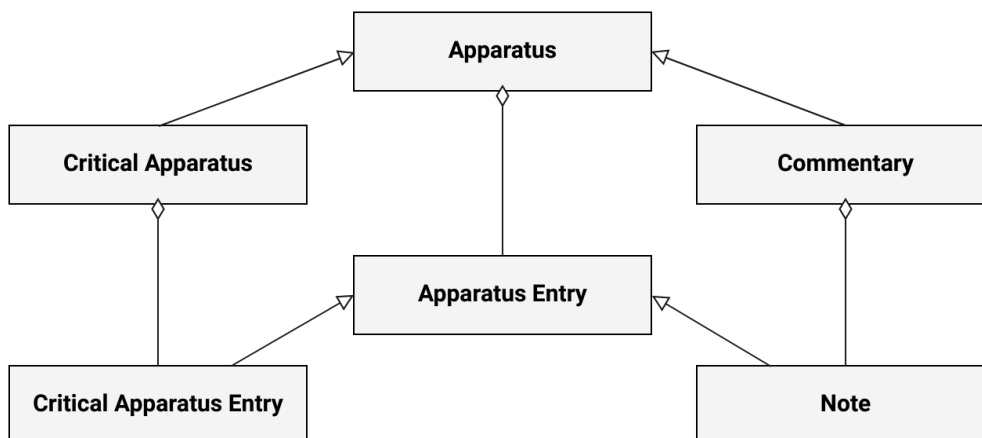


Figura 4.72 Class diagram che rappresenta il rapporto di aggregazione tra un apparato e una voce d'apparato così come tra le rispettive sottoclassi.

La composizione, come modo per descrivere il rapporto parte-intero, può essere utilizzata anche per definire a livello implementativo una fascia dell'apparato critico e un gruppo di lezioni all'interno di una voce dell'apparato critico, ricordando però come la suddivisione in fasce d'apparato e, soprattutto, in gruppi di lezioni possa essere facoltativa. La classe `Critical Apparatus` è composta dalla classe `Critical Apparatus Group`, a sua volta composta dalla classe `Critical Apparatus Entry`. In modo analogo, la classe `Critical Apparatus Entry` è composta dalla classe `Reading Group`, a sua volta composta dalla classe `Reading In Apparatus`.

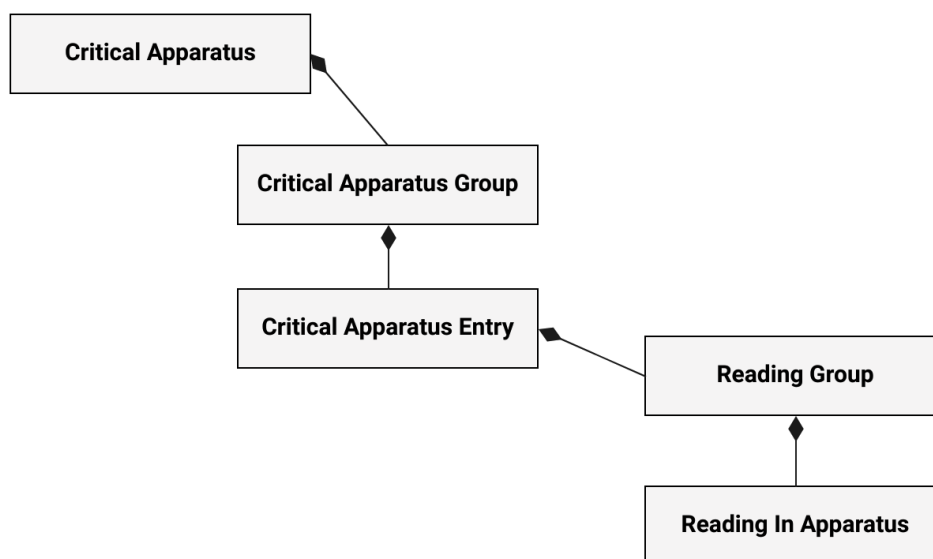


Figura 4.73 Class diagram che descrive i rapporti di composizioni tra gli elementi che possono strutturare l'apparato critico e una sua voce.

4.3.3 Particolarità dell'apparato

Nel modello concettuale, la voce d'apparato è un'annotazione, caratterizzata da un contenuto testuale (*Apparatus Entry Content*) e da un *target*, ovvero un luogo del testo edito (*Edited Text Passage*). Il luogo del testo edito è individuabile mediante un selettore (*Selector*), che può essere presentato anche nel contenuto della voce d'apparato, per esplicitare il collegamento al luogo del testo.

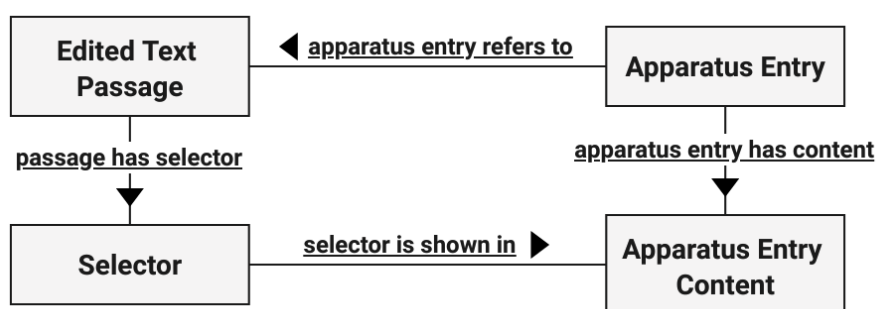


Figura 4.74 Class diagram che indica i rapporti tra una voce d'apparato e il luogo nel testo cui è riferita.

I contenuti di una voce dell'apparato critico sono principalmente le lezioni (*Reading In Apparatus*), quella accolta o stabilita congetturalmente nel testo critico e quelle rifiutate. Le lezioni possono essere organizzate in gruppi (*Reading Group*) e avere collegata una nota dell'editore (*Critical Apparatus Entry Content*).

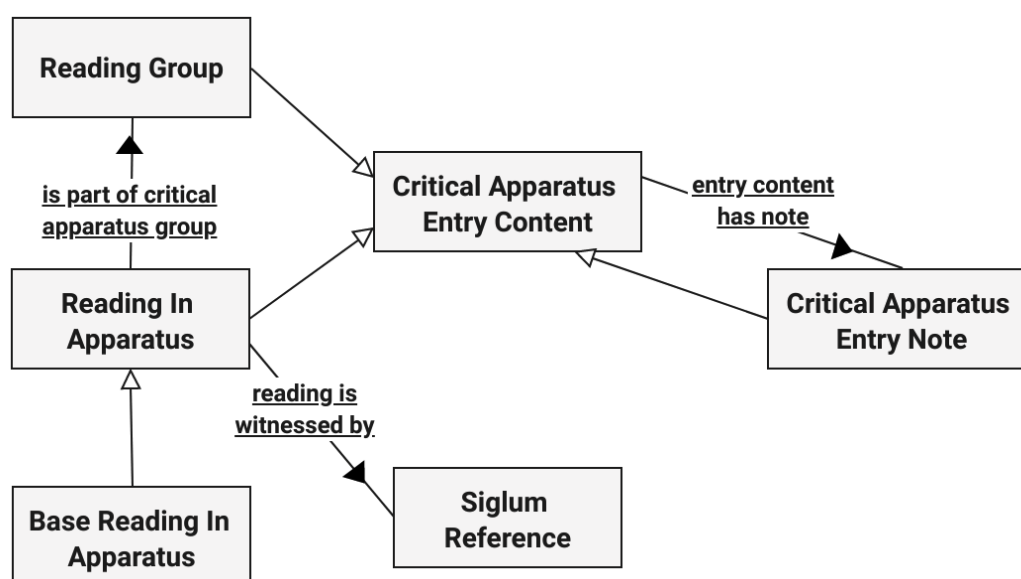


Figura 4.75 Class diagram che illustra i rapporti tra le lezioni presentate all'interno di una voce dell'apparato critico e le relative strutture.

4.3.4 La traduzione delle proprietà in attributo, funzionalità o qualità grafica

Nel modello concettuale, le proprietà definite esprimono diversi tipi di rapporto tra le classi, che a livello implementativo possono essere tradotte in modo diverso. Di base, le proprietà possono essere rese in UML semplicemente come associazioni rappresentate da frecce. Un'associazione in UML esprime in termini di sviluppo del codice ciò che può essere definito come un rapporto di dipendenza tra due classi, per cui una classe si serve di un'altra classe. Sulla base delle associazioni presenti nella traduzione in UML del modello, lo sviluppatore può progettare lo strumento informatico nel modo che ritiene più opportuno, purché le implementazioni siano coerenti con le relazioni logiche espresse dal modello concettuale.

Alcune proprietà si prestano a essere tradotte come degli attributi della classe cui sono associate (la classe “dominio” della proprietà, ovvero la classe da cui origina la freccia che rappresenta la proprietà). La classe `Reading In Apparatus`, per esempio, ha associate una serie di proprietà riflessive (ovvero che collegano un'istanza di `Reading In Apparatus` a un'altra istanza di `Reading In Apparatus`), che permettono all'editore di esplicitare il rapporto tra le diverse lezioni individuate nella tradizione testuale: `was revision of`, `was corrected to`, `is variant of`, `is formal variant of`, ecc. Queste proprietà potrebbero essere tradotte come degli attributi della classe `Reading In Apparatus`.

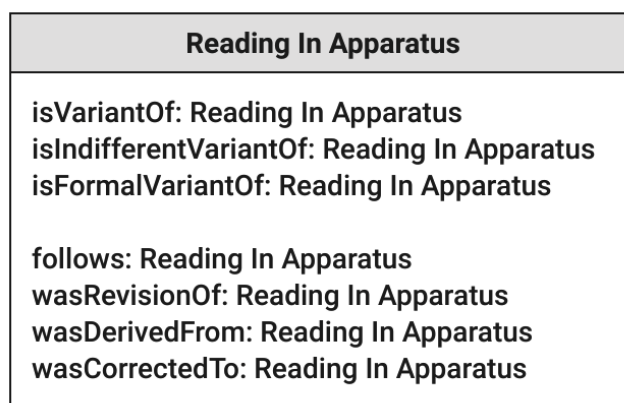


Figura 4.76 Class diagram della classe `Reading In Apparatus` e dei relativi attributi.

Le stesse proprietà possono essere utilizzate come criteri per suddividere l'apparato critico in fasce d'apparato oppure per raggruppare le lezioni all'interno di una voce dell'apparato critico. Al contempo, le proprietà che si vedono in figura possono essere utilizzate come parametri per progettare

delle funzionalità. Per esempio, in un'edizione di un'opera tramandata da un elevato numero di testimoni, una funzionalità di filtro potrebbe consentire all'utente di selezionare una lezione e mostrare nella voce dell'apparato critico solo le altre lezioni che sono varianti formali di quella selezionata.

Un'altra proprietà che si presta a essere tradotta come attributo della classe `Reading In Apparatus` è `reading is witnessed by`, che associa la classe `Reading In Apparatus` alla classe `Siglum Reference`.

La proprietà `refers to siglum`, che associa le classi `Siglum Reference` e `Siglum`, potrebbe essere tradotta, invece, come operazione della classe `Siglum Reference`. Immaginiamo che lo sviluppatore voglia fare in modo che, dalla voce dell'apparato critico, cliccando sulla sigla di un testimone, il lettore possa visualizzare la sigla del testimone nell'elenco dei testimoni. In questo caso la funzionalità può essere implementata come un'operazione dedicata, `refers to siglum`, che richiede in input la sigla del testimone.

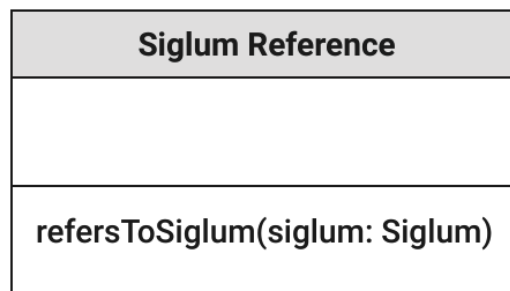


Figura 4.77 Class diagram della classe Siglum Reference.

Infine, nel modello concettuale ci sono proprietà dal significato più generico e sfumato, che non necessariamente devono essere tradotte in un attributo o in un'operazione, ma che in qualche modo devono essere tenute in considerazione nel progettare come lo strumento informatico si presenta e funziona. Per esempio, la proprietà `accompanies (is accompanied by)`, che associa le classi `Edited Text` e `Accompanying Material`, indica (in modo quasi lapalissiano) che il materiale di corredo accompagna il testo edito, quindi sottolinea come un materiale di corredo dipenda da un testo edito. Per sviluppare lo strumento informatico in modo coerente con il modello concettuale, il materiale di corredo può essere presentato nell'interfaccia grafica insieme al testo edito, magari posizionato al di sotto di esso, emulando la *mise en page* delle edizioni a stampa. Oppure il materiale di

corredo potrebbe essere direttamente accessibile dal testo edito: cliccando su un passo del testo edito, per esempio, compare in un pannello laterale o all'interno del testo la voce d'apparato. Il modo effettivo in cui lo sviluppatore implementa il rapporto tra il testo edito e i materiali di corredo è secondario, a condizione che l'utente percepisca in modo chiaro la differenza tra testo edito e materiale di corredo e che un certo materiale di corredo accompagni un determinato testo edito.

Da questi primi esempi di possibili implementazioni, emerge come l'analisi delle proprietà definite nell'ontologia e del loro significato costituisca un punto fondamentale per la progettazione dello strumento informatico.

Rispetto a quanto riportato finora come esempio, si può notare come le proprietà definite nell'ontologia traducibili in UML come rapporto parte-intero, quindi come una composizione o, più in generale, come un'aggregazione, siano spesso nella forma "classe X ha classe Y (classe Y è Y di X)" (es. `has apparatus (is apparatus of)`). Le proprietà traducibili come attributi, invece, sono articolate perlopiù in forma di predicato nominale (es. `is variant of`) o di predicato verbale in forma passiva (es. `reading is witnessed by`). Mentre le proprietà che possono denotare un'operazione o una funzionalità tendono a essere dei predicati verbali in forma attiva (es. `refers to siglum`).

Data l'importanza delle definizioni delle proprietà, in una futura revisione del modello concettuale sarà necessario valutarne accuratamente l'efficacia e se e come correggerle, in modo tale da guidare in modo più chiaro e preciso lo sviluppo informatico.

4.4 La progettazione dello strumento informatico

I *class diagram* presentati nella sezione precedente descrivono in notazione UML i principali oggetti e le relazioni reciproche del dominio applicativo di uno strumento informatico per la visualizzazione di un'edizione scientifica digitale. Per progettare nel dettaglio come lo strumento debba essere sviluppato è necessario analizzare le classi e le proprietà definite nel modello concettuale. Il punto di partenza per la progettazione consiste, però, nel mettere a fuoco quali sono a livello macroscopico i compiti che uno strumento di visualizzazione deve svolgere. I compiti principali sono tre: il recupero dell'edizione scientifica e dei suoi contenuti; l'inserimento dei contenuti nell'interfaccia grafica; la

gestione dell'interazione dell'utente con i contenuti. Questi compiti coinvolgono tutti i componenti di un'edizione scientifica digitale. Per cui, prima di addentrarsi nell'implementazione dei singoli componenti, occorre stabilire come svolgere questi tre compiti.

4.4.1 I compiti principali di uno strumento di visualizzazione

Il componente dell'edizione scientifica è un contenuto preparato dell'editore, che può essere:

- una stringa di testo;
- un'immagine;
- un oggetto multimediale;
- una collezione di dati.

Per comprendere in che modo la classe `Scholarly Edition Component` possa essere tracciata in un diagramma UML, bisogna analizzare le sue caratteristiche.

Prima di tutto, dato che lo strumento informatico che si intende modellizzare è uno strumento di sola visualizzazione, si presuppone che i suoi componenti non siano già al suo interno, ma che debbano essere in qualche modo recuperati. L'operazione principale che il componente di un'edizione scientifica deve svolgere, dunque, è il recupero dei contenuti: tale operazione si può denominare banalmente "retrieve".

Per individuare correttamente il componente da recuperare, occorrono delle informazioni. Prima di tutto si deve conoscere dove è collocato il file sorgente o un altro tipo di risorsa che costituisce il componente o che lo contiene. Se questa risorsa è un file locale, per esempio, è sufficiente il percorso al file (*file path*). Se si tratta di una risorsa web accessibile online, occorre l'indirizzo URL. Se, invece, il componente è salvato all'interno di un database locale o accessibile da remoto, bisogna poter lanciare una ricerca (*query*) all'interno del database, eventualmente mediante una API. A prescindere dal punto in cui si trova il componente, questa informazione può essere descritta come un parametro della funzionalità `retrieve` di `Scholarly Edition Component`, denominato "sourceLocation". Un'altra informazione fondamentale per il corretto recupero del componente è il formato del file sorgente o della risorsa. Questa informazione potrebbe essere salvata nel parametro "sourceType" ed espressa specificando il *media type* o *MIME type*:

“A media type (also known as a Multipurpose Internet Mail Extensions or MIME type) indicates the nature and format of a document, file, or assortment of bytes.”¹⁵

Infine, occorre una terza informazione nel caso in cui il componente corrisponda, a livello logico, solo a una parte della risorsa o file sorgente. Si pensi a un’edizione scientifica marcata in XML/TEI: è prassi comune marcare all’interno di un unico documento il testo, l’apparato critico, l’introduzione, ecc. Se il componente è il testo, occorre conoscere dove il testo inizia e dove finisce all’interno del documento, avere cioè delle coordinate per recuperare solo la porzione di documento che effettivamente è il componente. Se il file sorgente (o risorsa) viene recuperato dallo strumento informatico in forma di flusso di dati (in termine tecnico uno *stream*), le coordinate sono date da due numeri: il numero che rappresenta il punto della risorsa in cui inizia il contenuto del componente e il numero corrispondente al punto in cui finisce. Questa informazione può essere denominata “component-PositionInSource”.

A seconda di queste tre informazioni, cambia il modo in cui il componente deve essere recuperato. Il componente può essere per esempio un’immagine accessibile online mediante IIIF oppure un documento XML/TEI locale. L’operazione “retrieve” può essere scritta, in termini di codice, in modo diverso in base alle caratteristiche del file sorgente del componente e alle tecnologie dello strumento informatico.

Se `Scholarly Edition Component` venisse implementata come un’interfaccia, caratterizzata dall’operazione “retrieve” e dai relativi parametri sopra descritti, in UML potrebbe essere rappresentata mediante un *class diagram* come segue.

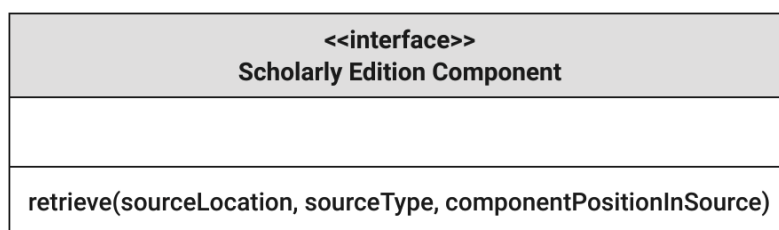


Figura 4.78 Diagramma UML dell’interfaccia *Scholarly Edition Component* e del metodo *retrieve*.

Una volta recuperato il componente dalla sua risorsa, lo strumento di visualizzazione deve integrare il componente nell’interfaccia grafica, adattandolo alle tecnologie dello strumento. Per spiegare

¹⁵ https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types.

che cosa si intende, si offre un esempio. Si immagini di dover recuperare un testo edito, un componimento poetico, allestito dall'editore in un file XML/TEI (il testo edito di esempio è tratto dalle linee guida TEI).¹⁶ La strofa è racchiusa nell'elemento <lg> (*line group*), mentre i singoli versi in altrettanti <l> (*line*). All'interno del testo, si notano degli elementi <emph> (*emphasis*) che indicano delle parole marcate per porvi un' enfasi linguistica o retorica, solitamente resa nelle edizioni a stampa con il corsivo o con il grassetto.

```
<lg>
<l>What projects men make—what queer turns they take,</l>
<l>Since <emph>steam</emph> has improved our condition;</l>
<l>They never are still, but must cure or must kill</l>
<l>With steam physic or steam ammunition.</l>
<l>But a short time ago, to a quack you would go,</l>
<l>To steam a fat man to a thinner;</l>
<l>Now changed from all that, if you wish to get <emph>fat</emph>,</l>
<l>Come to Barton's and eat a <emph>steam dinner!</emph>
</l>
<l>Oh dear! think of a scheme, odd though it seem—</l>
<l>I'm sure 'twill succeed if you make it by steam.</l>
</lg>
```

Il contenuto del testo edito sarebbe una stringa di testo.

“What projects men make—what queer turns they take, Since steam has improved our condition; They never are still, but must cure or must kill With steam physic or steam ammunition. But a short time ago, to a quack you would go, To steam a fat man to a thinner; Now changed from all that, if you wish to get fat, Come to Barton's and eat a steam dinner! Oh dear! think of a scheme, odd though it seem— I'm sure 'twill succeed if you make it by steam.”

Il testo, però, deve essere presentato al lettore con le formattazioni grafiche stabilite dall'editore, rispettando gli a capo e le enfasi come segue:

“What projects men make—what queer turns they take,
Since *steam* has improved our condition;
They never are still, but must cure or must kill
With steam physic or steam ammunition.
But a short time ago, to a quack you would go,
To steam a fat man to a thinner;
Now changed from all that, if you wish to get *fat*,
Come to Barton's and eat a *steam dinner!*
Oh dear! think of a scheme, odd though it seem—
I'm sure 'twill succeed if you make it by steam.”

¹⁶ <https://tei-c.org/release/doc/tei-p5-doc/en/html/examples-lg.html#index-egXML-d39e31394>.

Per integrare il testo edito nell'interfaccia grafica, bisogna “tradurre” sia il testo sia la struttura e la semantica espressa dalla marcatura in XML/TEI nel linguaggio utilizzato dallo strumento informatico, per esempio in HTML, ottenendo la seguente porzione di codice HTML:

```
<p>
  What projects men make—what queer turns they take,<br>
  Since <i>steam</i> has improved our condition;<br>
  They never are still, but must cure or must kill<br>
  With steam physic or steam ammunition.<br>
  But a short time ago, to a quack you would go,<br>
  To steam a fat man to a thinner;<br>
  Now changed from all that, if you wish to get <i>fat</i>,<br>
  Come to Barton's and eat a <i>steam dinner!</i><br>
  Oh dear! think of a scheme, odd though it seem—<br>
  I'm sure 'twill succeed if you make it by steam.<br>
</p>
```

Il testo è racchiuso nell'elemento `<p>`, che rappresenta un paragrafo. L'elemento `
` (*break*) fa sì che il testo vada a capo, mentre l'elemento `<i>` (*italic*) rende graficamente in corsivo l'enfasi posta dall'editore su alcune parole del testo.

L'operazione di formattazione del singolo componente sopra descritta rientra in un'operazione più ampia, che in gergo informatico viene chiamata *parsing* e consiste nella traduzione di un documento, per esempio un file XML/TEI, in un *object model*, ovvero un insieme di oggetti e classi.¹⁷ Per ogni componente è necessario specificare come eseguire il *parsing*, implementando un'operazione dedicata, che possiamo denominare “format”, che come parametro prenda il componente recuperato con l'operazione “retrieve” (“retrievedComponent”) e la tipologia del componente da formattare (per esempio una voce d'apparato, una nota di commento, ecc.). Questa operazione restituisce il componente formattato della tipologia specificata.

Il diagramma UML precedente si arricchisce, dunque, come si osserva in figura.

¹⁷ Un esempio di *object model* è il Document Object Model (DOM), un insieme di oggetti che in un *browser* rappresentano una pagina web e gli elementi contenuti al suo interno, utilizzata dai programmi per esaminare e modificare dinamicamente la pagina.

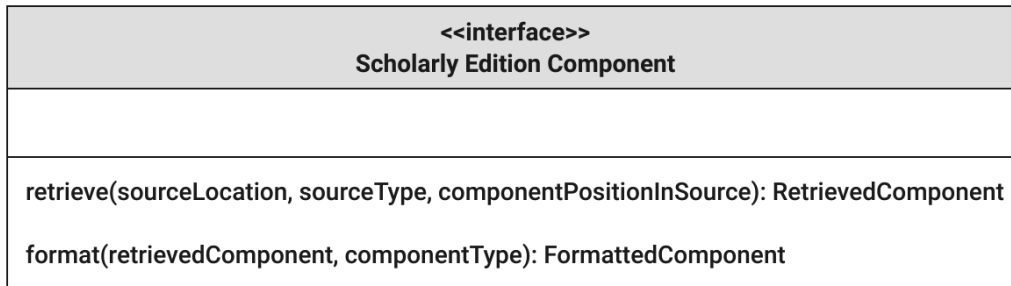


Figura 4.79 Diagramma UML dell'interfaccia *Scholarly Edition Component* con i metodi *retrieve* e *format*.

Dopo essere stato recuperato e formattato, occorre associare al componente delle operazioni e degli attributi per poter gestire l'interazione da parte dell'utente-lettore e rendere il componente definitivamente pronto per essere fruito. Prima di illustrare alcune modalità di base in cui l'utente può interagire con un'edizione scientifica digitale, occorre capire come ottimizzare e facilitare la scrittura del codice, favorendo anche un riuso delle sue diverse parti.

4.4.2 Il paradigma MVC

Per allestire un componente di un'edizione scientifica digitale nell'interfaccia grafica dello strumento informatico, è necessario scrivere diverse porzioni di codice: da un lato abbiamo il componente formattato, dall'altro le operazioni per il recupero e la formattazione del componente e la gestione delle sue interazioni. Implementare le tre diverse attività che lo strumento informatico deve svolgere all'interno di un'unica porzione del codice, rischia di rendere quella porzione molto corposa e difficile da mantenere. L'obiettivo è, invece, quello di strutturare il codice nel modo più flessibile e chiaro possibile, per cui conviene suddividere su più porzioni di codice le tre diverse attività.

Nel campo dell'ingegneria del software, si è affermato a partire dagli anni '80 un paradigma o *pattern* per la realizzazione delle interfacce grafiche chiamato Model View Controller (MVC).¹⁸ Questo paradigma separa l'insieme dei dati alla base dello strumento informatico e la loro struttura ("Model"), dal modo in cui sono presentati all'utente nell'interfaccia ("View") e dalle funzionalità mediante le quali viene gestita l'interazione ("Controller").

"MVC consists of three kinds of objects. The Model is the application object, the View is its screen presentation, and the Controller defines the way the user interface reacts to user input. Before MVC,

¹⁸ Al giorno d'oggi è più utilizzato MVVM ("Model", "View" e "View Model") una revisione di MVC (<https://learn.microsoft.com/en-us/dotnet/architecture/maui/mvvm>).

user interface designs tended to lump these objects together. MVC decouples them to increase flexibility and reuse.” (Gamma et al. 1994, p. 4)

Grazie a questa separazione, il codice risulta più flessibile e le diverse parti possono essere modificate indipendentemente l’una dall’altra. L’utente visualizza i dati sullo schermo (“View”), mentre interagisce con essi grazie al “Controller” che determina le modalità di interazione e come lo strumento informatico debba “rispondere”, modificando il modello. Il modello a sua volta aggiorna il modo in cui i dati si presentano sullo schermo (“View”), in modo da presentare all’utente la risposta alle proprie richieste.

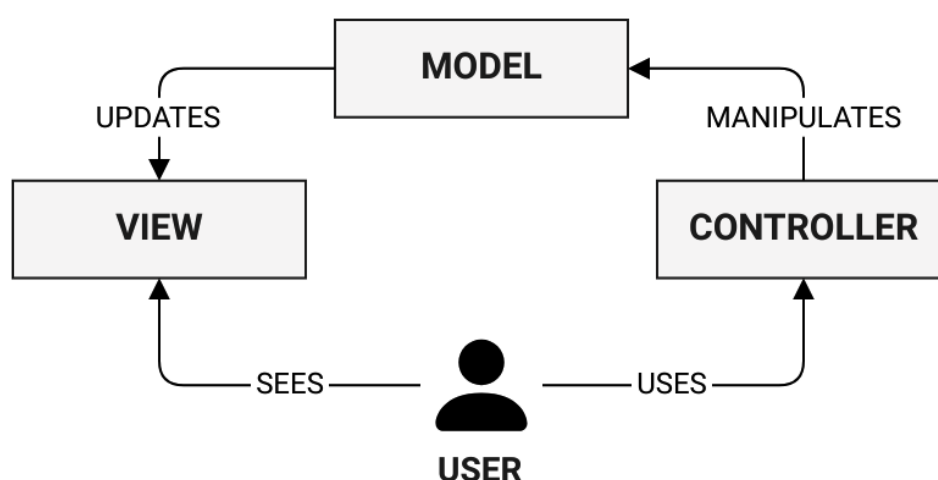


Figura 4.80 Schema del pattern MVC.

Per comprendere la distinzione tra i tre oggetti (Model, View e Controller), si propone un esempio di componente grafico, sviluppato secondo il paradigma MVC.

Nell’interfaccia grafica di un applicativo web pensato per rispondere a dei sondaggi (come per esempio Google Moduli) l’utente ha a disposizione una serie di campi da compilare e un pulsante (in inglese *button*) per salvare e inviare i dati inseriti. Questo pulsante sullo schermo dell’utente può presentarsi in modo diverso, ovvero con diverse modalità di visualizzazione implementate all’interno di un documento HTML: può essere un rettangolo verde contenente l’etichetta “Salva”, oppure di forma circolare e con all’interno l’icona di un floppy disk.

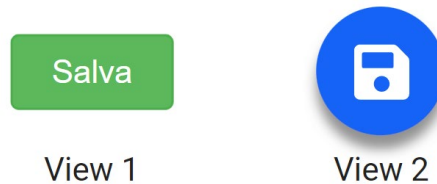


Figura 4.81 Esempio di due “view” differenti.

Il pulsante è associato a una proprietà, contenuta in un'altra porzione di codice dello strumento, denominata “datiSalvati” e di tipo booleano (cioè che può essere vera o falsa), che tiene traccia se i dati inseriti dall'utente sono stati salvati o meno. Se i dati sono ancora da salvare (datiSalvati = falso), l'utente può cliccare sul pulsante, altrimenti il pulsante viene disabilitato. La proprietà “datiSalvati” e il suo valore costituiscono il modello del pulsante.

Infine, in un'altra porzione del codice è definita la funzione associata al pulsante “salvaDati”, che viene eseguita quando l'utente clicca sul pulsante e che costituisce il “Controller” del componente. Questa funzione prende i dati inseriti dall'utente, li salva (per esempio in un database) e aggiorna il valore della proprietà “datiSalvati” (il modello), da falso a vero.

Nel campo della filologia digitale, il paradigma MVC è stato utilizzato per sviluppare la seconda versione del software EVT (vd. 1.3.4):

“EVT 2.0 is already under development using a different approach in order to improve flexibility and modularity, to make it easier to implement new features and to adjust the UI layout for different kinds of editions. This is why the development team decided to refactor the whole code of the viewer and base it on AngularJS (<https://angularjs.org>), a Javascript framework which implements the MVC (Model View Controller)¹⁹ pattern to separate presentation, data and logic components, providing a great modularity of the web application.” (Del Turco e Di Pietro 2017, p. 279)

Il pattern MVC rappresenta una soluzione per rendere il codice più facile da sviluppare e mantenere, separando i dati dalla loro visualizzazione e dalla gestione della loro interazione. Il modello concettuale costituisce, dunque, il nucleo del “Model” dello strumento informatico. Per ottimizzare la scrittura del codice, occorrono ulteriori strategie in modo tale da riusare il più possibile le diverse porzioni di codice.

¹⁹ A titolo informativo si precisa che la seconda versione del framework AngularJS, ovvero Angular, non è basato sul pattern MVC, ma su MVVM (vedi nota 18).

Si chiude questa sezione con un appunto tecnico sul termine *pattern* sopra utilizzato come sinonimo di paradigma e associato a MVC. Nella letteratura scientifica sulla programmazione a oggetti con *pattern* (o *design pattern*) si indica una soluzione generale a un particolare tipo di problema. Nel caso del *pattern* MVC il problema da risolvere è la separazione tra vista, logica e dati. L'idea di fondo è che nello sviluppo di software alcuni problemi vengono riscontrati di frequente a prescindere dal campo applicativo. Problemi che possono essere risolti con le medesime soluzioni. A partire dagli anni '80 in poi la comunità di sviluppatori secondo il paradigma a oggetti ha elaborato e documentato accuratamente problematiche diffuse e le rispettive soluzioni. Lo sviluppatore informatico, invece, di dover trovare una soluzione da sé, può attingere a quelle che si sono consolidate in letteratura, i *design pattern* appunto. Nella letteratura scientifica dedicata a questo argomento uno dei libri più noti e che è diventato un riferimento imprescindibile è *Design Patterns: Elements of Reusable Object-Oriented Software* (Gamma et al, 1994), più volte citato anche nel corso di questo capitolo.

L'utilizzo di *pattern* comporta una serie di vantaggi a livello di sviluppo. Prima di tutto si ha la garanzia di adottare delle soluzioni robuste e flessibili, riconosciute dalla comunità degli sviluppatori. Dopo di che, permette anche a sviluppatori meno esperti di sviluppare codice qualitativamente migliore e più velocemente.

4.4.3 La realizzazione di implementazioni diverse

Come anticipato, le tre attività alla base di uno strumento di visualizzazione, cioè il recupero, la formattazione e la gestione dell'interazione, servono per ogni componente di un'edizione scientifica digitale. Parte del codice che viene scritto per recuperare un componente, per formattarlo o per gestirlo, può essere riutilizzata per un altro tipo di componente.

La principale strategia per poter riutilizzare il codice tra componenti diversi, come anticipato, consiste nel servirsi del meccanismo dell'ereditarietà. Le sottoclassi ereditano le operazioni definite nella superclasse, con la possibilità di espanderle e sovrascriverle, per adattarle alle proprie peculiarità.

Un'altra soluzione già anticipata, consiste nella definizione di interfacce. Le interfacce forniscono un insieme coerente di funzioni (come le funzionalità "retrieve" e "format" definite per l'interfaccia `Scholarly Edition Component`). Una classe poi può "realizzare" una o più

interfacce, implementando concretamente le operazioni definite al loro interno. Il rapporto che lega una classe del modello concettuale alle sue sottoclassi, a livello implementativo può essere tradotto non solo come una “generalizzazione” (vd. definizione alla nota 14), ma anche come una “realizzazione”:

“Realization is a specialized Abstraction relationship between two sets of model Elements, one representing a specification (the supplier) and the other represents an implementation of the latter (the client). Realization can be used to model stepwise refinement, optimizations, transformations, templates, model synthesis, framework composition, etc” (UML v. 2.5.1, p. 96)

Il meccanismo dell’ereditarietà e la distinzione tra l’interfaccia e l’implementazione concreta consentono una maggiore flessibilità nello sviluppo e nella manutenzione dello strumento informatico. Ci sono diversi casi in cui disporre di questa flessibilità è molto utile, per esempio, quando lo strumento informatico deve essere aggiornato, perché le tecnologie su cui era originariamente basato sono diventate obsolete. Un altro caso è quando lo sviluppatore vuole predisporre diverse visualizzazioni dello stesso componente dell’edizione scientifica digitale, per esempio, per mostrare una voce d’apparato all’interno del testo edito come pannello a scomparsa oppure all’interno dell’apparato critico. Un ulteriore caso è quando lo sviluppatore desidera riutilizzare il codice di un altro strumento informatico come base per realizzare il proprio applicativo.

La definizione di una classe astratta o di un’interfaccia, dalla quale derivare poi, con il meccanismo dell’ereditarietà, molteplici classi per realizzare delle implementazioni diverse di tale classe, è una prassi comune. Tuttavia, l’organizzazione gerarchica delle classi può portare a delle rigidità all’interno dello strumento informatico:

“When an abstraction can have one of several possible implementations, the usual way to accommodate them is to use inheritance. But this approach isn’t always flexible enough. Inheritance binds an implementation to the abstraction permanently, which makes it difficult to modify, extend, and re-use abstractions and implementations independently” (Gamma et al. 1994, p. 151).

Il problema sopra descritto, nel contesto della presente ricerca, che cosa comporta? Comporta un legame a doppio filo tra il modello concettuale e il modello implementativo, per cui gli sviluppatori informatici sono limitati nella possibilità di stabilire l’implementazione e, al contempo, se è necessario modificare il modello concettuale (per adattarlo a delle nuove esigenze scientifiche), lo strumento informatico non è più compatibile con la nuova versione del modello concettuale.

Una soluzione a questo problema consiste nell'applicare il *design pattern* chiamato “Bridge” (Gamma et al. 1994, p. 151), il cui scopo è proprio quello di separare l'astrazione dalla sua implementazione, così che le due possano variare in modo indipendente. Questo *design pattern* prevede la definizione nel codice di quattro diversi tipi elementi (cfr. Gamma et al. 1994, p. 154):

- “Abstraction”, che definisce l'interfaccia dell'astrazione e mantiene il riferimento a un oggetto di tipo “Implementor” (vd. sotto). Nel presente caso Abstraction è la classe Scholarly Edition Component;
- “RefinedAbstraction”, che estende l'interfaccia definita da “Abstraction”, dunque tutte le sottoclassi di Scholarly Edition Component (Edited Text, Accompanying Material, Introduction, ecc.);
- “Implementor”, che definisce l'interfaccia per l'implementazione delle classi. Questa interfaccia non deve corrispondere esattamente all'interfaccia di “Abstraction”, ma le due interfacce possono essere anche molto differenti. Solitamente l'interfaccia di Implementor fornisce solo delle operazioni di base, mentre “Abstraction” definisce delle operazioni di livello più alto, basate però su quelle basilari di “Implementor”;
- “ConcreteImplementor”, che implementa l'interfaccia dell’“Implementor” e definisce le sue implementazioni concrete.

I diversi elementi descritti e i rapporti reciproci in notazione UML possono essere rappresentati come in figura.

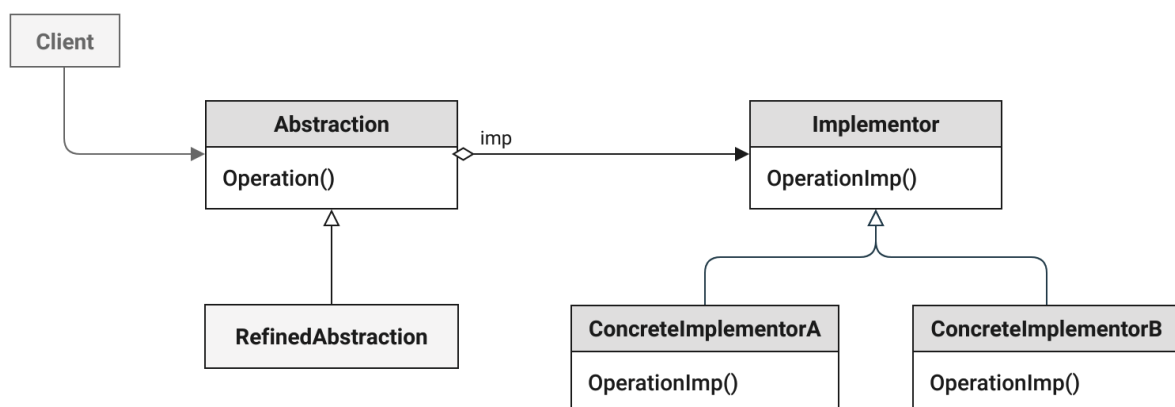


Figura 4.82 Schema che presenta gli elementi alla base del design pattern “Bridge” (Gamma et al. 1994).

La classe “Client” nel diagramma rappresenta la parte visibile o accessibile dello strumento informatico ed è collegata solo alla classe “Abstraction”, in altre parole il client vede solo l’astrazione, mentre le diverse implementazioni rimangono nascoste. A livello concreto, questo *design pattern* permette di sviluppare strumenti informatici per la visualizzazione che siano coerenti con la struttura astratta di un’edizione scientifica e al contempo compatibili tra di loro in termini di caratteristiche astratte dei componenti implementati. Inoltre, anche gli Implementor possono essere riutilizzati e condivisi tra oggetti diversi.

Per creare e configurare un “Bridge” si utilizza spesso un altro *design pattern* denominato “Abstract Factory” (letteralmente ‘fabbrica astratta’), il cui scopo è quello di fornire un’interfaccia per la creazione di famiglie di oggetti correlati o dipendenti tra di loro senza specificare le loro classi concrete (cfr. Gamma et al. 1994, p. 87 - 95).

4.4.4 La gestione dell’interazione con i componenti grafici

In questa parte del capitolo si offrono delle indicazioni sulle funzionalità da implementare nell’interfaccia grafica, affinché i componenti di un’edizione critica possano essere fruiti agevolmente e in modo coerente rispetto a quanto espresso nel modello concettuale.

Le classi di esempio qui riportate in notazione UML sono derivate dai *class diagram* presentati alla sezione 4.3 e costituiscono delle basi per dei potenziali “controller” (vd. 4.4.2), ovvero le porzioni di codice che si occupano di gestire l’interazione dell’utente-lettore con l’interfaccia grafica.

4.4.4.1 Una visualizzazione flessibile

Un primo aspetto da considerare è che l’ambiente digitale, rispetto alle edizioni a stampa, offre una notevole flessibilità nella disposizione e nell’organizzazione dei componenti alla base di un’edizione scientifica digitale. Flessibilità a vantaggio non solo dell’utente-editore, che può predisporre i materiali nel modo che ritiene più consono rispetto alle loro caratteristiche, ma anche (e soprattutto) a

vantaggio dell'utente-lettore, che può servirsi dell'interfaccia grafica di un'edizione scientifica digitale come di uno spazio di lavoro, "a digital workspace" (Del Turco 2019, p. 103).²⁰

Questa metafora dello "spazio di lavoro" è stata adottata in svariate edizioni scientifiche digitali e da diversi strumenti di visualizzazione e misti, tra cui EVT, eLaborate e Versioning Machine. Nell'interfaccia grafica di quest'ultimo, per esempio, l'utente può selezionare da un menù a tendina (1) quali versioni del testo edito nascondere o mostrare. Una volta aggiunto all'interfaccia, il testo appare in un riquadro (2) che può essere posizionato liberamente. Il riquadro può anche essere ridimensionato, utilizzando l'apposito pulsante (3), ma per evitare che il testo al suo interno diventi illeggibile, per il riquadro sono state impostate delle dimensioni minime di larghezza e altezza.

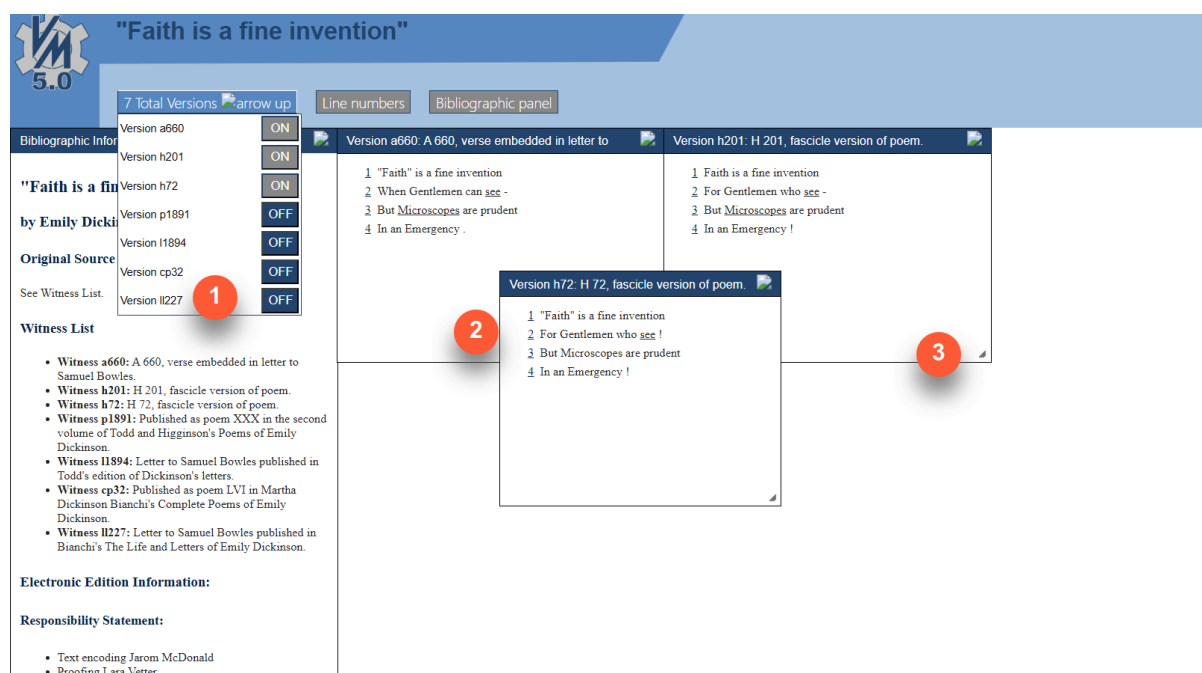


Figura 4.83 Schermata dell'interfaccia grafica dello strumento di visualizzazione Versioning Machine.

A livello implementativo, occorre prima di tutto domandarsi quali componenti rendere flessibili e in che modo. Nel caso di un'edizione composta da più testi allestiti dall'editore, come per esempio il testo critico e le trascrizioni dei testimoni, si può adottare una soluzione analoga a quella di Versio-

²⁰ Un'interfaccia flessibile non facilita semplicemente la consultazione dell'edizione scientifica digitale ma getta anche le fondamenta per favorire il riuso dei dati e per realizzare edizioni collaborative, in cui i materiali allestiti dall'editore possono essere rivisti, corretti, arricchiti e commentati dai lettori.

ning Machine. Nel caso di un'edizione composta da un solo testo edito, accompagnato da molti materiali di corredo, si può, invece, rendere fisso e sempre visibile il testo edito, mentre i materiali di corredo possono essere presentati su richiesta dell'utente e posizionati a piacere.

A prescindere dal modo in cui l'interfaccia viene resa flessibile, si può sfruttare il meccanismo dell'ereditarietà (descritto alla sezione 4.3.1), per stabilire nel *controller* associato a *Scholarly Edition Component* dei comportamenti comuni a tutti i componenti, specificando poi a livello di controller della singola sottoclasse se mantenere il comportamento ereditato o se modificarlo.

Nel *controller* di *Scholarly Edition Component* può convenire implementare alcune delle operazioni più comuni, tra cui quella di mostrare (“show”) o nascondere (“hide”) il componente dall'interfaccia, quella per posizionare il componente (“move”) e per ridimensionarlo (“resize”). Per tenere traccia se il componente è mostrato o meno nell'interfaccia, ovvero per conoscere lo “stato” di quell'oggetto, può essere utile la proprietà booleana “isVisible”. Per gestire il posizionamento del componente dell'interfaccia e il suo possibile ridimensionamento, conviene stabilire delle proprietà quali il suo posizionamento di default, le dimensioni minime da garantire e le dimensioni di default.

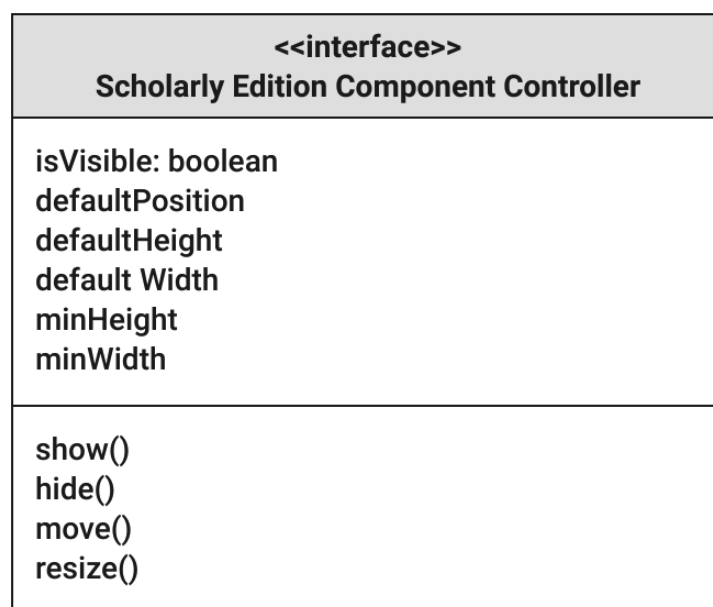


Figura 4.84 Class diagram di un controller per la classe *Scholarly Edition Component*.

4.4.4.2 La consultazione del testo edito e degli apparati

Un aspetto che emerge chiaramente dal modello concettuale è che le voci d'apparato sono collegate direttamente a luoghi del testo edito e che gli apparati sono consultati dal lettore nel mentre che legge il testo edito. A livello grafico il collegamento tra il testo edito e gli apparati può essere sviluppato in modi diversi, a seconda di come i diversi componenti sono posizionati e formattati nell'interfaccia grafica. A livello funzionale, è possibile individuare delle operazioni di base che devono essere implementate a prescindere dall'aspetto grafico dei componenti coinvolti.

Per comprendere concretamente quali siano i componenti coinvolti e come l'utente possa interagire con loro si propone un esempio di testo critico accompagnato da un apparato critico, tratto dall'edizione digitale del Faust di Goethe (a cura di Bohnenkamp, Henke e Jannidis, 2019),²¹ che a mio avviso è molto ben fatta in termini di interfaccia grafica. Il testo critico è presentato in versi e disposto al centro della pagina. Alla destra del testo critico figurano in alto un menù di navigazione, che rimane visibile anche quando l'utente fa scorrere la pagina verso il basso, e l'apparato critico. Le singole voci dell'apparato critico sono posizionate all'altezza del verso in cui è contenuta la porzione di testo interessata. Alcuni versi presentano uno sfondo grigio scuro, mentre altri di un grigio più chiaro, in questo modo l'utente riesce a individuare a colpo d'occhio i punti del testo soggetti a maggiore variabilità nella tradizione testuale.

²¹ Johann Wolfgang Goethe: Faust. Historisch-kritische Edition. Herausgegeben von Anne Bohnenkamp, Silke Henke und Fotis Jannidis unter Mitarbeit von Gerrit Brüning, Katrin Henzel, Christoph Leijser, Gregor Middell, Dietmar Pravida, Thorsten Vitt und Moritz Wissenbach. Version 1.2RC. Frankfurt am Main / Weimar / Würzburg 2019, Startseite, URL: <http://v1-2.faustedition.net/>, abgerufen am 1.6.2023.

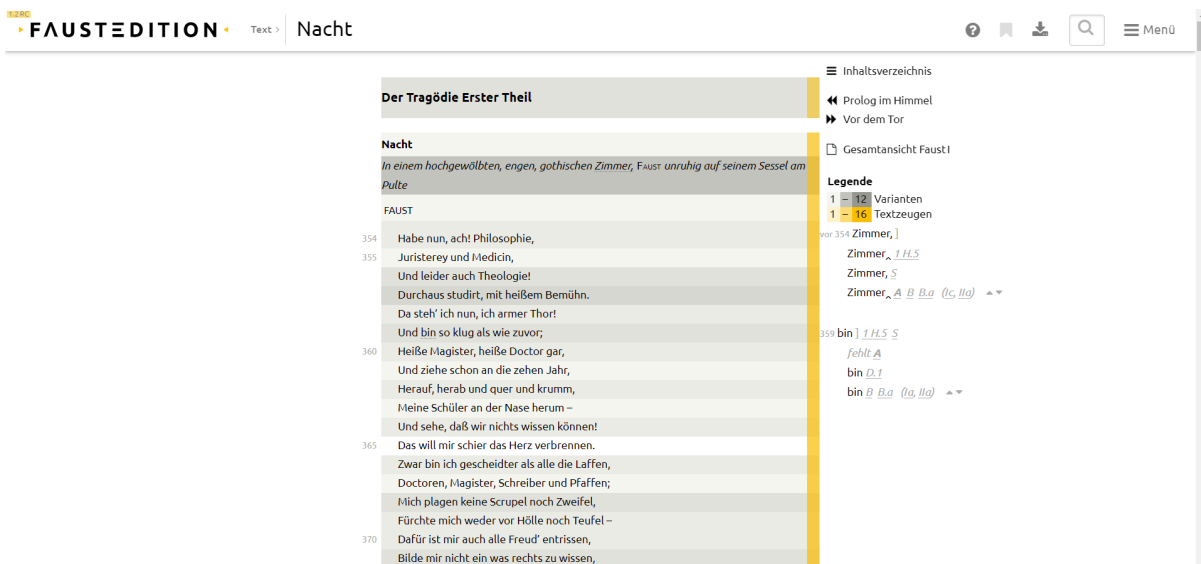


Figura 4.85 Schermata dell'interfaccia grafica della Faust Edition.

All'interno del verso, il passo collegato a una voce dell'apparato critico è evidenziata con una lieve sottolineatura tratteggiata (nella figura sotto si tratta di "bin"). Nella voce d'apparato, il passo, che coincide con la lezione messa a testo dall'editore, viene presentato con la formattazione tipica delle edizioni a stampa, ovvero seguita dal segno "]", preceduto dal numero del verso ("359") e seguito dalle sigle dei testimoni che concordano ("1 H.5 S") e dall'elenco delle lezioni rifiutate sotto con le rispettive sigle. Se l'utente passa il cursore sopra al passo nel testo critico ("bin"), compare un *tooltip* (una nuvoletta) che riassume il numero di varianti presenti nella tradizione e di testimoni coinvolti e, al contempo, vengono evidenziate con uno sfondo giallo sia il passo del testo che la voce dell'apparato critico corrispondente.

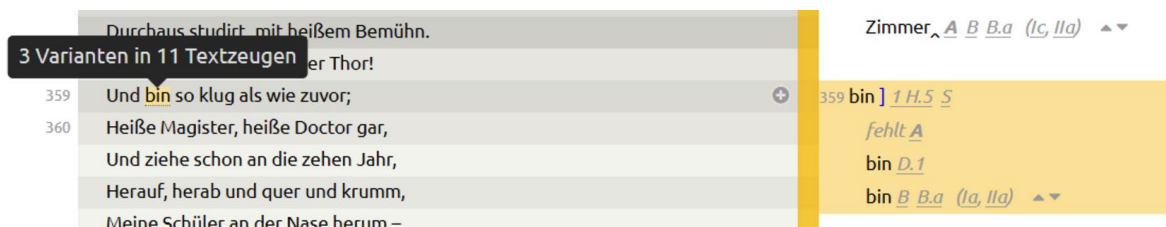


Figura 4.86 Schermata dell'apparato critico e del testo critico all'interno della Faust Edition.

Al click sul verso o sulla voce dell'apparato critico collegata, sotto al verso si apre un pannello a scomparsa all'interno del quale viene presentata per ogni testimone o gruppo di testimoni concordi la trascrizione completa di quel verso.

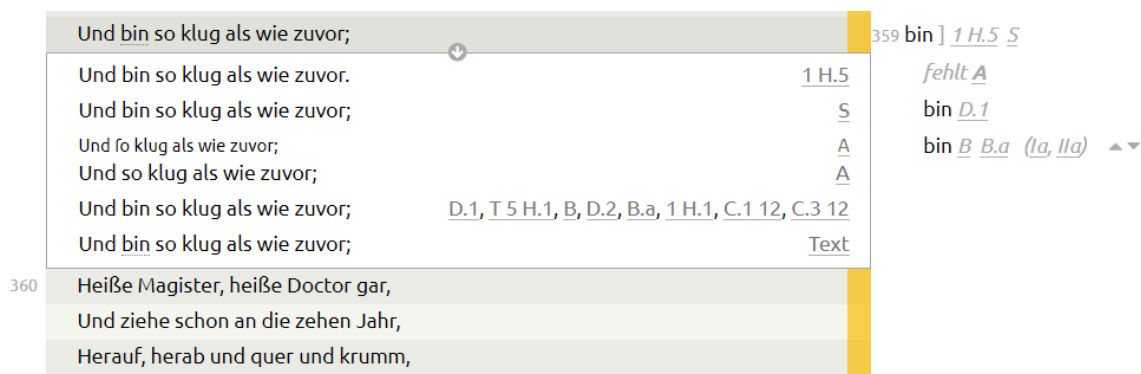


Figura 4.87 Schermata di un verso del testo critico e delle rispettive trascrizioni dei testimoni all'interno della *Faust Edition*.

Sulla base di questo esempio è possibile dedurre quali siano le funzionalità di base per una corretta fruizione del testo edito insieme ai suoi apparati. Prima di tutto, i passi del testo edito soggetti a variazione nella tradizione testuale devono essere riconoscibili e facilmente individuabili, attraverso una caratterizzazione grafica come la sottolineatura o soluzioni più articolate come la mappa di calore (grigio più scuro o più chiaro) dell'esempio.

I passi collegati a una o più voci d'apparato devono essere associati a un selettore, ovvero un elemento interattivo che rimandi il lettore alla voce d'apparato. Nell'edizione di esempio il selettore coincide con il passo stesso ("bin"), ma a livello concettuale sono due oggetti distinti (*Critical Text Passage* e *PassageSelector*). Anche in termini di interfaccia grafica possono essere separati, per esempio aggiungendo un elemento interattivo che costituisca il selettore, vicino al passo del testo critico, come si vede nell'immagine di esempio che segue.

Und bin  so klug als wie zuvor;

Figura 4.88 Esempio di selettore separato dal passo del testo critico.

Il modo in cui il selettore "rimanda" alla voce d'apparato può variare, ma l'importante è che l'utente-lettore venga guidato dal testo verso la voce d'apparato. Se l'apparato non è sempre visibile, per esempio, una volta selezionato il selettore, l'apparato deve essere mostrato. Oppure se l'apparato è visibile nell'interfaccia ma è distante dal testo edito, la pagina deve essere allineata al punto in cui si trova la voce d'apparato mediante una funzionalità di scorrimento automatico.

Il terzo aspetto che si può derivare dal modello è che il collegamento tra il passo del testo edito e la voce d'apparato non è a senso unico ma bidirezionale, per cui anche dalla voce d'apparato deve

essere possibile poter risalire al punto corrispondente del testo edito, con gli stessi accorgimenti presentati nel precedente paragrafo.

Infine, l'ultimo e quarto aspetto che emerge dal modello concettuale è che il selettore utilizzato nel testo edito viene (quasi sempre) mostrato anche all'interno della voce d'apparato (come indicato dalla proprietà `selector is shown in`), per esplicitare il collegamento tra i due e rendere la voce d'apparato più facile da individuare.

I requisiti funzionali sopra descritti potrebbero essere implementati a livello di codice, realizzando tre classi "controller": una per il selettore, una per il passo del testo edito e una per la voce d'apparato. Il controller per la voce d'apparato contiene in una proprietà (denominata nel modello concettuale `apparatus entry refers to`) il riferimento al passo del testo edito collegato. Analogamente il controller per il passo del testo edito contiene i riferimenti alle voci d'apparato collegate nella proprietà `is referred to by` (che nel modello concettuale è l'inverso di `apparatus entry refers to`). Mentre una voce d'apparato è legata a un solo passo del testo edito, un passo del testo edito può essere collegato a più voci di apparati di diverso tipo. Il collegamento dal passo del testo edito alla voce d'apparato e viceversa viene gestito con le operazioni "go to apparatus entry" del controller del passo del testo edito e "go to edited text passage" del controller della voce d'apparato. L'evidenziazione del passo edito e della voce d'apparato è gestita con le rispettive operazioni "highlight" e "remove highlight" e il supporto della proprietà "is highlighted". Il passo del testo edito è collegato a un selettore. Quest'ultimo per gestire l'interazione dell'utente presenta le operazioni complementari "select" e "deselect" e l'attributo "isSelected". Nel caso in cui non solo la singola voce ma l'intero apparato debba essere mostrato o nascosto, si possono utilizzare le funzionalità "show" e "hide" del controller dell'apparato. Infine, nel caso in cui l'editore volesse offrire il testo edito in una modalità semplificata, senza elementi interattivi o evidenziazioni a distrarre dalla lettura, i selettori potrebbero essere nascosti (con la funzionalità "hide") così come le evidenziazioni dei passi del testo edito ("remove highlight").

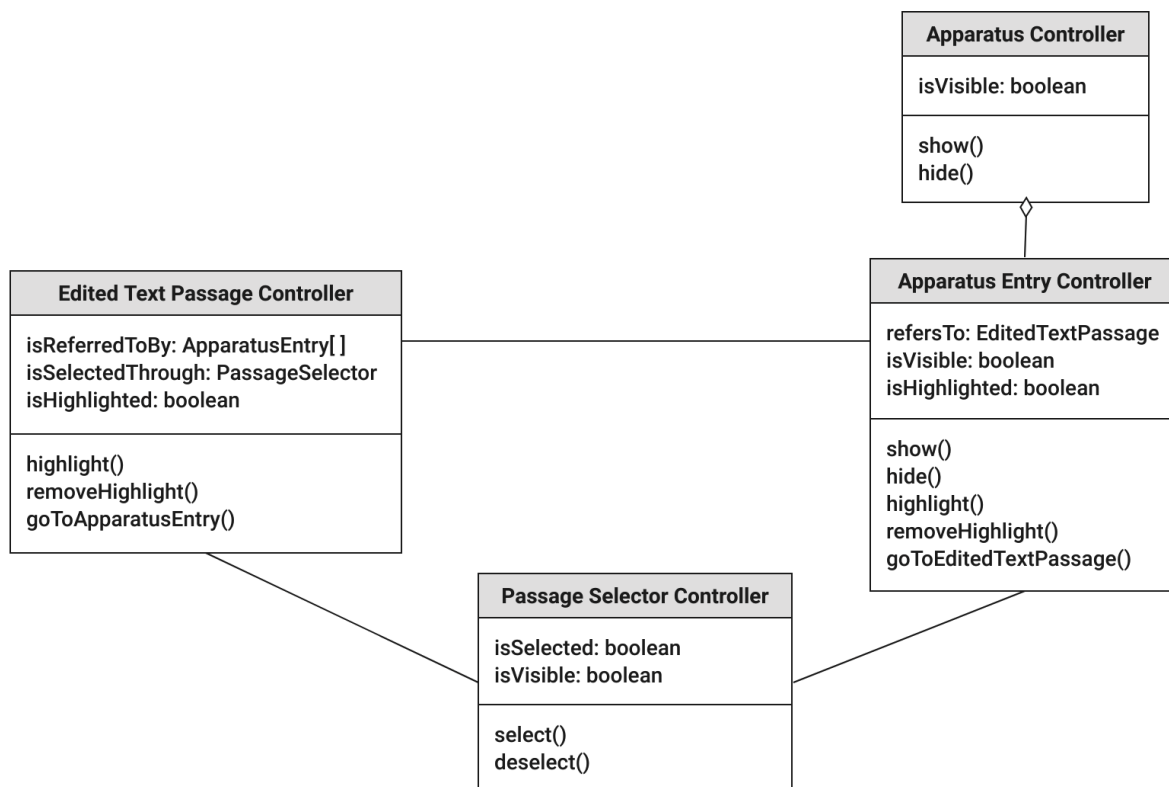


Figura 4.89 Class diagram dei controller per l'apparato, la voce d'apparato, il passo del testo edito e il selettore del passo.

Le proprietà e le operazioni presenti nel precedente *class diagram* non sono obbligatorie ma sono state inserite come esempio di possibile implementazione delle funzionalità coerente con quanto espresso dal modello concettuale.

4.4.4.3 Le informazioni sulla tradizione testuale

Nella voce dell'apparato critico le lezioni sono collegate ai rispettivi testimoni attraverso le sigle. In ambiente digitale, il rimando al testimone (così come tutti i rimandi) diventa prima di tutto un collegamento ipertestuale, per esempio all'elenco dei testimoni o alla trascrizione del testimone. Nel caso in cui non si voglia offrire un semplice collegamento, ma una funzionalità più complessa, occorre specificare nel controller della classe `Siglum Reference` delle operazioni dedicate, come si vede nel *class diagram* di esempio sotto.

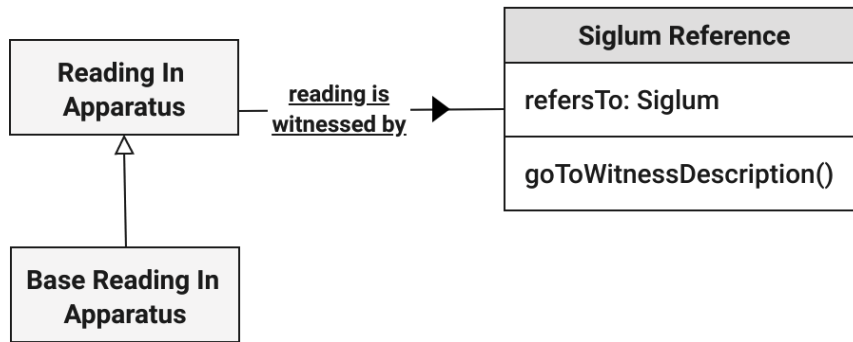


Figura 4.90 Class diagram delle classi *Reading In Apparatus* e *Siglum Reference*.

Nel modello concettuale l'unico componente, oltre a *Reading In Apparatus*, collegato alla classe *Siglum Reference* (anche se attraverso parecchi passaggi intermedi) è *Witness Description*, ovvero la descrizione del testimone all'interno dell'elenco dei testimoni. Tuttavia, il modello concettuale può essere ampliato in futuro, in modo tale da includere nell'edizione anche le trascrizioni dei testimoni e prevedere che dal riferimento di sigla in apparato sia possibile passare direttamente al punto corrispondente nella trascrizione. Un altro collegamento interessante potrebbe essere dalla sigla del testimone in apparato allo stemma codicum, specie nei casi in cui un intervento critico è stato effettuato in base a dei criteri stemmatici.

4.4.4.4 Il filtro delle varianti e delle diverse tipologie di intervento critico

Nel modello concettuale la parte dedicata alla descrizione del rapporto tra il testo critico e i testimoni offre una serie di classi per descrivere le lezioni offerte in apparato sotto diversi punti di vista, tra cui la loro tipologia (*Reading In Apparatus Type*), la loro causa (*Reading In Apparatus Cause*), la mano scribale e l'eventuale editore esterno che ha proposto la lezione. In modo analogo, il modello concettuale contempla delle classi per distinguere diverse tipologie di intervento critico (per esempio congettura, *emendatio ex fontibus*, ecc.).

In fase di implementazione, le caratteristiche delle lezioni e degli interventi critici possono essere utilizzate per costruire delle funzionalità di ricerca o di filtro, che potrebbero essere molto utili nel caso in cui l'opera edita sia caratterizzata da una vasta tradizione. In un apparato critico molto ricco e articolato, per esempio, l'utente potrebbe voler visualizzare solo le lezioni considerate frutto di un'omissione volontaria oppure effettuare una ricerca per individuare degli errori congiuntivi.

Il software EVT 2, per esempio, offre la possibilità di evidenziare con colori diversi le lezioni (stabilite nel testo critico o contenute nelle trascrizioni dei testimoni) filtrandole in base alla loro causa, alla tipologia, alla mano scribale e al responsabile scientifico.

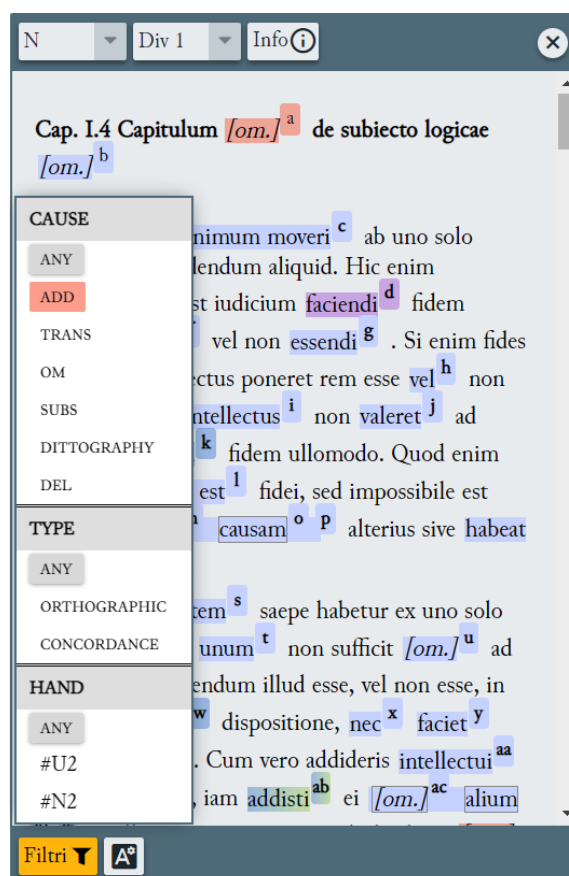


Figura 4.91 Funzionalità di filtro delle varianti offerta nel software EVT 2.

Da un punto di vista implementativo, per sviluppare queste funzionalità occorre determinare su quali classi del modello concettuale intervengono, per esempio la classe *Reading In Apparatus* o la classe *Critical Text Passage*, e aggiungere a tali classi degli attributi che rimandino alle classi del modello concettuale utilizzate come parametro per la funzionalità. Si consideri l'esempio sopra citato della funzionalità di filtro all'interno dell'apparato critico per mostrare solo le lezioni frutto di un'omissione volontaria. La classe *Reading In Apparatus* deve avere un attributo "type". La funzionalità "filtro" (rappresentata in figura dalla classe "Critical Apparatus Filter") controlla tutti gli oggetti di *Reading In Apparatus* e il valore del rispettivo attributo type (per esempio attraverso l'operazione "find matching readings"), nascondendo quelli in cui l'attributo type è diverso da "deliberate omission" con un'operazione denominata "hide readings".

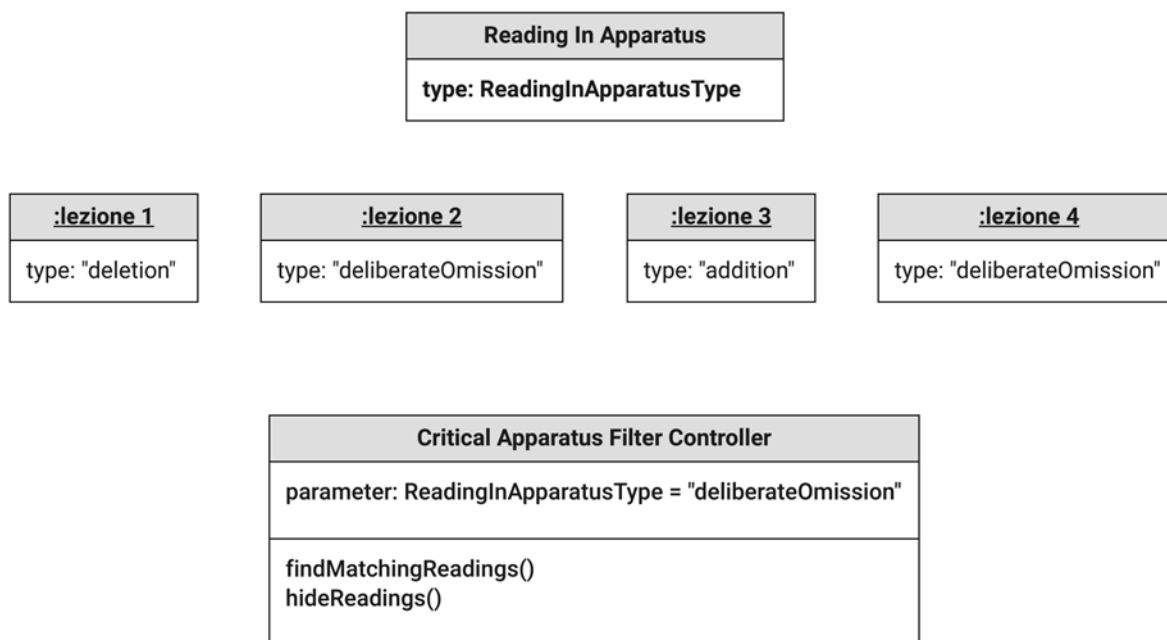


Figura 4.92 Class diagram della funzionalità di filtro delle varianti.

4.4.5 La configurazione dell'interfaccia grafica

Il modello concettuale si presta come riferimento per determinare un altro aspetto fondamentale di uno strumento per la visualizzazione di edizioni scientifiche digitali, ovvero le opzioni di configurazione. Come illustrato nel precedente capitolo (vd. 3.5.4 e 3.5.5), il dominio di uno strumento di visualizzazione per le edizioni scientifiche digitali è dominato da due attori: l'editore e il lettore. Per questo motivo, uno strumento di visualizzazione generico si rivolge a due utenti diversi: l'utente-lettore e l'utente-editore. Uno dei requisiti fondamentali per rendere uno strumento di visualizzazione adattabile a edizioni diverse e soprattutto appetibile per gli utenti-editori è che consenta un certo grado di personalizzazione. Più è ampio il margine per personalizzare la propria edizione che uno strumento informatico offre, maggiori sono le probabilità che lo strumento venga adottato e che soddisfi le esigenze scientifiche degli utenti-editori. A maggior ragione se le modalità di personalizzazione sono fondate su un'analisi del dominio applicativo e sull'individuazione di esigenze comuni a edizioni scientifiche digitali diverse.

La configurazione può riguardare diversi aspetti di uno strumento informatico, di cui alcuni più superficiali, come per esempio la scelta dei *font* e della tavolozza di colori, e altri, invece, che influ-

scono a un livello più profondo sul funzionamento e l'organizzazione dei componenti dell'interfaccia. In questo contesto si presentano alcuni esempi di parametri o opzioni di configurazione che si possono derivare dal modello concettuale di esempio.

4.4.5.1 La disposizione dei materiali di corredo

In precedenza abbiamo riportato come esempio di due visualizzazioni (“View”) diverse del medesimo componente la voce d'apparato presentata all'interno del testo e la voce d'apparato presentata nell'apparato critico. La scelta tra queste due viste dovrebbe essere configurabile da parte dell'utente-editore. Più in generale, si presta a essere configurata dall'editore la disposizione dei materiali di corredo rispetto al testo edito. Le diverse opzioni di configurazione dipendono dal tipo di componente e dal modo in cui l'interfaccia grafica è stata progettata. Ad ogni modo, per gli apparati possiamo immaginare come potenziali opzioni, che possano essere posti sotto al testo edito, come nella maggior parte delle edizioni a stampa, all'interno del testo edito oppure accanto al testo edito. L'introduzione non deve essere necessariamente presentata insieme al testo edito, anzi molto spesso viene presentata a parte, addirittura in un'altra pagina web. Tuttavia, se dovesse essere molto breve l'editore potrebbe voler specificare che l'edizione deve essere presentata insieme al testo, precedendolo.

4.4.5.2 La configurazione dell'apparato critico

Molto spesso l'editore, nel mentre che lavora a un'edizione critica digitale, inserisce nell'apparato critico tutte le tipologie di varianti individuate nella tradizione testuale. Tuttavia, in fase di pubblicazione, l'editore potrebbe decidere di non presentare nell'apparato critico alcune tipologie di lezioni ritenute superflue o di scarso interesse filologico. Invece di rimuovere le lezioni dall'edizione, l'editore potrebbe configurare lo strumento informatico, affinché lo strumento generico filtri le lezioni e nasconda quelle indicate dall'editore come da rimuovere.

In modo analogo, in fase di configurazione si potrebbe indicare come l'apparato critico debba essere formattato e strutturato, specificando:

- se l'apparato critico è negativo o positivo;

- le fasce d'apparato in cui suddividere l'apparato, il loro ordinamento e il criterio da utilizzare, per esempio la tipologia delle lezioni, rappresentata dalla classe `Reading In Apparatus Type`, o la famiglia stemmatica di appartenenza del testimone corrispondente;
- all'interno di una voce dell'apparato critico se le lezioni devono essere raggruppate e in base a quale criterio, per esempio mettendo vicine lezioni derivate l'una dall'altra, sfruttando la proprietà `was derived from` della classe `Reading In Apparatus`.

4.4.5.3 Il collegamento dal testo edito all'apparato

I luoghi del testo edito che sono associati a una voce d'apparato sono individuabili grazie a un selettore. Il modello concettuale prevede due diversi tipi di selettore: uno che ripete la porzione di testo interessata (`Passage Quote Selector`) e un altro che indica il punto nel testo edito in cui si trova il luogo interessato (`Passage Position Selector`). In fase di configurazione, si potrebbe consentire all'editore di scegliere la tipologia di selettore (o la combinazione di selettori) che preferisce.

4.4.6 La suddivisione in moduli

Un altro accorgimento che è consigliabile adottare quando si sviluppa uno strumento informatico è l'organizzazione del codice in moduli o componenti, come raccomandato anche dal progetto HRIT (vd. 1.5):

“Modular (component) design and structure are best for digital tools, tasks, and content types.” (Shillingsburg 2013, p. 3)

Nell'ontologia con “componente” si intende genericamente un contenuto, una parte di un'edizione scientifica digitale. In linguaggio informatico con “componente” si intende, invece, “a modular part of a system that encapsulates its contents and whose manifestation is replaceable within its environment” (UML v. 2.5.1, p. 251). In altre parole, un componente racchiude più pezzi di codice, che costituiscono il comportamento e lo stato di un certo numero di classi, specificando il modo in cui può essere chiamato e utilizzato dall'esterno da altri componenti, ovvero i “servizi che fornisce”, e al contempo i servizi che richiede agli altri componenti. Trattandosi di un'unità isolata e ben definita, un componente può essere sostituito da uno equivalente, cioè che offra e che richieda gli stessi

servizie. Per cui i “meccanismi” interni al componente possono essere modificati liberamente, a patto che non venga modificato il modo in cui il componente si interfaccia con gli altri componenti.

La suddivisione in componenti, facilita la manutenzione del software. Se un componente dipende da una libreria esterna o da tecnologie diventate obsolete, è possibile modificare il solo componente, senza dover intervenire sul resto del codice. Inoltre, un software modulare può essere espanso e arricchito più facilmente. Questa funzionalità è particolarmente utile per sviluppare strumenti di visualizzazione per le edizioni scientifiche digitali. Il modello concettuale di esempio descrive pochi elementi di base, ma se l’edizione da visualizzare presenta altri componenti, come per esempio l’apparato delle fonti, la trascrizione dei testimoni, ecc., o necessita di altre funzionalità *ad hoc* è possibile aggiungere nuovi moduli dedicati all’interno del software, senza intervenire sui moduli già realizzati. La suddivisione in componenti permette di comporre software generici come se fossero dei puzzle: a seconda della tipologia dell’edizione scientifica digitale (e del metodo filologico con cui è stata realizzata) un pezzo del puzzle può essere aggiunto o rimosso. E se il pezzo del puzzle deve essere sostituito, è sufficiente crearne uno nuovo che si incastri con gli altri pezzi.

4.4.7 Qualche spunto per il *layout*

Il modello concettuale offre delle indicazioni utili per progettare l’interfaccia di uno strumento di visualizzazione non solo in termini funzionali e di organizzazione del codice, ma anche per determinarne l’aspetto grafico.

Prima di tutto le definizioni stabilite nel modello concettuale possono essere utilizzate anche nell’interfaccia per presentare i contenuti all’utente-lettore. Per applicare pienamente i principi del *domain-driven design*, è fondamentale che l’*ubiquitous language* (vd. 2.4), costituito dal modello concettuale, sia utilizzato per produrre uno strumento informatico dall’inizio alla fine, cioè dalla fase di progettazione a quando l’utente-lettore fruisce l’edizione digitale. Sullo schermo, per esempio, l’utente deve leggere “apparato critico” e non “note critiche” o “annotazioni editoriali”, in questo modo può essere sicuro che si tratti dell’apparato critico.

Un altro aspetto del modello da sfruttare in fase di progettazione dello strumento informatico è la sua organizzazione gerarchica. Rendendo evidenti ed espliciti i rapporti gerarchici tra i componenti, l'utente viene facilitato nella navigazione all'interno dello strumento e nel riconoscimento dei suoi contenuti.

“Increasing the visibility of the hierarchical relationships within a system is one of the most effective ways to increase knowledge about the system” (Lidwell et al. 2010, p. 122).

In termini concreti come si può applicare il principio di *design* della gerarchia nel nostro contesto?

Per esempio, presentando i materiali di corredo con un corpo di carattere inferiore rispetto a quello del testo edito, così da rendere visivamente evidente che il testo è il contenuto più importante, mentre i materiali di corredo ricoprono una funzione ancillare. Un altro esempio riguarda la suddivisione in fasce dell'apparato critico: le fasce devono essere separate, ma devono far parte di un unico elemento grafico, per esempio un riquadro posto in fondo al testo critico. Molti di questi accorgimenti grafici possono essere mutuati dalle convenzioni tipografiche delle edizioni a stampa, come per esempio la separazione tra la lezione stabilita nel testo (*Base Reading In Apparatus*) dalle lezioni rifiutate (*Reading In Apparatus*) mediante la parentesi quadra (“”).

Il modello concettuale indica anche quali sono gli elementi da strutturare e presentare in modo omogeneo, perché a livello concettuale sono analoghi o sono sottoclassi della medesima classe, come per esempio le diverse tipologie di apparato. Questo principio del *design* è definito “consistency”, traducibile con coerenza, e in un sistema può essere implementato a livello estetico, funzionale, interno ed esterno:

“According to the principle of consistency, systems are more usable and learnable when similar parts are expressed in similar ways. Consistency enables people to efficiently transfer knowledge to new contexts, learn new things quickly, and focus attention on the relevant aspects of a task. There are four kinds of consistency: aesthetic, functional, internal, and external.

Aesthetic consistency refers to consistency of style and appearance. Aesthetic consistency enhances recognition, communicates membership, and sets emotional expectations. [...]

Functional consistency refers to consistency of meaning and action. Functional consistency improves usability and learnability by enabling people to leverage existing knowledge about how the design functions. [...]

Internal consistency refers to consistency with other elements in the system. [...]

External consistency refers to consistency with other elements in the environment. External consistency extends the benefits of internal consistency across multiple, independent systems. It is more difficult to achieve because different systems rarely observe common design standards.” (Lidwell et al. 2010, p. 56)

Riprendendo l'esempio degli apparati, la coerenza estetica si ottiene presentando tutti gli apparati nella stessa forma, per esempio le singole voci d'apparato sono implementate come *pop-up* che compaiono sopra al testo edito, cliccando sulla porzione di testo interessata. La coerenza funzionale implica che con componenti simili l'utente possa interagire allo stesso modo e aspettandosi lo stesso comportamento, per esempio: cliccando sulla porzione di testo associata a una voce d'apparato, quest'ultima compare sempre nel testo edito invece che accanto o sotto, a prescindere che sia una voce dell'apparato critico o una nota di commento. La coerenza tra apparati diversi è una coerenza “interna” allo strumento informatico, mentre il già menzionato utilizzo della parentesi quadra, per dividere la lezione accolta a testo da quelle rifiutate all'interno di una voce dell'apparato critico, è un esempio di coerenza “esterna”.

Per rispettare il principio della coerenza e ottimizzare la scrittura di codice, emerge quanto venga concepire i componenti principali del modello concettuale come dei *widget*, cioè dei blocchi per la costruzione dell'interfaccia grafica, da poter riutilizzare per implementare la modalità di presentazione degli altri componenti. Si ripropone l'esempio degli apparati: realizzando solamente due widget, uno per l'apparato (*Apparatus*) e l'altro per la voce d'apparato (*Apparatus Entry*), si dispone di una base, eventualmente da adattare, per implementare tutti i diversi tipi di apparato.

Le indicazioni fin qui fornite per il layout sono molto generiche, e lasciano sostanzialmente carta bianca allo sviluppatore informatico. Tuttavia, in fase di progettazione può rivelarsi molto utile studiare le soluzioni grafiche adottate in altre edizioni scientifiche digitali e negli strumenti di visualizzazione generici esistenti,²² e, in virtù della “coerenza esterna” sopra descritta, conoscere le convenzioni tipografiche delle edizioni a stampa.

²² Come esempio di studio grafico, a mio avviso molto ben fatto, si rimanda a Di Pietro 2015: nel secondo capitolo (pp. 61-89), l'autrice illustra lo studio svolto per progettare e implementare l'interfaccia grafica della seconda versione del software EVT, rispondendo puntualmente a dei problemi comuni nel campo della visualizzazione di edizioni scientifiche digitali, come la riduzione della leggibilità quando si visualizzano sinotticamente un elevato numero di testi.

Infine, per completare il discorso sulla progettazione grafica, pare opportuno ricordare brevemente alcune pratiche consolidate nel campo del *web design* da prendere a riferimento anche per la realizzazione di strumenti per la visualizzazione di edizioni scientifiche digitali, che sono da un lato il *responsive design* e dall'altro l'accessibilità. Si presuppone che lo sviluppatore informatico conosca queste pratiche e le applichi nel quotidiano. È bene che anche lo studioso le conosca e ne comprenda l'importanza, per valutare più accuratamente l'idoneità di uno strumento informatico e per poterli proporre tra i *desiderata* quando commissiona la realizzazione di uno strumento informatico.

Con “responsive web design” si intende un insieme di metodologie dello sviluppo web, ideate per sviluppare in modo più rapido e pratico dei siti web che si adattino automaticamente alle dimensioni e alle caratteristiche dei diversi dispositivi con i quali vengono visualizzati. Per realizzare uno strumento informatico dal *design* flessibile e adattabile a diversi dispositivi, è consigliabile adottare tali metodologie, ricordando però che, nella maggior parte dei casi, l'utente-lettore consulta un'edizione scientifica digitale attraverso un personal computer, quindi da un dispositivo *desktop* invece che *mobile*. Questo perché un'edizione scientifica digitale è prima di tutto uno strumento di lavoro per l'utente che, realisticamente, per lavorare al meglio avrà bisogno di uno schermo di certe dimensioni e di non essere in movimento. L'approccio più ragionevole per lo sviluppo di uno strumento di visualizzazione è quello, dunque, di concentrarsi sulla progettazione della versione *desktop*, alla quale affiancare una versione *mobile*, eventualmente semplificata cioè con funzionalità ridotte e in cui presentare solo alcuni dei contenuti dell'edizione.

Con il termine “accessibilità”, invece, si fa riferimento a una serie di accorgimenti tecnici per rendere una risorsa web accessibile alle persone con disabilità. Un riferimento molto importante in questo ambito è la Web Accessibility Initiative²³ del W3C, che offre delle guide e degli strumenti dedicati. Tralasciando il fatto che in svariati casi un sito web deve essere accessibile per legge, l'implementazione di questi accorgimenti è a prescindere una buona pratica anche nel contesto della filologia digitale, per mettere le edizioni scientifiche digitali a disposizione di tutta la comunità di studiosi.

²³ <https://www.w3.org/WAI/>.

Conclusioni

Nel presente elaborato abbiamo tracciato una strategia di modellizzazione dei requisiti per lo sviluppo di strumenti informatici funzionali alla visualizzazione di edizioni scientifiche digitali. La strategia proposta si basa su due diverse soluzioni emerse in letteratura riguardo il problema della limitata usabilità o dell’“inadeguatezza”, secondo la mia definizione, degli strumenti informatici attualmente disponibili nel campo della filologia digitale. Da un lato, il bisogno da parte della comunità scientifica di stabilire dei “modelli editoriali” (Pierazzo 2019), dall’altro la necessità di fare riferimento alle pratiche che si sono consolidate nell’ingegneria del software, per sviluppare prodotti informatici robusti, durevoli e usabili (Del Grosso et al. 2017). La soluzione da me proposta, dunque, è frutto della convergenza di queste due prospettive, quella del filologo e quella dello sviluppatore informatico, ed è stata guidata da tre principali riflessioni. In primo luogo, un modello editoriale deve essere sviluppato in modo tale da poter guidare concretamente lo sviluppo di uno strumento informatico. In secondo luogo, non deve essere troppo astratto, ma indicare con precisione i *desiderata* dei filologi, né troppo specifico, dipendendo, per esempio, dalle tecnologie del momento, in modo da risultare flessibile e durevole nel tempo. Infine, un modello editoriale deve essere comprensibile sia per gli sviluppatori informatici sia per i filologi, affinché i secondi possano valutare l’adeguatezza non solo del modello stesso ma anche degli strumenti sviluppati in base ad esso.

La strategia di modellizzazione prevede che i modelli editoriali, inquadrati teoricamente rispetto all’approccio filologico in cui sono emersi, siano il punto di partenza per la definizione dei requisiti di uno strumento per la visualizzazione e siano sviluppati come modelli concettuali (o *domain models*). Applicando i principi del *domain-driven design* (Evans 2004), un modello editoriale permette di individuare i componenti dell’interfaccia grafica e di avviare la progettazione delle funzionalità di uno strumento di visualizzazione. La strategia di modellizzazione consente, dunque, di creare dei modelli concettuali che durino nel tempo, dai quali derivare di volta in volta dei modelli implementativi sulla base delle tecnologie del momento. Come illustrato ai capitoli 2 e 4, il lavoro di elaborazione di un modello editoriale, secondo la strategia di modellizzazione delineata, prevede due fasi, rispettivamente:

1. la formalizzazione come ontologia in linguaggio OWL 2, quindi la definizione dei componenti alla base di un particolare tipo di edizione e dei loro rapporti reciproci;
2. la definizione di una funzione di trasformazione, per tradurre l'ontologia OWL in UML e guidare in modo più dettagliato lo sviluppo dello strumento informatico, senza modificare la semantica espressa dall'ontologia.

Oltre alla definizione della strategia di modellizzazione in sé, la presente ricerca ha prodotto un modello concettuale di esempio, che descrive l'edizione critica inquadrata secondo l'approccio neolachmanniano, al fine di dimostrare in che cosa consistono concretamente le fasi di lavoro previste dalla strategia. Chiaramente il modello ottenuto necessita di essere ampliato e rivisto sia a livello di organizzazione gerarchica, sia a livello di definizioni, eliminando eventuali classi o proprietà superflue. Affinché la revisione sia efficace, occorrono diversi accorgimenti. In primo luogo, il modello deve essere testato, applicandolo nella formalizzazione integrale di esempi concreti di edizioni digitali. In secondo luogo, bisogna interrogarsi su come coinvolgere la comunità dei filologi in modo da raccogliere riscontri puntuali sulle classi e le proprietà alla base del modello e sulle rispettive definizioni. Oltre a invitare gli studiosi a lasciare osservazioni attraverso il sistema di segnalazioni (*issues*) di GitHub, dove l'ontologia è già pubblicata, una soluzione più efficace potrebbe essere la realizzazione di questionari. Affinché i questionari siano efficaci, occorre riflettere a monte sulle domande da porre. Si potrebbe chiedere, per esempio, di valutare l'appropriatezza di ogni definizione fornita nell'ontologia con dei valori predefiniti (come una scala da 1 a 5, dove 1 sta per “per niente appropriato” e 5 per “molto appropriato”) e di fornire dei suggerimenti, tra cui delle definizioni alternative o dei riferimenti bibliografici aggiuntivi.

Una volta revisionato e consolidato il modello concettuale di esempio, il passo successivo consisterà nello stabilire una funzione di trasformazione definitiva, partendo dagli spunti forniti al capitolo 4. Questa fase porterà probabilmente a un'ulteriore revisione dell'efficacia semantica del modello concettuale e a specificare alcuni aspetti più tecnici, che per ora sono stati omessi, ma che sono fondamentali per guidare in modo più dettagliato la progettazione dello strumento software. Tra questi c'è l'indicazione della cardinalità per le proprietà, ovvero quante istanze della classe *target* di una proprietà possono essere associate a un'istanza della classe dominio. La proprietà *preserves*, per esempio, permette di associare un'istanza di *Witness Carrier* a una sola istanza di

Witness, per indicare che un testimone ha un solo supporto fisico. La proprietà inversa, *is preserved in*, invece, permette di associare più istanze di Witness a un'istanza di Witness Carrier, dato che in un supporto fisico quale un codice manoscritto possono essere conservati i testimoni di opere diverse o, in casi rari, diversi testimoni della medesima opera. Per il momento nell'ontologia sono state indicate le cardinalità solo di alcune proprietà, mentre per molte altre questa analisi deve ancora essere svolta. I diagrammi UML prodotti con la funzione di trasformazione definitiva potranno essere pubblicati nello stesso *repository* dell'ontologia. Terminata questa fase, l'ontologia dovrebbe essere sufficientemente stabile da poter essere "modularizzata" (vd. 3.7), ovvero suddivisa in più ontologie separate. In questo modo ciascuna parte dell'ontologia potrebbe essere ulteriormente sviluppata in modo autonomo, facilitando il riuso del modello concettuale.

Infine, dato che l'obiettivo ultimo della strategia di modellizzazione è quello di guidare lo sviluppo di uno strumento informatico, l'efficacia del modello concettuale deve essere messa alla prova realizzando uno strumento per la visualizzazione di edizioni critiche digitali di approccio neo-lachmanniano. Per quest'ultima fase bisognerebbe coinvolgere, idealmente, diversi sviluppatori informatici il più possibile digiuni di competenze filologiche: se il modello risulta comprensibile e applicabile in modo corretto da chi non conosce il dominio applicativo, significa che è chiaro e ben fatto. Anche per quest'ultima fase di lavoro i dettagli operativi sono tutt'altro che banali e da valutare accuratamente. Bisogna non solo individuare delle persone che realizzino lo strumento, ma anche determinare come lo strumento verrà a sua volta valutato e quali siano i parametri per determinare il successo o il fallimento dello sviluppo. Il modello e lo strumento di visualizzazione, una volta pronti e stabili, potranno essere ulteriormente messi alla prova da parte della comunità di studiosi con dei casi d'uso: lo strumento informatico per visualizzare e pubblicare edizioni diverse, il modello per guidare lo sviluppo di strumenti in diversi progetti di edizione.

Tra i futuri sviluppi della presente ricerca, vi è innanzitutto la realizzazione di altri modelli concettuali, per rappresentare ulteriori tipologie di edizione, e, di conseguenza, l'implementazione di strumenti di visualizzazione completi, cioè che siano in grado di gestire molteplici tipi di edizione. Un ulteriore sviluppo, già descritto nel corso del secondo capitolo, consiste nell'applicare il modello concettuale, specialmente la parte dedicata al rapporto tra il testo edito e il testimone, nel campo delle

edizioni semantiche digitali, che sfruttano le tecnologie del web semantico per offrire i propri contenuti in forma di *linked open data*, favorendo un potenziale riuso dei dati (vd. 2.5.1). Le classi e le proprietà proposte in questa parte dell'ontologia di esempio si prestano a rendere "assertivo" (Vogeler 2019) non tanto il testo edito in sé, quanto la sua trasmissione e i problemi che pone, insieme al modo in cui i diversi studiosi hanno provato a risolverli e al lavoro filologico svolto dall'editore. Applicando la seconda parte dell'ontologia in quest'ottica, sarebbe possibile non solo descrivere i dati contenuti nel testo e nell'apparato critico di un'edizione in modo chiaro e verificabile, ma anche renderli interrogabili e confrontabili con i dati di altre edizioni.

Rispetto alla quantità di lavoro necessario per ultimare la strategia di modellizzazione, emerge quanto la ricerca condotta finora sia stata di tipo preliminare. Tuttavia, l'idea di fondo di combinare un approccio filologico con quello informatico rappresenta il punto di forza di questa strategia, che punta fortemente su un approccio collaborativo e interdisciplinare, per risolvere un problema complesso e di grande interesse.

Bibliografia

- Alhudithi, Ella. 2021. «Review of Voyant Tools: See through Your Text». *Language Learning & Technology* 25 (3): 43–50.
- Avalle, Arco Silvio. 1993. «La funzione del “punto di vista” nelle strutture oppositive binarie». *Lettere Italiane* 45 (2): 179–87.
- Barabucci, Gioele. 2016. «CATview». *Digital Medievalist* 10 (0). <https://doi.org/10.16995/dm.57>.
- Barabucci, Gioele, Elena Spadini, e Magdalena Turska. 2017. «Data vs. Presentation. What Is the Core of a Scholarly Digital Edition?» In *Advances in Digital Scholarly Editing*, 37–46. Sidestone Press. https://serval.unil.ch/notice/serval:BIB_09C6C598108A.
- Berners-Lee, Tim, James Hendler, e Ora Lassila. 2001. «The Semantic Web». *Scientific American* 284 (5): 34–43.
- Berté, Monica. 2015. «Giovanni Malpaghini Copista Di Petrarca?» *Giovanni Malpaghini Copista Di Petrarca?*, 205–16. <https://doi.org/10.1400/233574>.
- Bleeker, Elli, Bram Buitendijk, e Ronald Haentjens Dekker. 2019a. «From graveyard to graph: Visualisation of textual collation in a digital paradigm». *International Journal of Digital Humanities* 1 (giugno). <https://doi.org/10.1007/s42803-019-00012-w>.
- . 2019b. «Agree to disagree: Modelling co-existing scholarly perspectives on literary text». *Digital Scholarship in the Humanities* 34 (4): 844–54. <https://doi.org/10.1093/lhc/fqz061>.
- Bleier, Roman, Martina Bürgermeister, Helmut W. Klug, Frederike Neuber, e Gerlinde Schneider. 2018. *Digital Scholarly Editions as Interfaces*. BoD – Books on Demand.
- Boot, Peter, Anna Cappellotto, Wout Dillen, Franz Fischer, Aodhán Kelly, Andreas Mertgens, Anna-Maria Sichani, Elena Spadini, e Dirk Van Hulle, a c. di. 2017. *Advances in Digital Scholarly Editing: Papers Presented at the DiXiT Conferences in the Hague, Cologne, and Antwerp*. Leiden: Sidestone Press.

- Boot, Peter, Marijn Koolen, Hugh A. Cayless, Matteo Romanello, Iian Neill, Desmond Schmidt, Miller C. Prosser, et al. 2021. *Graph Data-Models and Semantic Web Technologies in Scholarly Digital Editing*. A cura di Elena Spadini, Francesca Tomasi, e Georg Vogeler. Vol. 15. Norderstedt: BoD. <http://www.uni-koeln.de/>.
- Bordalejo, Barbara. 2013. «The Texts We See and the Works We Imagine: the Shift of Focus of Textual Scholarship in the Digital Age». *Ecdotica*, fasc. 1/2013. <https://doi.org/10.7385/99160>.
- Borek, Luise, Quinn Dombrowski, Jody Perkins, e Christof Schöch. 2016. «TaDiRAH: a Case Study in Pragmatic Classification». *Digital Humanities Quarterly* 010 (1).
- Boschetti, Federico. 2018. *Copisti digitali e filologi computazionali*. Roma: CNR Edizioni. <http://eprints.bice.rm.cnr.it/17545/>.
- Boschetti, Federico, e Gloria Mugelli. 2021. «Il metodo Euporia per creare nuovi archivi digitali sulla tragedia greca». *FuturoClassico FCl*, fasc. 7: 83–113.
- Brunetti, Giuseppina. 2012. «L'edizione critica delle liriche medievali: considerazioni dall'Italia». *Perspectives médiévales. Revue d'épistémologie des langues et littératures du Moyen Âge*, fasc. 34 (settembre). <https://doi.org/10.4000/peme.1696>.
- Brusuelas, James H. 2021. «Scholarly Editing and AI: Machine Predicted Text and Herculanum Papyri». *Magazén*, fasc. 1 (giugno): JournalArticle_5056. <https://doi.org/10.30687/mag/2724-3923/2021/03/002>.
- Buzzetti, Dino. 2017. «Alle origini dell'Informatica Umanistica: Humanities Computing e/o Digital Humanities.» In *Il museo virtuale dell'informatica archeologica. Una collaborazione tra l'Accademia nazionale dei Lincei e il Consiglio nazionale delle ricerche. Atti della «Segnatura»*. Roma, 13:9. <http://web.dfc.unibo.it/buzzetti/dbuzzetti/pubblicazioni/originiIU.pdf>.
- Buzzetti, Dino, e Jerome McGann. 2006. «Critical editing in a digital horizon». *Electronic textual editing*, 53–73.
- Buzzoni, Marina, e Roberto Rosselli Del Turco. 2016. «Evolution or Revolution? Digital Philology and Medieval Texts: History of the Discipline and a Survey of Some Italian Projects». *Mittelalterphilologien heute. Medieval Philologies Today*, 1–31.

- Carbè, Emmanuela. 2023. *Digitale d'autore: Macchine, archivi e letterature*. Firenze University Press. <https://doi.org/10.36253/979-12-215-0023-3>.
- Cayless, Hugh. 2018. «Critical editions and the data model as interface». *Digital Scholarly Editions as Interfaces*, 249–63.
- Ciotti, Fabio. 2014. «Digital Literary and Cultural Studies: State of the Art and Perspectives». *Between* 4 (8). <https://doi.org/10.13125/2039-6597/1392>.
- . 2018. «A Formal Ontology for the Text Encoding Initiative». *Umanistica Digitale*, fasc. 3 (novembre). <https://doi.org/10.6092/issn.2532-8816/8174>.
- Ciotti, Fabio, Elena Corradini, Elisa Cugliana, Giulia D'Agostino, Lorenzo Ferroni, Franz Fischer, Maurizio Lana, et al. 2022. «Manifesto per le edizioni scientifiche digitali». *Umanistica Digitale*, fasc. 12 (giugno): 103–8. <https://doi.org/10.6092/issn.2532-8816/14814>.
- Ciotti, Fabio, e Francesca Tomasi. 2016. «Formal Ontologies, Linked Data, and TEI Semantics». *Journal of the Text Encoding Initiative*, fasc. Issue 9 (settembre). <https://doi.org/10.4000/jtei.1480>.
- Cipolla, Adele. 2018. *Digital Philology: New Thoughts on Old Questions*. libreriauniversitaria.it Edizioni.
- Ciula, Arianna, e Øyvind Eide. 2014. «Reflections on cultural heritage and digital humanities: modelling in practice and theory». In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, 35–41. DATeCH '14. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/2595188.2595207>.
- Cools, Hans, e Roberta Padlina. 2021. «Formal Semantics for Scholarly Editions». In , a cura di Elena Spadini, Francesca Tomasi, e Georg Vogeler, 15:97–124. Norderstedt: BoD. <http://www.uni-koeln.de/>.
- Daquino, Marilena, Francesca Giovannetti, e Francesca Tomasi. 2019. «Linked Data per le edizioni scientifiche digitali. Il workflow di pubblicazione dell'edizione semantica del quaderno di appunti di Paolo Bufalini». *Umanistica Digitale*, fasc. 7 (dicembre). <https://doi.org/10.6092/issn.2532-8816/9091>.
- Del Grosso, Angelo, D. Albanesi, Emiliano Giovannetti, e Simone Marchi. 2016. «Defining the Core Entities of an Environment for Textual Processing in Literary Computing». In

- Digital Humanities 2016: Conference Abstracts*, 771–75. Jagiellonian University & Pedagogical University, Kraków. <https://dh2016.adho.org/abstracts/425>.
- Del Grosso, Angelo Mario, Emiliano Giovannetti, e Simone Marchi. 2017. «The Importance of Being... Object-Oriented: Old Means for New Perspectives in Digital Textual Scholarship». In *Advances in Digital Scholarly Editing*, 269–74. Leiden: Sidestone Press.
- Del Turco, Roberto Rosselli. 2012. «After the Editing Is Done: Designing a Graphic User Interface for Digital Editions». *Digital Medievalist* 7 (0). <https://doi.org/10.16995/dm.30>.
- . 2019. «Designing an advanced software tool for Digital Scholarly Editions: The inception and development of EVT (Edition Visualization Technology)». *Textual Cultures* 12 (2): 91–111.
- Del Turco, Roberto Rosselli, e Chiara Di Pietro. 2017. «Edition Visualization Technology 2.0 Affordable DSE publishing, support for critical editions, and more». In *Advances in Digital Scholarly Editing*, 275–81. Leiden: Sidestone Press.
- Di Pietro, Chiara. 2015. «EVT per le edizioni critiche digitali: progettazione e sviluppo di una nuova GUI basata sullo schema progettuale MVC». Tesi di Laurea Magistrale, Pisa, Italia: Università di Pisa. <https://etd.adm.unipi.it/t/etd-03292016-203644/>.
- Doerr, Martin. 2003. «The CIDOC Conceptual Reference Module: An Ontological Approach to Semantic Interoperability of Metadata». *AI Magazine* 24 (3): 75–75. <https://doi.org/10.1609/aimag.v24i3.1720>.
- Doran, Paul, Valentina Tamma, e Luigi Iannone. 2007. «Ontology module extraction for ontology reuse: an ontology engineering perspective». In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 61–70. CIKM '07. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/1321440.1321451>.
- Driscoll, Matthew James, e Elena Pierazzo, a c. di. 2017. *Digital Scholarly Editing: Theories and Practices*. *Digital Scholarly Editing: Theories and Practices*. Digital Humanities Series. Cambridge: Open Book Publishers. <http://books.openedition.org/obp/3381>.
- Eide, Øyvind. 2014. «Ontologies, Data Modeling, and TEI». *Journal of the Text Encoding Initiative*, fasc. Issue 8 (dicembre). <https://doi.org/10.4000/jtei.1191>.

- Eide, Øyvind, e Christian-Emil Smith Ore. 2018. «Ontologies and data modeling». In *The Shape of Data in Digital Humanities*. Routledge.
- Evans, Eric. 2004. *Domain-driven design : tackling complexity in the heart of software*. Addison-Wesley.
- . 2015. «DDD Reference». <https://www.domainlanguage.com/ddd/reference/>.
- Fiormonte, Domenico. 2003. *Scrittura e filologia nell'era digitale / Domenico Fiormonte*. Nuova didattica Arte e letteratura. Torino: Bollati Boringhieri.
- Franzini, G., S. Mahony, e M. Terras. 2016. «A Catalogue of Digital Editions». In: *Pierazzo, E and Driscoll, M, (Eds.) Digital Scholarly Editing: Theories and Practices. (Pp. 161-182). Open Book Publishers: Cambridge, UK. (2016)*, a cura di E. Pierazzo e M. Driscoll, 4:161–82. Cambridge, UK: Open Book Publishers. <https://doi.org/10.11647/OBP.0095>.
- Franzini, Greta, Melissa Terras, e Simon Mahony. 2019. «Digital Editions of Text: Surveying User Requirements in the Digital Humanities». *J. Comput. Cult. Herit.* 12 (1): 1:1-1:23. <https://doi.org/10.1145/3230671>.
- Gabay, Simone, Jean-Baptiste Camps, Ariane Pinche, e Nicola Carboni. 2021. «SegmOnto, A Controlled Vocabulary to Describe the Layout of Pages». Paris/Genève. <https://github.com/SegmOnto>.
- Gamma, Erich, Richard Helm, Ralph Johnson, e John Vlissides. 1994. *Design Patterns: Elements of Reusable Object-Oriented Software*. Pearson Education.
- Giovannetti, Francesca, e Francesca Tomasi. 2022. «Linked data from TEI (LIFT): A Teaching Tool for TEI to Linked Data Transformation». *Digital Humanities Quarterly* 016 (2).
- Gratta, Riccardo Del, Federico Boschetti, Angelo Mario Del Grosso, Luigi Bambaci, e Simone Zenzaro. 2022. «La Filologia come sistema dinamico». *Umanistica Digitale*, fasc. 13 (ottobre): 1–20. <https://doi.org/10.6092/issn.2532-8816/13684>.
- Griesinger, Christian. 2020. «Review of the Tübinger System von Textverarbeitungs-Programmen (TUSTEP)». *RIDE (blog)*. gennaio 2020. <https://doi.org/10.18716/ride.a.11.2>.
- Guarino, Nicola. 1998. *Formal Ontology in Information Systems: Proceedings of the First International Conference (FOIS'98)*, June 6-8, Trento, Italy. IOS Press.

- Guarino, Nicola, Daniel Oberle, e Steffen Staab. 2009. «What Is an Ontology?» In *Handbook on Ontologies*, a cura di Steffen Staab e Rudi Studer, 1–17. International Handbooks on Information Systems. Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-540-92673-3_0.
- Hockey, Susan. 2004. «The history of humanities computing». *A companion to digital humanities*, 3–19.
- Hodel, Tobias, Anna Janka, e Jonas Widmer. 2022. «Digital Mappa – Simple and Web-Based Annotations». *RIDE* (blog). dicembre 2022. <https://doi.org/10.18716/ride.a.15.1>.
- Italia, Paola. 2020. *Editing Duemila. Per una filologia dei testi digitali*. Salerno.
- Jones, Mason, Francesco Romano, e Abidalrahman Mohd. 2022. «Machine Learning in Textual Criticism: An examination of the performance of supervised machine learning algorithms in reconstructing the text of the Greek New Testament». In *2022 7th International Conference on Machine Learning Technologies (ICMLT)*, 1–5. ICMLT 2022. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/3529399.3529400>.
- Kiessling, Benjamin, Robin Tissot, Peter Stokes, e Daniel Stökl Ben Ezra. 2019. «eScriptorium: An Open Source Platform for Historical Document Analysis». In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, 2:19–19. <https://doi.org/10.1109/ICDARW.2019.10032>.
- Kluzinski, Philip, e John Moore. 2016. *Mastering JavaScript Single Page Application Development*. Packt Publishing Ltd.
- Leblanc, Elina. 2020. «Omeka Classic. Un environnement de recherche pour les éditions scientifiques numériques». *RIDE* (blog). gennaio 2020. <https://doi.org/10.18716/ride.a.11.3>.
- Leonardi, Lino. 2007. «Filologia elettronica tra conservazione e ricostruzione», 11.
- . 2021. «Filologia digitale del Medioevo italiano». *Griseldaonline* 20 (2): 77–89. <https://doi.org/10.6092/issn.1721-4777/12817>.
- Lidwell, William, Kritina Holden, e Jill Butler. 2010. *Universal Principles of Design, Revised and Updated: 125 Ways to Enhance Usability, Influence Perception, Increase Appeal, Make Better Design Decisions, and Teach Through Design*. Rockport Publishers.

- Maas, Paul. 1927. *Textkritik*. Einleitung in die Altertumswissenschaft Bd. 1, Hft. 2. Leipzig: B.G. Teubner.
- Madasu, Vamsi, Trinadh Venna, e Tarik Eltaeib. 2015. «SOLID Principles in Software Architecture and Introduction to RESM Concept in OOP». *Journal of Engineering Science and Technology* 2 (marzo): 3159–40.
- Mancinelli, Tiziana, e Elena Pierazzo. 2020. *Che cos'è un'edizione scientifica digitale*. Carocci.
- Martignano, Chiara. 2017. «Progettazione e sviluppo di un apparato critico modellato sulla tradizione a stampa in EVT». Pisa, Italia: Università di Pisa. <https://etd.adm.unipi.it/t/etd-06132017-202934/>.
- . 2021. «Un modello concettuale per favorire lo sviluppo e il riutilizzo di app per edizioni digitali». *Umanistica Digitale*, fasc. 10 (settembre): 71–88. <https://doi.org/10.6092/issn.2532-8816/12620>.
- . 2023. «On why and how we should build a catalogue of software products for digital scholarly editing». In *La memoria digitale: forme del testo e organizzazione della conoscenza*. Atti del XII Convegno Annuale AIUCD. Siena: Università degli Studi di Siena. <http://www.aiucd2023.unisi.it/atti/>.
- McGann, Jerome. 1998. «Textual Scholarship, Textual Theory, and the Uses of Electronic Tools: A Brief Report on Current Undertakings». *Victorian Studies* 41 (4): 609–19.
- . 2008. «The Future is Digital». *Journal of Victorian Culture* 13 (1): 80–88. <https://doi.org/10.3366/E135555020800009X>.
- Mertgens, Andreas. 2020. «Ediarum. A Toolbox for Editors and Developers». *RIDE*, gennaio. <https://doi.org/10.18716/ride.a.11.4>.
- Modiano, Raimonda, Leroy Searle, e Peter L. Shillingsburg. 2004. *Voice, Text, Hypertext: Emerging Practices in Textual Studies*. University of Washington Press.
- Moretti, Franco. 2005. *La letteratura vista da lontano*. Vol. 865. Einaudi.
- Muehlberger, Guenter, Louise Seaward, Melissa Terras, Sofia Ares Oliveira, Vicente Bosch, Maximilian Bryan, Sebastian Colutto, et al. 2019. «Transforming scholarship in the archives through handwritten text recognition: Transkribus as a case study». *Journal of Documentation* 75 (5): 954–76. <https://doi.org/10.1108/JD-07-2018-0114>.

- Noy, Natalya F, e Deborah L McGuinness. 2001. «Ontology development 101: A guide to creating your first ontology». Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880. <http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html>.
- Nury, Elisa. 2018. «Automated Collation and Digital Editions: from Theory to Practice». Theses, King's College London. <https://hal.archives-ouvertes.fr/tel-02493805>.
- Nury, Elisa, e Elena Spadini. 2020. «From Giant Despair to a New Heaven: The Early Years of Automatic Collation». *It - Information Technology* 62 (2): 61–73. <https://doi.org/10.1515/itit-2019-0047>.
- Nussbaumer, Philipp, e Bernhard Haslhofer. 2007. «Putting the CIDOC CRM into Practice - Experiences and Challenges». Technical Report. University of Vienna. settembre 2007. <https://eprints.cs.univie.ac.at/404/>.
- Ore, Christian-Emil, e Øyvind Eide. 2009. «TEI and cultural heritage ontologies: Exchange of information?» *Literary and Linguistic Computing* 24 (2): 161–72. <https://doi.org/10.1093/llc/fqp010>.
- Orlandi, Tito. 1990. *Informatica umanistica*. Studi superiori NIS Lettere 0078. Roma: NIS.
- . 2010. *Informatica testuale: teoria e prassi*. Manuali Laterza 308. Roma [etc.: GLF editori Laterza.
- Pan, Jeff Z., Steffen Staab, Uwe Aßmann, Jürgen Ebert, e Yuting Zhao, a c. di. 2013. *Ontology-Driven Software Development*. Berlin Heidelberg: Springer-Verlag. <https://doi.org/10.1007/978-3-642-31226-7>.
- Peroni, Silvio, e David Shotton. 2012. «FaBiO and CiTO: Ontologies for Describing Bibliographic Resources and Citations». *Journal of Web Semantics* 17 (dicembre): 33–43. <https://doi.org/10.1016/j.websem.2012.08.001>.
- Peroni, Silvio, David Shotton, e Fabio Vitali. 2012. «Scholarly publishing and linked data: describing roles, statuses, temporal and contextual extents». In *Proceedings of the 8th International Conference on Semantic Systems*, 9–16. I-SEMANTICS '12. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/2362499.2362502>.

- Pierazzo, Elena. 2011. «A rationale of digital documentary editions». *Literary and Linguistic Computing* 26 (4): 463–77. <https://doi.org/10.1093/lc/fqr033>.
- . 2014a. «Digital documentary editions and the others». *Scholarly Editing* 35: 1–23.
- . 2014b. *Digital Scholarly Editing: Theories, Models and Methods*. <http://hal.univ-grenoble-alpes.fr/hal-01182162>.
- . 2017. «Modelling Digital Scholarly Editing: From Plato to Heraclitus». In *Digital Scholarly Editing: Theories and Practices*, a cura di Matthew James Driscoll, 41–58. Digital Humanities Series. Cambridge: Open Book Publishers. <http://books.openedition.org/obp/3399>.
- . 2019. «What Future for Digital Scholarly Editions? From Haute Couture to Prêt-à-Porter». *International Journal of Digital Humanities*, maggio. <https://doi.org/10.1007/s42803-019-00019-3>.
- Pöckelmann, Marcus, André Medek, Jörg Ritter, e Paul Molitor. 2022. «LERA - An interactive platform for synoptical representations of multiple text witnesses». *Digital Scholarship in the Humanities*, giugno. <https://doi.org/10.1093/lc/fqac021>.
- Porter, Dot, Alberto Campagnolo, e Erin Connelly. 2017. «VisColl: A New Collation Tool for Manuscript Studies». In *Kodikologie und Paläographie im Digitalen Zeitalter 4 – Codicology and Palaeography in the Digital Age 4*, a cura di Hannah Busch, Franz Fischer, e Patrick Sahle, 11:81–100. Norderstedt: Books on Demand. <http://nbn-resolving.de/urn:nbn:de:hbz:38-77742>.
- Raboni, Giulia, e Paola Italia. 2010. *Che cos'è la filologia d'autore*. Le bussole 408. Roma: Carocci.
- Rahtz, Sebastian 2006. *Storage, Retrieval, and Rendering*. Modern Language Association of America. <https://ora.ox.ac.uk/objects/uuid:a9063068-f02e-442d-8fcf-1df18149ed0f>.
- Rasmussen, Krista Stinne Greve. 2017. «Reading or Using a Digital Edition? Reader Roles in Scholarly Editions». In *Digital Scholarly Editing: Theories and Practices*, a cura di Matthew James Driscoll e Elena Pierazzo, 119–33. Digital Humanities Series. Cambridge: Open Book Publishers. <http://books.openedition.org/obp/3406>.
- Ries, Thorsten. 2023. «Digital Archives, Digital Cultures: An Opportunity for German Studies». *German Studies Review* 46 (1): 136–38. <https://doi.org/10.1353/gsr.2023.0013>.

- Riva, Pat, Patrick Le Boeuf, e Maja Žumer. 2017. «IFLA Library Reference Model: Un modello concettuale per le informazioni bibliografiche». A cura di Istituto centrale per il catalogo unico delle biblioteche italiane e per le informazioni bibliografiche, novembre 2020, Roma: ICCU. <https://repository.ifla.org/handle/123456789/44>.
- Robinson, Peter. 2005. «Current Issues in Making Digital Editions of Medieval Texts—or, Do Electronic Scholarly Editions Have a Future?» *Digital Medievalist* 1 (0). <https://doi.org/10.16995/dm.8>.
- . 2017. «Some principles for making collaborative scholarly editions in digital form». *Digital Humanities Quarterly* 011 (2).
- Rodríguez, Pilar Arrabal. 2022. «TEITOK, a Visual Solution for XML/TEI Encoding: Editing, Annotating and Hosting Linguistic Corpora». *RIDE* (blog). dicembre 2022. <https://doi.org/10.18716/ride.a.15.5>.
- Roeder, Torsten. 2020. «Juxta Web Service, LERA, and Variance Viewer. Web Based Collation Tools for TEI». *RIDE* (blog). gennaio 2020. <https://doi.org/10.18716/ride.a.11.5>.
- Roncaglia, Gino. 2011. *La quarta rivoluzione: Sei lezioni sul futuro del libro*. Gius. Laterza & Figli Spa.
- Sahle, Patrick. 2016. «What is a Scholarly Digital Edition?» In *Digital Scholarly Editing*, a cura di Matthew James Driscoll e Elena Pierazzo, 1^a ed., 4:19–40. Theories and Practices. Open Book Publishers. <https://www.jstor.org/stable/j.ctt1fzhh6v.6>.
- Scheithauer, Hugo, Alix Chagué, e Laurent Romary. 2021. «From eScriptorium to TEI Publisher». In *Brace your digital scholarly edition!*
- Schmidt, Desmond, e Robert Colomb. 2009. «A data structure for representing multi-version texts online». *International Journal of Human-Computer Studies* 67 (6): 497–514. <https://doi.org/10.1016/j.ijhcs.2009.02.001>.
- Schmiedehausen, Kim. 2018. «Single Page Application Architecture with Angular», 48.
- Schreibman, Susan, Raymond George Siemens, e John Unsworth, a c. di. 2004. *A Companion to Digital Humanities*. Blackwell Companions to Literature and Culture 26. Malden, MA: Blackwell Pub.
- Shadbolt, N., T. Berners-Lee, e W. Hall. 2006. «The Semantic Web Revisited». *IEEE Intelligent Systems* 21 (3): 96–101. <https://doi.org/10.1109/MIS.2006.62>.

- Shillingsburg, Peter. 1996. *Scholarly Editing in the Computer Age: Theory and Practice*. University of Michigan Press.
- . 2006. *From Gutenberg to Google: Electronic Representations of Literary Texts*. Cambridge University Press.
- . 2013. «Development Principles for Virtual Archives and Editions». *Center for Textual Studies and Digital Humanities Publications*, gennaio. https://ecommons.luc.edu/ctsdh_pubs/4.
- Siemens, R., M. Timney, C. Leitch, C. Koolen, A. Garnett, e with the ETCL, INKE, and PKP Research Groups. 2012. «Toward Modeling the Social Edition: An Approach to Understanding the Electronic Scholarly Edition in the Context of New and Emerging Social Media». *Literary and Linguistic Computing* 27 (4): 445–61. <https://doi.org/10.1093/llc/fqs013>.
- Silva, António Rito, e Manuel Portela. 2014. «TEI4LdoD: Textual Encoding and Social Editing in Web 2.0 Environments». *Journal of the Text Encoding Initiative*, fasc. Issue 8 (dicembre). <https://doi.org/10.4000/jtei.1171>.
- Smiraglia, Richard P. 2001. *The Nature of «A Work»: Implications for the Organization of Knowledge*. Scarecrow Press.
- Spadini, Elena. 2017. «A Spoonful of Sugar: Encoding and Publishing in the Classroom». In *DiXiT Workshop: The Educational and Social Impact of Digital Scholarly Editions*. <https://pure.knaw.nl/portal/en/publications/7ca2a6de-d35e-4c1f-88c7-819b3bb26f6e>.
- Stadler, Peter, Anne Ferger, e Daniel Röwenstrunk. 2022. «From OxGarage to TEIGarage and MEIGarage». settembre 9. <https://doi.org/10.5281/zenodo.7061525>.
- Tarrant, Richard. 2016. *Texts, Editors, and Readers: Methods and Problems in Latin Textual Criticism*. Cambridge University Press.
- Tomasi, Francesca, Fabio Ciotti, Maurizio Lana, Fabio Vitali, Silvio Peroni, e Diego Magro. 2013. «Dialogue and Linking between TEI and Other Semantic Models». In *The Linked TEI: Text Encoding in the Web*, 145–58. Roma: DIGILAB Sapienza University & TEI Consortium. <https://hdl.handle.net/11585/185113>.
- Tractinsky, N, A.S Katz, e D Ikar. 2000. «What is beautiful is usable». *Interacting with Computers* 13 (2): 127–45. [https://doi.org/10.1016/S0953-5438\(00\)00031-X](https://doi.org/10.1016/S0953-5438(00)00031-X).

- Varvaro, Alberto, e Marcello Cherchi. 1999. «The “New Philology” from an Italian Perspective». *Text* 12: 49–58.
- Vogeler, Georg. 2019. «The ‘Assertive Edition’». *International Journal of Digital Humanities* 1 (2): 309–22. <https://doi.org/10.1007/s42803-019-00025-5>.
- White, Alexander. 2011. *The Elements of Graphic Design, Second Edition*. Allworth Press.
- Wilkinson, Mark D., Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, et al. 2016. «The FAIR Guiding Principles for Scientific Data Management and Stewardship». *Scientific Data* 3 (1): 160018. <https://doi.org/10.1038/sdata.2016.18>.
- Winslow, Sean M, Gerlinde Schneider, Roman Bleier, Christian Steiner, Christopher Pollin, e Georg Vogeler. 2019. «Ontologies in the Digital Repository: Metadata Integration, Knowledge Management and Ontology-Driven Applications». In *Joint Ontology Workshops*. Graz: CEUR Workshops Proceedings.
- Zaccarello, Michelangelo. 2018. «Progetto Di Un Osservatorio Permanente Sulle Edizioni Digitali Di Autori Italiani (OPEDIt). Prime Indagini Sulle Pratiche Di Digitalizzazione e Sull’autorevolezza Dell’edizione Di Testi Letterari Italiani in Formato Elettronico - BOLOGNA». *Prassi Ecdotiche Della Modernità Letteraria*, fasc. 3. <https://doi.org/10.13130/2499-6637/9491>.

Risorse informatiche

- A Catalogue of Digital Editions** Franzini, Greta. s.d. «Dig-Ed-Cat». Consultato 28 maggio 2023. <https://dig-ed-cat.acdh.oeaw.ac.at/>.
- A catalog of Digital Scholarly Editions** Sahle, Patrick. s.d. «A catalog of Digital Scholarly Editions». Consultato 17 agosto 2020. <http://www.digitale-edition.de/>.
- CAO** Giovannetti, Francesca. 2019. «The Critical Apparatus Ontology (CAO)». OWL 2. <https://fgiovannetti.github.io/cao/>.
- CIDOC CRM v. 7.1.2** Bekiari, Chryssoula, George Bruseker, Erin Canning, Martin Doerr, Philippe Michon, Christian-Emil Ore, Stephen Stead, e Athanasios Velios. 2022a. «Definition of the CIDOC Conceptual Reference Model v. 7.1.2».

- CIDOC CRM v. 7.2.2** Bekiari, Chryssoula, George Bruseker, Erin Canning, Martin Doerr, Philippe Michon, Christian-Emil Ore, Stephen Stead, e Athanasios Velio. 2022b. «Definition of the CIDOC Conceptual Reference Model v. 7.2.2».
- CIDOC CRM v. 7.1.2 Classes & Properties** «Classes & Properties Declarations of CIDOC-CRM version: 7.1.2». 2022. giugno 2022. https://cidoc-crm.org/html/cidoc_crm_v7.1.2.html.
- DoCO** Constantin, Alexandru, Silvio Peroni, Steve Pettifer, David Shotton, e Fabio Vitali. 2016. «The Document Components Ontology (DoCO)». A cura di Oscar Corcho. *Semantic Web* 7 (2): 167–81. <https://doi.org/10.3233/SW-150177>.
- FaBiO** Peroni, Silvio, e David Shotton. 2019. «The FRBR-aligned Bibliographic Ontology (FaBiO)». OWL 2. <http://www.sparontologies.net/ontologies/fabio>.
- HiCO** Daquino, Marilena. 2020. «The Historical Context Ontology (HiCO)». OWL 2. <https://w3id.org/hico/>.
- LexiconSE** Dillen, Wout. 2020. «Lexicon of Scholarly Editing». Lexicon of Scholarly Editing. 2020. <https://lexiconse.uantwerpen.be/lexicon.html>.
- LRMoo v. 0.9** Bekiari, Chryssoula, Martin Doerr, Patrick Le Boeuf, e Pat Riva, a c. di. 2022. «LRMoo (formerly FRBROO) object-oriented definition and mapping from IFLA LRM». https://cidoc-crm.org/frbroo/sites/default/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf.
- MDA Guide** «Object Management Group Model Driven Architecture (MDA) Guide Rev. 2.0». 2014. <https://www.omg.org/cgi-bin/doc?ormsc/14-06-01>.
- ODM v. 1.1** «Ontology Definition Metamodel Specification Version 1.1». 2014. <https://www.omg.org/spec/ODM>.
- OWL 1** McGuinness, Deborah L, e Frank Van Harmelen. s.d. «OWL Web Ontology Language Guide». Consultato 30 aprile 2023. <https://www.w3.org/TR/2004/REC-owl-guide-20040210/>.
- OWL 2** «OWL 2 Web Ontology Language Document Overview (Second Edition)». 2012. W3C OWL Working Group. <https://www.w3.org/TR/owl2-overview/>.

- OWL 2 Specification** Motik, Boris, Peter Patel-Schneider, e Bijan Parsia. s.d. «OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition)». Consultato 21 maggio 2023. <https://www.w3.org/TR/2012/REC-owl2-syntax-20121211/>.
- PLS** Roelli, Philipp, e Caroline Macé. 2015. «Parvum Lexicon Stemmatologicum». 13 novembre 2015. <https://wiki.helsinki.fi/display/stemmatology/Parvum+lexicon+stemmatologicum>.
- PROV-O** Lebo, Timothy, Satya Sahoo, e Deborah L McGuinness. 2013. «PROV-O: The PROV Ontology». OWL 2. <https://www.w3.org/TR/prov-o/>.
- SE** Cools, Hans. 2020. «Scholarly Editing Ontology». OWL 2. e-editiones. <https://e-editiones.ch/ontology/scholarly-editing>.
- SPAR Ontologies** Peroni, Silvio, e David Shotton. 2018. «The SPAR Ontologies». In *The Semantic Web – ISWC 2018*, a cura di Denny Vrandečić, Kalina Bontcheva, Mari Carmen Suárez-Figueroa, Valentina Presutti, Irene Celino, Marta Sabou, Lucie-Aimée Kaffee, e Elena Simperl, 11137:119–36. Lecture Notes in Computer Science. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-00668-6_8.
- Systems and Software Engineering Vocabulary** «ISO/IEC/IEEE International Standard - Systems and software engineering–Vocabulary». 2017. *ISO/IEC/IEEE 24765:2017(E)*, 1–541. <https://doi.org/10.1109/IEEESTD.2017.8016712>.
- TaDiRAH** «ACDH_CH_Vocabularies: TaDiRAH: Taxonomy of Digital Research Activities in the Humanities». s.d. Consultato 20 febbraio 2023. <https://vocab.sdariah.eu/tadirah/en/>.
- TaDiRAH Repository** «TaDiRAH - Taxonomy of Digital Research Activities in the Humanities (v. 0.5.1, 05/2014)». (2014) 2022. Digital Humanities Taxonomy Group. <https://github.com/dhtaxonomy/TaDiRAH/blob/b454077dc5cb543c88e14811ec56e33be548d4ff/readme.md>.
- TAPoR** Rockwell, Geoffrey, Stéfan Sinclair, e Milena Radzikowska. 2020. «Text Analysis Portal for Research (TAPoR) Archive». ERA. 1 gennaio 2020. <https://doi.org/10.7939/r3-tnha-4924>.
- TAPoR website** «TAPoR». s.d. Consultato 20 ottobre 2022. <https://tapor.ca/home>.

TEI P5 Guidelines Text Encoding Initiative Consortium. 2023. «P5: Guidelines for Electronic Text Encoding and Interchange». <https://tei-c.org/release/doc/tei-p5-doc/en/html/index.html>.

UML v. 2.5.1 «Unified Modeling Language Specification Version 2.5.1». 2017. <https://www.omg.org/spec/UML/>.

Web Annotation DM Sanderson, Robert, Paolo Ciccarese, e Benjamin Young. 2017a. «Web Annotation Data Model. W3C Recommendation 23 February 2017». 23 febbraio 2017. <https://www.w3.org/TR/annotation-model/>.

Web Annotation Vocabulary Sanderson, Robert, Paolo Ciccarese, e Benjamin Young. 2017b. «Web Annotation Vocabulary». <https://www.w3.org/TR/annotation-vocab/>.

Riferimenti bibliografici alla base del modello concettuale di esempio

Avalle, D'Arco Silvio. 1978. *Principi di critica testuale*. 2. ed. riveduta e Corretta. Vulgares eloquentes Collana di testi e manuali per la scuola universitaria 0007. Padova: Antenore.

Barbi, Michele. 1977. *La nuova filologia e l'edizione dei nostri scrittori da Dante al Manzoni*. 3. ed. Nuova biblioteca. Firenze: Sansoni.

Beltrami, Pietro G. 2010. *A che serve un'edizione critica?: leggere i testi della letteratura romanza medievale*. Itinerari. Filologia e critica letteraria. Bologna: Il Mulino.

———. 2017. *La filologia romanza: profilo linguistico e letterario*. Strumenti. Linguistica e critica letteraria. Bologna: Il mulino.

Brambilla Ageno, Franca. 1967. *L'edizione critica dei testi volgari*.

———. 1984. *L'edizione critica dei testi volgari*. 2. ed. riv. e Ampl. Medioevo e umanesimo 0022. Padova: Antenore.

Chiesa, Paolo. 2002. *Elementi di critica testuale*. Pàtron.

———. 2012. *Elementi di critica testuale*. 2. ed. Testi e manuali per l'insegnamento universitario del latino 72. Bologna: Pàtron.

Contini, Gianfranco. 2014. *Filologia*. Introduzioni. Filologia e critica letteraria. Bologna: Il mulino.

- Dillen, Wout. 2020. «Lexicon of Scholarly Editing». *Lexicon of Scholarly Editing*. 2020. <https://lexiconse.uantwerpen.be/lexicon.html>.
- Leonardi, Lino. 2022. *Filologia romanza. Vol. 1: Critica del testo*. Vol. 1. 3 voll. Le Monnier Università.
- Montanari, Elio. 2003. *La critica del testo secondo Paul Maas: testo e commento*. Millennio medievale 41. Firenze: SISMEL Edizioni del Galluzzo, Sismel, SISMEL, Edizioni del Galluzzo.
- Pasquali, Giorgio. 1974. *Storia della tradizione e critica del testo*. Gli Oscar Studio. Milano: Mondadori.
- Roelli, Philipp, a c. di. 2020. *Handbook of Stemmataology: History, Methodology, Digital Approaches*. De Gruyter. <https://doi.org/10.1515/9783110684384>.
- Roelli, Philipp, e Caroline Macé. 2015. «Parvum Lexicon Stematologicum». 13 novembre 2015. <https://wiki.helsinki.fi/display/stemmatology/Parvum+lexicon+stemmatologicum>.
- Stussi, Alfredo. 1994. *Introduzione agli studi di filologia italiana*. 3. ed. Strumenti. Linguistica e critica letteraria. Bologna: Il mulino.
- . 2006. *Fondamenti di critica testuale*. Il Mulino.
- . 2011. *Introduzione agli studi di filologia italiana*. 4. ed. Manuali. Filologia e critica letteraria. Bologna: Il mulino.
- Timpanaro, Sebastiano. 2003. *La genesi del metodo del Lachmann*. UTET libreria.
- Trovato, Paolo. 2017. *Everything You Always Wanted to Know about Lachmann's Method: A Non-Standard Handbook of Genealogical Textual Criticism in the Age of Post-Structuralism, Cladistics, and Copy-Text*. Revised edition. *Storie e Linguaggi* 7. Padova: Libreriauniversitaria.it edizioni.

Appendice A

Strumenti informatici per la filologia digitale

In questa appendice vengono presentati in forma di elenco i prodotti software e i *tool* informatici che possono essere usati per realizzare edizioni scientifiche digitali (ESD) analizzati nel capitolo 1. L'elenco include sia strumenti creati appositamente per essere applicati nel campo della filologia digitale, prodotti perlopiù da studiosi e ricercatori del settore, sia strumenti *general purpose* che sono stati largamente adottati in questo campo.

L'appendice è suddivisa in tre sezioni principali, che corrispondono alle tre macro-categorie che sono state individuate nel corso della ricognizione: gli strumenti pensati esclusivamente per la visualizzazione (o pubblicazione) di una ESD; gli strumenti che possono essere usati sia per elaborare sia per pubblicare una ESD; infine la terza presenta gli strumenti di produzione (o elaborazione) che possono sostituire o affiancare il filologo in una particolare fase del processo editoriale.

Nell'ultima categoria, gli strumenti sono ulteriormente suddivisi in base alla fase del processo editoriale per la quale possono essere impiegati. Come illustrato al capitolo 1, le fasi editoriali sono quelle che si prestano particolarmente a una automatizzazione, completa o parziale, tra cui la trascrizione e la collazione. A esse si aggiungono anche l'annotazione e l'analisi del testo, in cui è compresa anche la lemmatizzazione (Leonardi 2021). Per ogni categoria, macro e non, vengono brevemente presentate le attività corrispondenti della tassonomia TaDiRAH.¹

Il distinguo tra le categorie proposte non è sempre netto, tendenzialmente gli strumenti sono stati inquadrati in base alle loro principali funzionalità e al tipo di *output* o risultato che permettono di ottenere: uno strumento è categorizzato come di visualizzazione o misto quando consente di produrre un risultato che può essere definito un'edizione scientifica digitale.

Per ogni strumento si presentano le seguenti informazioni:

¹ <https://tadirah.info/>.

- “Sito web”, l’indirizzo del sito web ufficiale, del *repository* o del *wiki* che i creatori dello strumento offrono come “vetrina” per conoscere lo strumento;
- “Anno di realizzazione”, l’anno in cui la prima versione pubblica dello strumento è stata rilasciata. Nel caso in cui lo strumento è stato prodotto solo in via sperimentale senza raggiungere una prima versione ufficiale, è stato considerato l’anno di creazione del *repository* del rispettivo codice sorgente o l’anno indicato dal creatore come data di realizzazione nel sito web o in una pubblicazione;
- “Descrizione”, testo più o meno breve utilizzato nel sito web, nel *repository* o nel *wiki* dello strumento per presentarlo ai potenziali utenti. Si tratta di un testo di carattere più “commerciale” che tecnico, in quanto ha lo scopo di incoraggiare a utilizzare lo strumento, e presenta in breve le principali funzionalità e caratteristiche dello strumento, con riferimenti anche agli enti partner. Si riporta la descrizione poiché è interessante analizzare le parole con le quali il creatore presenta il proprio strumento;
- “Creatore”, il nome del creatore o più creatori dello strumento. Nel caso di uno strumento commerciale, si riporta il nome dell’azienda che lo ha prodotto. Analogamente, se lo strumento è stato prodotto da un centro di ricerca, da un collettivo o da un altro tipo di gruppo con una denominazione ufficiale, si riporta il nome del gruppo. Se, invece, lo strumento è stato realizzato da uno o più studiosi, si indicano i rispettivi nomi ed eventualmente l’affiliazione;
- “Ente/i partner”, gli enti che hanno contribuito finanziariamente alla realizzazione dello strumento. Si tratta di università, centri di ricerca, enti culturali, progetti e fondi pubblici. Dato che molti di questi fondi sono a tempo determinato, ho scelto di aggiungere questa informazione, in quanto permette di capire quanto uno strumento sia stabile o meno in termini di sviluppo, manutenzione e supporto agli utenti;
- “Formati di input”, i diversi formati che lo strumento accetta in input. In alcuni casi non è possibile reperire i formati esatti in quanto le fonti ufficiali indicano genericamente “testo” o “immagine”. In questi casi si riportano i formati più utilizzati e che molto probabilmente sono supportati: TXT per i testi e JPEG per le immagini;

- “Formati di output”, i formati di output in cui lo strumento permette di esportare la ESD o altri tipi di dati. Si sottolinea come alcuni strumenti, in particolare tra quelli misti o di sola visualizzazione, non prevedano l’esportazione dei dati;
- “Tecnologie utilizzate per lo sviluppo”, ovvero i linguaggi di programmazione, i *tool*, le librerie e i *framework* utilizzati per sviluppare lo strumento. Alcuni degli strumenti elencati sono stati integrati in altri strumenti (per esempio in EVT sono stati integrati OpenSeadragon e VisColl);
- “Utilizzo”, indica la natura informatica dello strumento, se è per esempio un software da installare su *desktop*, un servizio offerto attraverso il web, una libreria, un’applicazione web-based, ecc. e i requisiti di sistema (per esempio i sistemi operativi compatibili);
- “Consente il lavoro collaborativo”, campo che indica se lo strumento è stato concepito per lavorare collaborativamente, ovvero permettendo a più utenti di lavorare in contemporanea sugli stessi materiali;
- “Open source”, indica se il codice sorgente dello strumento è liberamente accessibile;
- “Repository del codice sorgente”, l’indirizzo del *repository* in cui è conservato il codice sorgente dello strumento;
- “Licenza”, la licenza attraverso la quale lo strumento è disponibile. Se lo strumento è a pagamento, viene indicato con la dicitura “A pagamento”. Se lo strumento è libero, ma non viene specificata la licenza *open source* esatta,² la licenza è marcata semplicemente come “Gratuita”;
- “Ultima versione rilasciata”, il numero dell’ultima versione pubblicata dello strumento;
- “Edizioni Critiche Digitali di esempio”, il titolo delle ESD che sono state realizzate con lo strumento. Questo campo è molto utile per i potenziali utenti-editori, in quanto permette di vedere concretamente come una ESD si presenta grazie allo strumento e/o quali sono i risultati scientifici che lo strumento consente di ottenere. I nomi sono dei collegamenti ipertestuali ai rispettivi siti ufficiali delle ESD;

² Per maggiori informazioni sulle diverse tipologie di licenze *open source* si rimanda alle linee guida proposte dall’AGID: <https://docs.italia.it/italia/developers-italia/lg-acquisizione-e-riuso-software-per-pa-docs/it/stabile/attachments/allegato-c-guida-alle-licenze-open-source.html>.

- “Pubblicazioni”, i riferimenti bibliografici delle pubblicazioni in cui i creatori presentano il proprio strumento o altri studiosi recensiscono o riportano la propria esperienza di utilizzo del software con un caso d’uso concreto.

Per molti strumenti non è stato possibile fornire tutte le informazioni sopra elencate, in quanto non reperibili nelle fonti ufficiali (*in primis* i siti web e i *repository*) o perché non è stato possibile identificare fonti bibliografiche aggiornate e affidabili. Il fatto che l’informazione sia stata omessa, non significa che non possa essere reperita conducendo uno studio ancora più approfondito o rivolgendosi ai creatori stessi. Inoltre, le informazioni potrebbero non essere più aggiornate o valide, ad esempio i collegamenti ipertestuali ai siti web potrebbero non essere più attivi.

L’elenco proposto in questa appendice non è completo, in quanto contempla strumenti prodotti perlopiù in Italia, in Europa e negli Stati Uniti. L’obiettivo principale del lavoro svolto è quello di offrire una prima panoramica degli strumenti per la filologia digitale, per quanto parziale e incompleta, e di proporre delle categorie utili per analizzare e classificare questi strumenti dalla prospettiva dell’utente-editore che desidera realizzare una ESD. Questa rassegna, infatti, costituisce il punto di partenza per un progetto di ricerca più ampio che vorrei intraprendere in futuro, ovvero realizzare un catalogo degli strumenti informatici per la filologia digitale (Martignano 2023).

Ringrazio i professori Roberto Rosselli Del Turco e Lino Leonardi per avermi fornito i riferimenti a diversi degli strumenti analizzati.

L’elenco è disponibile in *open access* in formato XLSX mediante la piattaforma [Zenodo](#) ed è consultabile anche in una versione più *user-friendly* [online](#).

Risorse utilizzate

La principale difficoltà riscontrata in fase di analisi dello stato dell’arte riguarda il recupero delle informazioni. Non sono disponibili dei cataloghi dedicati e completi. Il catalogo TAPoR riporta solo alcuni dei principali strumenti, ma non fornisce una panoramica esaustiva di tutti gli strumenti utilizzati nel campo della filologia digitale. Nel catalogo delle edizioni critiche digitali curato da Patrick Sahle, in una pagina a parte vengono forniti alcuni riferimenti sommari sui principali strumenti e

sulle principali infrastrutture.³ All'interno delle descrizioni dei singoli record vengono fornite alcune indicazioni sull'infrastruttura e sugli strumenti utilizzati in forma discorsiva, ma non è possibile effettuare una ricerca full text sulle descrizioni. Nel *Catalogue of Digital Editions* di Franzini è disponibile un campo "infrastructure", nel quale sono elencati i linguaggi di programmazione, le tecnologie e gli strumenti utilizzati per lo sviluppo delle edizioni digitali. Tuttavia, all'interno di questi elenchi non è facile riconoscere gli strumenti per la pratica ecdotica, in quanto il campo offre solo il nome, senza collegamenti a risorse esterne o pubblicazioni che permetterebbero di individuarli con certezza. Inoltre, molti degli strumenti citati sono poco noti, perché poco utilizzati dalla comunità scientifica o perché non sono più disponibili, oppure sono stati registrati con il nome di una loro versione precedente.

Alcune fonti preziose per sopperire alla mancanza di un catalogo sono:

- le rassegne degli strumenti condotte all'interno di pubblicazioni scientifiche, per esempio quella realizzata da Cinzia Pusceddu in appendice al volume *Scrittura e filologia nell'era digitale* di Domenico Fiormonte (2003);
- le recensioni della rivista RIDE;⁴
- gli atti dei principali convegni di *digital humanities*, in occasione dei quali vengono spesso presentati nuovi strumenti o forniti resoconti sullo stato di sviluppo di quelli esistenti.

Un'altra grande difficoltà è data dall'instabilità di molti strumenti, che cambiano nome, che cambiano URL o che semplicemente diventano irreperibili, perché non vengono più mantenuti.

Strumenti di produzione

Per descrivere secondo la tassonomia TaDiRAH le funzionalità di buona parte degli strumenti di produzione elencati ci si può servire dell'attività "editing"⁵ oppure della più generica attività "enriching": "the activity of annotating, editing, or cleaning information to an object of enquiry, by making its origin, nature, structure, meaning, or elements explicit".⁶

³ <https://www.digitale-edition.de/exist/apps/editions-browser/edition.html>.

⁴ <https://ride.i-d-e.de/>.

⁵ <https://vocabs.dariah.eu/tadirah/en/page/editing>.

⁶ <https://vocabs.dariah.eu/tadirah/en/page/enriching>.

Trascrizione

Nella tassonomia TaDiRAH è presente l'attività "transcribing" così definita: "the activity of creating a representation of a manuscript or of audio or video recordings. The representation is generally textual for the verbal aspects of recordings and structured for example by speech turns. It can also contain multimodal information like gestures or events and multimedia information like time synchronization and relation to media files".⁷

ABBYY Fine Reader

Software commerciale per la scansione OCR, basata sull'intelligenza artificiale. Si presta a essere utilizzato per la trascrizione automatica di fonti primarie, sia a stampa sia manoscritte. Il software è in grado di distinguere tra caratteri in tondo e in corsivo, funzionalità che facilita, per esempio, il riconoscimento delle abbreviazioni.

<i>Sito web</i>	https://pdf.abbyy.com/
<i>Descrizione</i>	FineReader PDF empowers professionals to maximize efficiency in the digital workplace. Featuring ABBYY's latest AI-based OCR technology, FineReader PDF makes it easier to digitize, retrieve, edit, protect, share, and collaborate on all kinds of documents in the same workflow. Now, information workers can focus even more on their expertise and less on administrative tasks.
<i>Creatore/i</i>	ABBYY
<i>Formati di input</i>	PDF
<i>Formati di output</i>	DOC, XLS, RTF, EPUB, HTML, PDF
<i>Utilizzo</i>	Desktop, standalone
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	No
<i>Licenza</i>	A pagamento
<i>Ultima versione rilasciata</i>	15

⁷ <https://vocabs.dariah.eu/tadirah/en/page/transcribing>.

eScriptorium

eScriptorium è un applicativo web per la trascrizione, *open source*, gratuito e nato in ambiente accademico, che combina i due approcci illustrati al capitolo 1 (vd. 1.3.3.1): da un lato utilizza algoritmi di *machine learning* per effettuare automaticamente la trascrizione, dall'altro offre una piattaforma per lavorare manualmente alla trascrizione in modo collaborativo. Creato nell'ambito dei progetti Scripta, RESILIENCE and Biblissima+, eScriptorium è uno strumento ancora in fase di sviluppo ma che offre già diverse funzionalità, tra cui: la trascrizione automatica mediante Kraken; la ricerca e l'applicazione di filtri; l'annotazione; infine, alcune forme basilari di lavoro collaborativo, quali la condivisione e il controllo delle versioni dei documenti. L'obiettivo finale dei progetti che hanno creato eScriptorium è quello di offrire in un unico applicativo web una collezione di strumenti integrati per trascrivere, annotare, tradurre e pubblicare documenti storici.

<i>Sito web</i>	https://escriptorium.openiti.org/
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	<p>A Digital Text Production Pipeline for Print and Handwritten Texts using machine learning techniques.</p> <p>Its goal is provide researchers in the humanities with an integrated set of tools to transcribe, annotate, translate and publish historical documents. The eScriptorium app itself is at the 'center'. It is a work in progress but will implement at least automatic transcriptions through kraken, indexation for complex search and filtering, annotation and some simple forms of collaborative working such as sharing and versioning.</p>
<i>Creatore/i</i>	Daniel Stökl Ben Ezra (Co-Director), Peter Stokes (Co-Director), Benjamin Kiessling (machine vision engineer, responsible for Kraken), Robin Tissot (lead project engineer for eScriptorium)
<i>Ente/i partner</i>	<p>École Pratique des Hautes Études – Université Paris Sciences et Lettres, in AOROC in Paris</p> <p>Scripta PSL programme (PSL University, Ecole normale supérieure, Ecole pratique des hautes études, Ecole nationale de chartes, Ecole des hautes études en sciences sociales, Ecole française de l'extrême orient, Collège de France, Institut de recherche d'histoire de textes)</p> <p>A team from Teklia, and from the Open Islamicate Texts Initiative at the University of Maryland and Northeastern University</p>
<i>Formati di input</i>	TIFF, PNG, JPG, ALTO, IIF, PDF, XML, PageXML, zip
<i>Formati di output</i>	PageXML and TEI-XML (Text Encoding Initiative) import and export are in preparation
<i>Tecnologie utilizzate per lo sviluppo</i>	nginx, uwsgi, django, daphne (channel server for websockets), celery, postgres, elasticsearch (integration not started yet), redis (cache, celery broker, other disposable data), kraken, docker (deployment)

<i>Utilizzo</i>	Applicativo web
<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
<i>Repository del codice sorgente</i>	https://gitlab.com/scripta/escrptorium/
<i>Ultima versione rilasciata</i>	0.12.3
<i>Pubblicazioni</i>	<p>Stokes, P., B. Kiessling, D. Stökl Ben Ezra, R. Tissot, and E. H. Gargem. "The EScriptorium VRE for Manuscript Cultures." Edited by Claire Clivaz and Garrick V. Allen. <i>Classics@ Journal, Ancient Manuscripts and Virtual Research Environments</i>, 18 (2021). https://classics-at.chs.harvard.edu/classics18-stokes-kiessling-stokl-ben-ezra-tissot-gargem/.</p> <p>B. Kiessling, R. Tissot, P. Stokes and D. Stökl Ben Ezra, "eScriptorium: An Open Source Platform for Historical Document Analysis", 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), Sydney, NSW, Australia, 2019, pp. 19-19, doi: 10.1109/ICDARW.2019.10032.</p>

FairCopy

Software proprietario e a pagamento ideato appositamente per elaborare testi in XML/TEI, partendo dalla trascrizione delle immagini. Supporta IIIF e permette di modificare la struttura del documento TEI attraverso una funzionalità di *drag and drop*. Faircopy è stato realizzato dall'azienda americana Performant, che negli anni ha lavorato in diversi progetti di *digital humanities* a stretto contatto con studiosi di università americane ed europee, producendo diversi prodotti, tra cui Juxta.

<i>Sito web</i>	https://www.faircopyeditor.com/
<i>Anno di realizzazione</i>	2022
<i>Descrizione</i>	FairCopy is a simple and powerful tool for reading, transcribing, and encoding text.
<i>Creatore/i</i>	Performant Software Solutions LLC Lead designer and developer: Nick Laiacona
<i>Formati di input</i>	XML-TEI, TXT (Unicode), IIIF Manifest
<i>Formati di output</i>	XML-TEI
<i>Tecnologie utilizzate per lo sviluppo</i>	IIIF, OpenSeadragon, ProseMirror, Font Awesome, Electron, keygen, FlexSearch
<i>Utilizzo</i>	Desktop for Windows, macOS and Linux
<i>Consente il lavoro collaborativo</i>	Sì

Open source	No
<i>Licenza</i>	A pagamento
<i>Ultima versione rilasciata</i>	1.0

FromThePage

FromThePage si distingue dagli altri software presentati in questa sezione, in quanto si basa esclusivamente su una trascrizione manuale effettuata da volontari. È uno strumento pensato per biblioteche, archivi ed enti culturali che intraprendono grandi campagne di digitalizzazione di documenti storici. Tuttavia, può essere usato in modo collaborativo anche in progetti più piccoli. Lo strumento è disponibile sia come applicativo *online* a pagamento oppure come applicativo web *open source* e gratuito. Nel primo caso la piattaforma mette a disposizione più di 5000 volontari registrati alla piattaforma, nel secondo, invece, occorre individuare in autonomia un gruppo di persone che effettuino la trascrizione. Inoltre, l'applicativo web *open source* deve essere installato su un *server* e reso accessibile da remoto.

<i>Sito web</i>	https://fromthepage.com/
<i>Anno di realizzazione</i>	2005
<i>Descrizione</i>	FromThePage is a crowdsourcing platform for archives and libraries where volunteers transcribe, index, and describe historic documents
<i>Creatore/i</i>	Sara Brumfield, Ben Brumfield
<i>Formati di input</i>	CONTENTdm, IIIF Manifest, PDF, zip, JPEG
<i>Formati di output</i>	doc, PDF, TXT, CONTENTdm
<i>Tecnologie utilizzate per lo sviluppo</i>	IIF, Ruby, JavaScript, HTML, CSS, SCSS, Slim
<i>Utilizzo</i>	Online Requisiti per la versione <i>open source</i> dell'applicativo: Ruby on Rails versione 6.0.3.2, RMagick, hpricot, will_paginate, OAI gems.
<i>Consente il lavoro collaborativo</i>	Sì
Open source	Sì
Repository del codice sorgente	https://github.com/benwbrum/fromthepage
<i>Licenza</i>	Le prime 200 pagine gratis, le successive a pagamento. Versione <i>open source</i> : AGPL-3.0
<i>Ultima versione rilasciata</i>	23.5

Kraken

Kraken è un software *open source* per il riconoscimento automatico dei caratteri basato sul linguaggio di programmazione python e realizzato nel medesimo contesto istituzionale di eScriptorium.

<i>Sito web</i>	https://kraken.re/main/index.html
<i>Anno di realizzazione</i>	2015
<i>Descrizione</i>	<p>Kraken is a turn-key OCR system optimized for historical and non-Latin script material.</p> <p>kraken's main features are:</p> <ul style="list-style-type: none">Fully trainable layout analysis and character recognitionRight-to-Left, BiDi, and Top-to-Bottom script supportALTO, PageXML, abbyXML, and hOCR outputWord bounding boxes and character cutsMulti-script recognition supportPublic repository of model filesVariable recognition network architecture
<i>Creatore/i</i>	Benjamin Kiessling
<i>Ente/i partner</i>	École Pratique des Hautes Études, Université PSL, RESILIENCE project, Biblissima+
<i>Formati di input</i>	TIFF
<i>Formati di output</i>	ALTO, PageXML, abbyXML, hOCR
<i>Tecnologie utilizzate per lo sviluppo</i>	Python
<i>Utilizzo</i>	Desktop only on Linux or macOS
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://github.com/mittagessen/kraken
<i>Licenza</i>	Apache 2.0
<i>Ultima versione rilasciata</i>	4.3.10

OCROpus

OCROpus è una raccolta di motori OCR basati su reti neurali originariamente sviluppati da Thomas Breuel, con il contributo di studenti, aziende e ricercatori. I diversi motori sono disponibili in *open source* in altrettanti *repository* GitHub.

<i>Sito web</i>	https://ocropus.github.io/
<i>Anno di realizzazione</i>	2007
<i>Descrizione</i>	OCROPUS is a collection of neural-network based OCR engines originally developed by Thomas Breuel, with many contributions from students, companies, and researchers. The github.com/ocropus organization collects many of the repositories.
<i>Creatore/i</i>	Thomas Breuel
<i>Ente/i partner</i>	German Research Centre for Artificial Intelligence in Kaiserslautern, Germany
<i>Tecnologie utilizzate per lo sviluppo</i>	C++, Python
<i>Utilizzo</i>	For Linux and macOS
<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
Repository del codice sorgente	https://github.com/ocropus
<i>Licenza</i>	Apache 2.0

Online Transcription Editor (OTE)

OTE è stato sviluppato presso il Trier Center for Digital Humanities a partire dal 2011 ed è la versione web del software Transcribo. OTE è stato concepito come parte integrante di un ambiente di lavoro per l'*editing* collaborativo in due progetti di edizioni di testi storici.⁸ Secondo quanto riportato nella pagina web ufficiale dello strumento e le ultime attività rilevabili sulla relativa pagina SourceForge, lo strumento non viene più sviluppato dal 2021, anno in cui sono terminati i progetti mediante i quali la realizzazione di OTE era stata finanziata.

<i>Sito web</i>	https://tcdh.uni-trier.de/en/projekt/online-transcription-editor-ote
<i>Anno di realizzazione</i>	2011
<i>Descrizione</i>	A Web-Based Transcription Tool
<i>Creatore/i</i>	Dr Thomas Burch
<i>Ente/i partner</i>	Project Management: Universität Trier - Trier Center for Digital Humanities (TCDH) Project participants: Universität Münster – Institut für Neutestamentliche

⁸ Per maggiori informazioni si rimanda a <https://tcdh.uni-trier.de/en/projekt/workspace-collaborative-editing>.

Textforschung (INTF) · University of Birmingham – Institute for Textual Scholarship and Electronic Editing (ITSEE) · University of London – School of Oriental and African Studies (SOAS)
Sponsors: Deutsche Forschungsgemeinschaft (DFG), European Research Council (ERC), Advanced Grant

<i>Formati di output</i>	TEI-XML, HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	HTML, Javascript/Typescript, XML
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	No
Open source	No

Tesseract

Tesseract è come OCROpus un motore OCR *open source* basato sulle reti neurali. Fu sviluppato tra il 1984 e il 1994 presso l'azienda Hewlett-Packard. Nel 2005 venne reso *open source* e tra il 2006 e il 2018 fu sviluppato da Google. L'ultima versione stabile (la 5.0.0) è stata pubblicata a novembre del 2021. Tesseract può essere utilizzato direttamente tramite la riga di comando o mediante un'API per estrarre il testo stampato dalle immagini. Supporta un'ampia varietà di lingue. Tesseract non ha una GUI (*graphical user interface*) integrata, ma ce ne sono diverse disponibili come per esempio Rescribe.⁹

<i>Sito web</i>	https://tesseract-ocr.github.io/tessdoc/ .
<i>Anno di realizzazione</i>	1985
<i>Descrizione</i>	<p>Tesseract 4 adds a new neural net (LSTM) based OCR engine which is focused on line recognition, but also still supports the legacy Tesseract OCR engine of Tesseract 3 which works by recognizing character patterns. Compatibility with Tesseract 3 is enabled by using the Legacy OCR Engine mode (--oem 0). It also needs traineddata files which support the legacy engine, for example those from the tessdata repository.</p> <p>Tesseract has unicode (UTF-8) support, and can recognize more than 100 languages “out of the box”. This project does not include a GUI application.</p>
<i>Creatore/i</i>	Ray Smith (lead developer until 2018), Stefan Weil (current lead developer), Zdenko Podobny (maintainer)
<i>Ente/i partner</i>	Hewlett-Packard Laboratories (1985 -1994), Google (2006 - 2018)

⁹ <https://rescribe.xyz/rescribe/>.

<i>Formati di input</i>	PNG, JPEG, TIFF
<i>Formati di output</i>	plain text, hOCR (HTML), PDF, invisible-text-only PDF, TSV and ALTO (the last one - since version 4.1.0)
<i>Tecnologie utilizzate per lo sviluppo</i>	C++
<i>Utilizzo</i>	Desktop, C++ compiler required
<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
<i>Repository del codice sorgente</i>	https://github.com/tesseract-ocr/tesseract
<i>Licenza</i>	Apache 2.0
<i>Ultima versione rilasciata</i>	5.3.1
<i>Pubblicazioni</i>	Smith, Ray. "An overview of the Tesseract OCR engine." Ninth international conference on document analysis and recognition (ICDAR 2007). Vol. 2. IEEE, 2007.

TextLab

Prodotto dalla stessa azienda che ha realizzato FairCopy, può essere descritto come il predecessore di quest'ultimo, ma è gratuito e disponibile in *open source*. Inoltre, può essere utilizzato anche come ambiente di pubblicazione di edizioni digitali.

<i>Sito web</i>	http://www.textlab.org/
<i>Anno di realizzazione</i>	2017
<i>Descrizione</i>	TextLab enables users to transcribe revisions in manuscript of a work, collate variant print versions, narrate the sequence of revision, display diplomatic and base versions, and generate a corrected, emended, and annotated reading text version as a critical edition of the work.
<i>Creatore/i</i>	Performant Software Solutions LLC Lead designer and developer: Nick Laiacona
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	Ruby on Rails, JRuby, Backbone.js, Bootstrap, TEI XSLT
<i>Utilizzo</i>	Online

<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
Repository del codice sorgente	https://github.com/performant-software/textlab
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	0.3.12
<i>Edizioni Critiche Digitali di esempio</i>	Melville Electronic Library (a critical archive)
<i>Pubblicazioni</i>	Laiacona, Nick. “Editing Melville’s “Billy Budd” with TextLab, Juxta Editions, and MEL Catalog”, DH 2017.

T-PEN

Applicativo *web-based* gratuito e *open source*, che permette di esportare i dati finali in PDF o XML/TEI. T-PEN sembrerebbe essere giunto alla sua terza versione e nel sito viene segnalato che nel 2022 il software è tuttora mantenuto da un team dell’Università della Louisiana. Tuttavia, le ultime attività rilevabili nel *repository* GitHub risalgono a nove anni fa e pare che l’interfaccia grafica del software non venga aggiornata da diverso tempo.

<i>Sito web</i>	http://www.t-pen.org/TPEN/
<i>Anno di realizzazione</i>	2010
<i>Descrizione</i>	T-PEN is a web-based tool for working with images of manuscripts. Users attach transcription data (new or uploaded) to the actual lines of the original manuscript in a simple, flexible interface.
<i>Creatore/i</i>	Center for Digital Theology at Saint Louis University (SLU)
<i>Ente/i partner</i>	Andrew W. Mellon Foundation, National Endowment for the Humanities
<i>Formati di output</i>	XML-TEI, PDF, HTML
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	Sì
Open source	Sì
Repository del codice sorgente	https://github.com/jginther/T-PEN/tree/master/trunk

Transcribo

Transcribo è un applicativo desktop per la trascrizione di testi, sviluppato presso il Trier Center for Digital Humanities, a partire dal 2011, unitamente alla sua versione online, Online Transcription Editor. In Transcribo è stato integrato il motore OCR Tesseract per la lettura automatica dei testi contenuti nelle immagini caricate dall'utente. Una particolarità di Transcribo è che consente di lavorare collaborativamente, mettendo a confronto e correggendo automaticamente le trascrizioni prodotte da utenti diversi.

<i>Sito web</i>	http://transcribo.org/en/
<i>Anno di realizzazione</i>	2012
<i>Descrizione</i>	Trier Editing Tool for Transcribing Texts
<i>Creatore/i</i>	Thomas Burch Frank Queens
<i>Ente/i partner</i>	Project Management: Trier Center for Digital Humanities Project Participants: University of Cambridge, Bergische Universität Wuppertal, Cambridge University Library Sponsors: Nordrhein-Westfälische Akademie der Wissenschaften und der Künste
<i>Formati di output</i>	XML-TEI
<i>Tecnologie utilizzate per lo sviluppo</i>	Java, XML, Tesseract
<i>Utilizzo</i>	Desktop for Windows and macOS
<i>Consente il lavoro collaborativo</i>	Sì
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://bitbucket.org/tcdh/transcribodownload/src/master/

Transkribus

Transkribus è il software per la trascrizione automatica che negli ultimi anni ha riscosso maggiore successo nel campo delle *digital humanities*. Basato sull'intelligenza artificiale, questo strumento è una piattaforma *online* che consente di lavorare collaborativamente alla trascrizione automatica di testi. Registrandosi alla piattaforma è possibile caricare immagini e documenti in una propria collezione privata e procedere con la trascrizione, che di base prevede due fasi: una prima fase di segmentazione dei documenti in più linee o blocchi (con l'aiuto di un software per l'analisi dell'impaginazione); una seconda fase di collegamento tra i diversi segmenti e le corrispondenti porzioni di testo trascritto. La piattaforma offre anche un servizio personalizzato di HTR agli utenti/progetti che hanno trascritto almeno cento documenti, ovvero che dispongono di una base sufficiente per il training del motore di HTR. La piattaforma Transkribus è ospitata dal gruppo di Digitalizzazione e Preservazione Digitale dell'Università di Innsbruck ed è stata finanziata dalla Commissione Europea come

parte del progetto READ.¹⁰ Transkribus è ancora in fase di sviluppo. Buona parte del software alla base di Transkribus è disponibile *open source*.

<i>Sito web</i>	https://transkribus.eu/Transkribus/
<i>Anno di realizzazione</i>	2013
<i>Descrizione</i>	Transkribus is an AI platform that supports your work with historical documents. Transkribus enables you to automatically recognise text, layout, and structure in your documents with the power of AI. For this, you can train your own AI models that fit your specific documents. Transkribus also lets you enrich your material with metadata, you can collaborate with others, and searching your material is made easy too.
<i>Creatore/i</i>	READ-COOP SCE
<i>Formati di input</i>	File immagine (JPEG)
<i>Formati di output</i>	TEI, RTF, PDF, XML
<i>Tecnologie utilizzate per lo sviluppo</i>	Java
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	Sì
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://gitlab.com/readcoop/transkribus
<i>Licenza</i>	Le prime 500 pagine manoscritto o le prime 3000 pagine a stampa gratuite, le successive a pagamento
<i>Ultima versione rilasciata</i>	TranskribusLite 2.3.0.44
<i>Pubblicazioni</i>	Kahle, Philip, et al. "Transkribus-a service platform for transcription, recognition and retrieval of historical documents." 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). Vol. 4. IEEE, 2017. Muehlberger, Guenter, et al. "Transforming scholarship in the archives through handwritten text recognition: Transkribus as a case study." Journal of documentation (2019).

¹⁰ <https://readcoop.eu/>.

Transym

Transym è un software OCR a pagamento ideato per essere integrato facilmente in altri sistemi informatici. È possibile utilizzare gratuitamente l'API del software.

<i>Sito web</i>	https://transym.com/
<i>Anno di realizzazione</i>	2002
<i>Descrizione</i>	TOCR is one of the most sophisticated OCR software packages on the market, specifically designed for ease of integration. With a free API and attractive volume pricing, it is designed for system integrators.
<i>Creatore/i</i>	Transym computer Services Ltd
<i>Formati di input</i>	TIFF, BMP, PDF
<i>Consente il lavoro collaborativo</i>	Sì
<i>Open source</i>	No
<i>Licenza</i>	A pagamento, prova gratuita per 50 fogli A4
<i>Ultima versione rilasciata</i>	5.1

Collazione

Nella tassonomia TaDiRAH è presente l'attività "collating" così definita: "the activity of comparing two or more versions of written or spoken natural language data by aligning them into alphabetical or numerical order to identify similarities and differences".¹¹

CATView

CATView (the Colored & Aligned Texts view) è uno strumento di visualizzazione interattiva per esplorare e navigare tra le differenze testuali di un'opera o documento. Elabora i dati generati da strumenti di collazione automatica, quali LERA, CollateX e Juxta, ed è pensato per essere integrato in siti web come componente aggiuntivo di edizioni digitali o di strumenti di editing. CATView è stato realizzato dallo stesso team di sviluppatori e nello stesso contesto istituzionale di LERA ed è disponibile come software *open source*.

<i>Sito web</i>	http://catview.uzi.uni-halle.de/
<i>Anno di realizzazione</i>	2015

¹¹ <https://vocabs.dariah.eu/tadirah/en/page/collating>.

<i>Descrizione</i>	CATview is an interactive visualization tool offering effective functionalities for navigating and exploring the textual differences of the variants of a manuscript. It processes data generated by collation tools such as LERA, CollateX or Juxta. CATview is meant to be embedded in websites as a powerful add-on for digital editions and web-based frameworks for the editing process.
<i>Creatore/i</i>	Marcus Pöckelmann, Paul Molitor and Jörg Ritter
<i>Ente/i partner</i>	Computer Science Institute of Martin Luther University Halle-Wittenberg German Federal Ministry of Education and Research
<i>Formati di input</i>	HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript, D3.js, Font Awesome
<i>Utilizzo</i>	Web-based Embeddable in an HTML file
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://gitlab.informatik.uni-halle.de/poeckelm/catview
<i>Licenza</i>	MIT
<i>Ultima versione rilasciata</i>	2.5.2
<i>Edizioni Critiche Digitali di esempio</i>	
<i>Pubblicazioni</i>	

CollateX

Tra gli attuali software per la collazione automatica il più celebre è CollateX*, nato come una completa riscrittura di Collate, il software sviluppato da Peter Robinson. CollateX consente di leggere e confrontare due o più versioni di un testo. Ciascuna versione viene divisa in più parti, o token, che possono essere comparati e allineati per individuare similarità e differenze tra le versioni. Il risultato di questo allineamento può essere utilizzato per realizzare un apparato critico o per l'analisi stemmatica della genesi di un testo. Inoltre, l'applicativo permette di visualizzare graficamente gli allineamenti, generando un output in formato GraphML e GraphViz DOT.¹²

Sito web <https://collatex.net/>

¹² Per approfondire la questione della validità scientifica ed efficacia della rappresentazione attraverso un grafo del risultato di una collazione si rimanda a Schmidt e Colomb 2009.

<i>Anno di realizzazione</i>	2010
<i>Descrizione</i>	CollateX is a software to: read multiple (≥ 2) versions of a text, splitting each version into parts (tokens) to be compared; identify similarities of and differences between the versions (including moved/transposed segments) by aligning tokens; output the alignment results in a variety of formats for further processing, for instance; support the production of a critical apparatus or the stemmatical analysis of a text's genesis.
<i>Creatore/i</i>	Interedition Development Group Project Leads: Ronald Haentjens Dekker (Huygens ING) Gregor Middell
<i>Formati di input</i>	Plain Text (.txt), JSON, XML
<i>Formati di output</i>	JSON, TEI P5, XML, GraphML, GraphViz DOT
<i>Tecnologie utilizzate per lo sviluppo</i>	Tool suite comprised of a command-line tool and a self-contained RESTful web service or Apache-Maven-compatible artifacts of CollateX' core.
<i>Utilizzo</i>	Desktop
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/interedition/collatex/
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	1.7.1
<i>Edizioni Critiche Digitali di esempio</i>	Samuel Beckett Digital Manuscript Project
<i>Pubblicazioni</i>	Dekker, R. Haentjens, and Gregor Middell. "Computer-supported collation with CollateX: managing textual variance in an environment with varying requirements." Supporting Digital Humanities 2011: Answering the unaskable. 2011.

HyperCollate

HyperCollate è stato concepito per confrontare diverse versioni di un testo non sotto forma di stringhe testuali, ma come grafi. Il software accetta in input solo documenti XML/TEI, che prima provvede a trasformare in altrettanti grafi e che poi collaziona, generando un nuovo grafo. Per il momento HyperCollate è disponibile solo come prototipo per la collazione dei diversi livelli di interpretazione che possono essere espressi mediante la marcatura di fenomeni quali aggiunte, sostituzioni, cancellature e variazioni testuali.

<i>Sito web</i>	https://github.com/HuygensING/hyper-collate
<i>Anno di realizzazione</i>	2020
<i>Descrizione</i>	HyperCollate is a prototype collation engine that is able to handle in-text variation (i.e. textual variation within one witness) if it's marked with the TEI/XML elements , <add>, <subst>, <app> or <rdg>. HyperCollate considers these elements as an interruption in the linear flow of the text: when its parser encounters a , <add>, <subst>, <app> or <rdg> element, the stream of text tokens splits up into two or more branches. During the alignment process, HyperCollate looks at each branch and selects the one with the most matches. This means you do not have to linearize or “flatten” your TEI/XML transcription: the multiple layers within an individual witness can be preserved.
<i>Creatore/i</i>	Elli Bleker, Bram Buitendijk and Ronald Haentjens Dekker
<i>Ente/i partner</i>	Huygens Institute for the History of the Netherlands
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	XML-TEI
<i>Tecnologie utilizzate per lo sviluppo</i>	Java, HTML, Kotlin, Jupyter Notebook
<i>Utilizzo</i>	Desktop Java Developer Kit required
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/HuygensING/hyper-collate
<i>Licenza</i>	Apache 2.0
<i>Ultima versione rilasciata</i>	1.3.4
<i>Pubblicazioni</i>	Bleeker, Elli, Bram Buitendijk, and Ronald Haentjens Dekker. 2019. “From Graveyard to Graph: Visualisation of Textual Collation in a Digital Paradigm.” <i>International Journal of Digital Humanities</i> 1 (June). https://doi.org/10.1007/s42803-019-00012-w .

Juxta

Questo software non è più disponibile, ma è stato catalogato in quanto ha contribuito allo sviluppo e all'affermazione della pratica della collazione automatica. Sviluppato nel 2009 dall'Università della Virginia, Juxta richiede in input documenti in formato testo o XML e consente di creare diverse visualizzazioni grafiche delle

varianti. Nella versione desktop del software è possibile generare in output una tabella lemmatizzata in formato HTML delle varianti presenti in ogni confronto. Juxta è stato sviluppato anche come applicativo web dall'azienda Performant Software, con il nome di Juxta Web Service, nel contesto dell'infrastruttura di ricerca NINES (vd. 1.3.6). L'applicativo web era disponibile gratuitamente attraverso il sito Juxta Commons, all'interno del quale era possibile caricare i propri file sorgente, selezionare i testimoni da collazionare (da un minimo di due a un massimo di venti testimoni) e di procedere alla collazione automatica. Il risultato era una visualizzazione in HTML dei diversi testimoni da soli o in parallelo con le varianti allineate. La versione online di Juxta fungeva non solo da ambiente di *visual analytics* e di collazione online, ma anche da piattaforma per la pubblicazione di edizioni digitali, in quanto gli utenti potevano condividere le proprie visualizzazioni e integrarle in altri siti web.

<i>Sito web</i>	https://www.juxtasoftware.org/
<i>Anno di realizzazione</i>	2009
<i>Creatore/i</i>	Performant Software Solutions LLC Lead designer and developer: Nick Laiacona
<i>Ente/i partner</i>	University of Virginia
<i>Formati di output</i>	HTML
<i>Consente il lavoro collaborativo</i>	No
Open source	No

LERA

Applicativo web ideato come base per la realizzazione di edizioni critiche di documenti storici, che permette di indagare le differenze e le somiglianze tra le molteplici versioni di un testo. Su richiesta LERA viene messo a disposizione *online* dall'Università Martin Luther di Halle-Wittenberg che lo ha realizzato. Il software offre diverse modalità di visualizzazione, tra cui *word cloud* interattivi, una visualizzazione gestita con il software CATView, apparati critici ed evidenziazioni delle porzioni interessate nei testi. Le *word cloud* sono offerte a livello di singola unità di variazione testuale e presentano al loro interno in corpo decrescente le attestazioni maggioritarie e quelle via via meno attestate.

<i>Sito web</i>	https://lera.uzi.uni-halle.de/
<i>Anno di realizzazione</i>	2012
<i>Descrizione</i>	LERA - Locate, Explore, Retrace and Apprehend complex text variants

LERA is an interactive, digital tool for inspecting similarities and differences between multiple versions of a text. To achieve this, LERA divides the texts into passages and displays them next to each other. Using colored highlighting, differences between versions can be located easily in a critical apparatus.

LERA features a variety of tools for efficiently exploring differences between versions of texts: an interactive outline visualization (CATview), interactive word clouds and clever search functionality. This way, LERA supports the user retrace and finally apprehend the authors intentions. LERA has been developed for collating texts to provide a basis for historical critical editions.

<i>Creatore/i</i>	Marcus Pöckelmann, Paul Molitor, Jörg Ritter
<i>Ente/i partner</i>	Martin-Luther-Universität Halle-Wittenberg
<i>Formati di input</i>	XML-TEI, XML, TXT (Unicode)
<i>Formati di output</i>	PDF, TEX, HTML, XML-TEI, JSON
<i>Utilizzo</i>	Web-based, service
<i>Consente il lavoro collaborativo</i>	No
Open source	No
<i>Edizioni Critiche Digitali di esempio</i>	Keter Shem Tov
<i>Pubblicazioni</i>	Roeder, Torsten. n.d. 'Juxta Web Service, LERA, and Variance Viewer. Web Based Collation Tools for TEI – RIDE'. Accessed 24 May 2023. https://ride.i-d-e.de/issues/issue-11/web-based-collation-tools/ . M. Pöckelmann, A. Medek, J. Ritter, and P. Molitor. LERA - An interactive platform for synoptical representations of multiple text witnesses In: Digital Scholarship in the Humanities (DSH). Oxford University Press 2022. DOI: 10.1093/llc/fqac021

TRAViz

TRAViz¹³ è una libreria JavaScript che genera grafi di varianti testuali aventi la particolarità di essere interattivi. Passando con il cursore sopra una parola, le ramificazioni che non la riportano vengono temporaneamente nascoste. L'utente può modificare la struttura del grafo, separando o unendo vertici di ramificazione diverse, mediante una funzionalità di *drag and drop*. Questa particolare funzionalità è stata pensata per permettere di correggere o rimuovere possibili errori prodotti dall'algoritmo di allineamento. Infine, cliccando su una porzione di testo vengono visualizzate le versioni che la attestano, con la possibilità di invertire la posizione della porzione di testo, per analizzare l'effetto di possibili trasposizioni.

Sito web <http://www.traviz.vizcovery.org/>

¹³ TRAViz (Text Re-Use Alignment Visualization) è stato ideato nel 2014 da Stefan Jänicke presso l'Università di Lipsia. Lo strumento è disponibile come software *open source*: <https://github.com/stjaenicke/TRAViz>.

<i>Anno di realizzazione</i>	2014
<i>Descrizione</i>	TRAViz is a JavaScript library that was developed within the Digital Humanities project eTRACES. It generates visualizations for Text Variant Graphs that show the variations between different editions of texts. TRAViz supports the collation task by providing methods to: align various editions of a text; visualize the alignment; improve the readability for Text Variant Graphs compared to other approaches; interact with the graph to discover how individual editions disseminate.
<i>Creatore/i</i>	Stefan Jänicke
<i>Formati di input</i>	JSON
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript, jQuery, Raphaël and qTip.
<i>Utilizzo</i>	Desktop
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
<i>Repository del codice sorgente</i>	https://github.com/stjaenicke/TRAViz
<i>Licenza</i>	Fair Academic License (FAL)
<i>Pubblicazioni</i>	S. Jänicke, A. Geßner, M. Büchler and G. Scheuermann (2014). Visualizations for Text Re-use. In Proceedings of the 5th International Conference on Information Visualization Theory and Applications, IVAPP 2014, pages 59–70.

Variance-Viewer

Variance-Viewer è uno strumento *open source* per il confronto testuale automatico pubblicato nel 2019 e sviluppato presso l'Università di Würzburg. Il *tool* permette di espandere l'insieme di regole su cui è basato il confronto, che di base viene condotto a livello di parola e presentato all'utente a livello di singolo carattere.

<i>Sito web</i>	https://github.com/cs6-uniwue/Variance-Viewer
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	Variance-Viewer is an automatic open-source tool for text comparison with an extendable rule based variance analysis. Documents are compared on word basis, while their variance is displayed on character basis.
<i>Creatore/i</i>	Nico Balbach
<i>Ente/i partner</i>	Department for Artificial Intelligence and Applied Computer Science of the University of Würzburg (Germany)

<i>Formati di input</i>	Plain Text (.txt), XML-TEI
<i>Formati di output</i>	PDF, XML-TEI, JSON
<i>Tecnologie utilizzate per lo sviluppo</i>	Java, JavaScript, CSS
<i>Utilizzo</i>	Web-based (The online version, http://variance-viewer.informatik.uni-wuerzburg.de/Variance-Viewer/ , is no longer accessible)
<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
Repository del codice sorgente	https://github.com/cs6-uniwue/Variance-Viewer
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	1.0.0
<i>Pubblicazioni</i>	Roeder, Torsten. n.d. 'Juxta Web Service, LERA, and Variance Viewer. Web Based Collation Tools for TEI – RIDE'. Accessed 24 May 2023. https://ride.i-d-e.de/issues/issue-11/web-based-collation-tools/ .

Annotazione

Nella tassonomia TaDiRAH è presente l'attività "annotating" così definita: "the activity of comparing two or more versions of written or spoken natural language data by aligning them into alphabetical or numerical order to identify similarities and differences".¹⁴

L'attività "annotating" comprende al suo interno le seguenti attività: "associating", "audio annotation", "contextualizing", "encoding", "georeferencing", "highlighting", "linked open data", "named entity recognition", "POS-tagging", "segmenting", "tagging", "visual annotation".

Alcune di queste attività si prestano a rappresentare anche tipologie di strumenti che nella presente rassegna sono stati catalogati come strumenti per l'analisi del testo oppure come strumenti di visualizzazione.

¹⁴ <https://vocabs.dariah.eu/tadirah/en/page/annotating>.

Annotation Studio

Annotation Studio*, un'applicazione web *open source* che consente di leggere e annotare diversi tipi di materiali (immagini, video, documenti, ecc.) in un ambiente collaborativo, ideata presso il MIT prevalentemente per essere applicata in un contesto scolastico/educativo, ma che può essere utilizzata anche per la creazione di archivi digitali.

<i>Sito web</i>	https://www.annotationstudio.org/
<i>Anno di realizzazione</i>	2012
<i>Descrizione</i>	<p>Annotation Studio is a suite of collaborative web-based annotation tools currently under development at MIT.</p> <p>Read Engage and reflect more critically upon texts instead of passively reading them.</p> <p>Annotate Bring the millennia old humanistic tradition of annotation into contemporary electronic media in the classroom.</p> <p>Share Create discussion groups for your classes and control the size and publicness of those groups.</p> <p>Compose Support the entire writing process, from initial reading to brainstorming, writing, and review.</p>
<i>Creatore/i</i>	MIT HyperStudio
<i>Ente/i partner</i>	National Endowment for the Humanities (Digital Humanities Start-Up grant in 2012 and Digital Humanities Implementation grant in 2013), MIT's SHASS Teaching and Learning Fund, MIT's Alumni Fund
<i>Tecnologie utilizzate per lo sviluppo</i>	Client: HTML, CSS, JavaScript. Server: Ruby on Rails, PostgreSQL, MongoDB, Node.js, Backbone.js, Bootstrap, jQuery. Github and Heroku.
<i>Utilizzo</i>	Web application
<i>Consente il lavoro collaborativo</i>	Si
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/hyperstudio/Annotation-Studio
<i>Licenza</i>	GPL-2.0

Arethusa

Arethusa è un *framework* per l'annotazione morfosintattica *open source* sviluppato dal progetto Alpheios¹⁵ in collaborazione con Perseus, la biblioteca digitale di testi in latino e greco antico. Le ultime attività rilevabili nel *repository* GitHub risalgono a due anni fa, mentre l'ultima versione del software è stata pubblicata nel 2017, per cui è probabile che il software non sia più sviluppato.

<i>Sito web</i>	https://www.perseids.org/tools/arethusa/app/#/
<i>Anno di realizzazione</i>	2015
<i>Descrizione</i>	<p>A backend-independent client-side annotation framework designed to:</p> <ul style="list-style-type: none">● Make annotating accessible and enjoyable, lowering the entry barrier for the user by providing access to related resources and services at the point of annotation.● Make the developer's job easier, by giving her tools and a framework for adding new functionality in a modular fashion and for binding diverse resources together in new ways.● Be fully open and have the ability to participate in external workflows that route digital content, together with its provenance, to and from diverse destinations.
<i>Creatore/i</i>	Alpheios Project
<i>Ente/i partner</i>	Perseids, Perseus Digital Library, Digital Humanities (University of Leipzig) This project has received support from the Andrew W. Mellon Foundation, the Institute of Museum and Library Services and the European Social Fund.
<i>Tecnologie utilizzate per lo sviluppo</i>	AngularJS
<i>Utilizzo</i>	Client-only framework
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://github.com/alpheios-project/arethusa
<i>Ultima versione rilasciata</i>	0.3.14

¹⁵ <https://alpheios.net/>.

Digital Mappa

Software *web based e open source* realizzato dal Center for the History of Print and Digital Culture dell'Università di Wisconsin-Madison. Digital Mappa consente di caricare i propri testi e file immagine, per poi evidenziare al loro interno delle porzioni o delle aree, collegandole tra di loro e annotandole.

<i>Sito web</i>	https://www.digitalmappa.org/
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	An open-source DH platform for open-access workspaces, projects, collaborations, and scholarly publications. The premise of DM is simple: if you have a collection of digital images and texts, then you should be able to develop a project where you can identify specific moments on these images and texts, annotate them as much as you want, link them together, generate searchable content, collaborate with your friends, and publish your work online for others to see and share. DM was developed on the Heroku cloud-server platform, where installation and administration is straightforward, after which developing individual projects requires no specific IT expertise.
<i>Creatore/i</i>	v. 1: Martin Foys v. 2 (refactoring): Performant Software
<i>Ente/i partner</i>	Center for the History of Print and Digital Culture (University of Wisconsin-Madison) and the Schoenberg Institute for Manuscript Studies (University of Pennsylvania)
<i>Formati di input</i>	JPG, PNG, IIIF Manifest, TXT (Unicode)
<i>Tecnologie utilizzate per lo sviluppo</i>	DM v2.1 is a single page React application backed by a Ruby on Rails server running a Postgres database. It uses ActiveStorage for image uploads and ImageMagick for image processing. It includes out the box support for storing images on Amazon S3. It can utilize a service like Postmark or SendGrid for outbound SMTP.
<i>Utilizzo</i>	(DM v2.1) was developed to be deployed on the Heroku cloud-server platform, using Amazon S3 Bucket to store image data. Additionally, the platform can now be set up on a local server relatively quickly using Docker Compose, where the only requirements are Docker and Docker Compose themselves. PostgreSQL v11.1, Ruby 2.5.7, and Bundler 2.2.23
<i>Consente il lavoro collaborativo</i>	Si
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/performant-software/dm-2
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	2.1

Edizioni Critiche Digitali di esempio [Amalarius's Bells: An Old English and Medieval Latin Edition](#), [Ezra Pound's "The Music of Beowulf"](#), [OEPF 2.0: Old English Poetry in Facsimile, Restorative Editions](#)

Pubblicazioni Hodel, Tobias, Anna Janka, and Jonas Widmer. 2022. 'Digital Mappa – Simple and Web-Based Annotations'. RIDE (blog). December 2022. <https://doi.org/10.18716/ride.a.15.1>.

Euporia

Sviluppato dal CoPhiLab del CNR di Pisa, Euporia è un *applet*¹⁶ di exist-db¹⁷ per l'annotazione testuale basato sui *Domain Specific Languages* (DSLs),¹⁸ in modo tale da permettere all'utente di stabilire la grammatica e il vocabolario del linguaggio di annotazione.

Sito web <https://github.com/CoPhi/euporia>

Anno di realizzazione 2020

Descrizione Euporia is an annotation tool based on Domain Specific Languages (DSLs). You can create the Context-Free Grammar for your DSL and generate the parser for your language.

Creatore/i CoPhiLab (Collaborative and Cooperative Philology Lab)

Formati di input XML

Formati di output XML

Tecnologie utilizzate per lo sviluppo JavaScript, eXist-db, Docker

Utilizzo Euporia is an applet for eXist-db 5.3.0. Install the docker for eXist-dba

Consente il lavoro collaborativo No

Open source Sì

Repository del codice sorgente <https://github.com/CoPhi/euporia>

Licenza GPL-3.0

Ultima versione rilasciata alpha

¹⁶ Con *applet* si intende un programma che viene eseguito all'interno di un altro programma.

¹⁷ <http://exist-db.org/exist/apps/homepage/index.html>.

¹⁸ "Un Domain-Specific Language (DSL) è un linguaggio interpretabile (o compilabile) da un calcolatore, descritto da una grammatica formale (nella maggior parte dei casi da una Context-Free Grammar) e ottimizzato per uno specifico dominio di conoscenza o di attività. A differenza dei linguaggi di programmazione general purpose, come Python o Java, o di mark-up, come XML, l'essere destinati ad un numero ristretto di compiti permette ai DSL di essere concisi e familiari, per vocabolario e sintassi, agli esperti del dominio di riferimento." (Boschetti e Mugelli, 2021, p. 89)

Edizioni Critiche Digitali di [Le Varianti della Rosa](#)
esempio

Pundit

Pundit è una collezione di estensioni per il browser pensata per l'annotazione di pagine web, realizzata dall'azienda italiana Net7. Lo strumento può essere utilizzato insieme a Pundit Dashboard,¹⁹ un'applicazione web mediante la quale è possibile visualizzare tutte le annotazioni effettuare, eseguire delle ricerche sulle annotazioni ed esportarle.

<i>Sito web</i>	https://thepund.it/
<i>Anno di realizzazione</i>	2011
<i>Descrizione</i>	Web annotation made easy With Pundit Annotator you can highlight or comment on any text on the web. It is the perfect tool for your work, research, study. Pundit Annotator also helps you broaden your everyday reading on the web.
<i>Creatore/i</i>	Net7 srl
<i>Ente/i partner</i>	EU-funded projects: SEMLIB, StoM
<i>Formati di input</i>	
<i>Formati di output</i>	(For Pundit Dashboard) ODT
<i>Tecnologie utilizzate per lo sviluppo</i>	Angular, Node.JS
<i>Utilizzo</i>	Available as Chrome extension or can be used in an embedded environment
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
<i>Repository del codice sorgente</i>	https://github.com/net7/pundit-client
<i>Ultima versione rilasciata</i>	1.9.2

Recogito

Recogito è una piattaforma *online* per l'annotazione e la pubblicazione sotto forma di *open data* delle informazioni geografiche contenute all'interno di testi, immagini e database. Gli utenti, dopo essersi registrati alla piattaforma, possono creare le proprie collezioni di documenti e lavorare collaborativamente.

Sito web <https://recogito.pelagios.org/>

¹⁹ <https://thepund.it/dashboard-web-annotation/>.

<i>Anno di realizzazione</i>	2017
<i>Descrizione</i>	Linked Data annotation without the pointy brackets Crea mappe dal testo. Annota immagini. Connetti la tua ricerca ai dati che sono disponibili in rete, per renderla più visibile e disponibile per riuso. Senza dover diventare un esperto di Semantic Web.
<i>Creatore/i</i>	Pelagios Commons (Elton Barker, Leif Isaksen, Rebecca Kahn, Rainer Simon, Valeria Vitale)
<i>Ente/i partner</i>	Austrian Institute of Technology, Exeter University, Humboldt Institute for Internet and Society, The Open University and University of London. Funded by the Andrew W. Mellon Foundation
<i>Formati di input</i>	Plain Text (.txt)
<i>Formati di output</i>	CSV, RDF, GeoJSON, KML, TEI/XML
<i>Tecnologie utilizzate per lo sviluppo</i>	IIIFViewer, IIIFHosting, CETEIcean Java 8 JDK, SBT (version 1.0.x), node.js (version 10.4.1), along with npm (version 6.1.0) and the webpack and webpack-cli npm packages (install globally via <code>npm install -g {package-name}</code>), PostgreSQL DB (tested with version 9.5), Elasticsearch v5.6.5 (important: do not use ES v6.x, since this introduced breaking changes not compatible with the current version of Recogito), vips image processing system.
<i>Utilizzo</i>	Available as web application online and for local installation
<i>Consente il lavoro collaborativo</i>	Si
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/pelagios/recogito2
<i>Licenza</i>	Apache 2
<i>Ultima versione rilasciata</i>	3.3

SPEEDy

SPEEDy è un modulo per l'editing di proprietà *standoff* da associare a elementi di documenti HTML e XML, che può essere integrato in altre applicazioni web. Le ultime attività rilevabili nel *repository* GitHub di SPEEDy risalgono a due anni fa, per cui è probabile che lo strumento non sia più sviluppato. Inoltre, si può presupporre che questo strumento sia stato realizzato in forma sperimentale e pubblicato solo come prototipo, in quanto non è disponibile una *release* ufficiale.

Sito web <https://github.com/argimenes/standoff-properties-editor>

<i>Anno di realizzazione</i>	2020
<i>Descrizione</i>	<p>I use the term Standoff Property Document (SPD) to refer to the combination of the raw text and its standoff properties.</p> <p>Separation of annotations from text source</p> <p>Text is dynamically editable without corrupting annotations</p> <p>Annotations can be overlaid freely</p> <p>Supports zero-point annotations (ZPA), like word-breaking hyphens that need to be marked up from a manuscript but which shouldn't be stored in the text itself</p> <p>Documents are exported and imported as JSON</p> <p>Annotations can be grouped into layers, to manage visual complexity</p> <p>The standoff property data model can easily be extended by the user in their application code through various event hooks, to store as much as or as little data as they need</p> <p>The editor makes no assumption about where LOD data is stored, and can be easily extended to query any data source</p>
<i>Creatore/i</i>	Andreas Kuczera, Iian Neill
<i>Formati di input</i>	JSON
<i>Formati di output</i>	JSON
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript, HTML, CSS
<i>Utilizzo</i>	Module that can be used in any web application
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
<i>Repository del codice sorgente</i>	https://github.com/argimenes/standoff-properties-editor
<i>Licenza</i>	MIT
<i>Pubblicazioni</i>	<p>Neill, I. and Kuczera, A. (2019). The Codex – an Atlas of Relations. ZfdG - Zeitschrift für digitale Geisteswissenschaften. Sonderband 4. http://www.zfdg.de/sb004_008</p> <p>Neill, I. and Schmidt, D. (2020). SPEEDy A practical editor for texts annotated with standoff properties. GraphSDE2019: Workshop on Scholarly Digital Editions, Graph Data-Models and Semantic Web Technologies. Draft available at https://www.researchgate.net/publication/340006851</p>

TEITOK

TEITOK è una piattaforma web creata per la marcatura e annotazione linguistica di corpora testuali.

<i>Sito web</i>	http://teitok.corpuswiki.org/
<i>Anno di realizzazione</i>	2014
<i>Descrizione</i>	TEITOK is a web-based platform for viewing, creating, and editing corpora with both rich textual mark-up and linguistic annotation
<i>Creatore/i</i>	Maarten Janssen
<i>Ente/i partner</i>	Initially developed at the Centro de Linguística da Universidade de Lisboa, later at CELGA-ILTEC, and currently maintained at the ÚFAL institute of Charles University, Prague.
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	KWIC
<i>Tecnologie utilizzate per lo sviluppo</i>	PHP, JavaScript, CSS
<i>Utilizzo</i>	web-based Linux, MacOS
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://gitlab.com/maartenes/TEITOK
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	1.3
<i>Edizioni Critiche Digitali di esempio</i>	Oralia Diacrónica del Español (ODE) , P.S. Post Scriptum A Digital Archive of Ordinary Writing (Early Modern Portugal and Spain)
<i>Pubblicazioni</i>	Rodríguez, Pilar Arrabal. 2022. ‘TEITOK, a Visual Solution for XML/TEI Encoding: Editing, Annotating and Hosting Linguistic Corpora’. RIDE (blog). December 2022. https://doi.org/10.18716/ride.a.15.5 .

Analisi del testo

Nella tassonomia TaDiRAH è presente l'attività "analyzing" così definita: "the activity of examining any kind of information from collections of data, of discovering recurring phenomena, units, elements, patterns, groupings, and the like. This can refer to structural, formal or semantic aspects of data".²⁰

Come emerge dalla definizione, si tratta di un'attività molto generica, all'interno della quale è possibile distinguere diverse tipologie di analisi in base all'oggetto, differenziando per esempio tra "structural analysis", "stylistic analysis" e "content analysis", e in base alla metodologia utilizzata, per esempio "POS-tagging", "tree-tagging", "machine learning", ecc.

GATTO

Gatto (Gestione degli Archivi Testuali del Tesoro delle Origini) è un software lessicografico sviluppato negli anni '90 dall'Opera del Vocabolario Italiano per allestire il corpus TLIO (Tesoro della Lingua Italiana delle Origini). Il software richiede in *input* documenti di testo in formato TXT, all'interno dei quali il testo deve essere stato marcato secondo un sistema di codifica interno (ovvero stabilito dagli sviluppatori di Gatto). Gatto è disponibile anche in una versione web, chiamata Gattoweb.²¹ L'interfaccia grafica consente di effettuare ricerche per lemmi o per forme e di visualizzare i risultati all'interno dei propri contesti.

<i>Sito web</i>	http://www.ovi.cnr.it/Il-Software.html
<i>Anno di realizzazione</i>	1995
<i>Descrizione</i>	<p>Il programma di punta è GATTO, un software lessicografico ideato e sviluppato da Domenico Iorio-Fili presso l'OVI. Questo software è nato come strumento finalizzato alla costruzione, gestione ed interrogazione del corpus di testi che è alla base del Vocabolario Storico della Lingua Italiana, in corso di realizzazione presso l'OVI.</p> <p>GATTO gestisce quindi, oltre a basi testuali di altri gruppi di studiosi, il corpus testuale dell'OVI.</p> <p>Più in generale GATTO permette di effettuare ricerche lessicografiche su un generico archivio testuale a condizione che i testi, opportunamente codificati, siano disponibili su file unitamente ai relativi dati bibliografici.</p>
<i>Creatore/i</i>	Domenico Iorio-Fili
<i>Ente/i partner</i>	OVI-CNR
<i>Tecnologie utilizzate</i>	Visual Basic 6.0

²⁰ <https://vocabs.dariah.eu/tadirah/en/page/analyzing>.

²¹ <http://gattoweb.ovi.cnr.it/>.

per lo sviluppo

<i>Utilizzo</i>	Desktop Online (GattoWeb)
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	No
<i>Licenza</i>	Gratuita
<i>Ultima versione rilasciata</i>	3.3

LEAF-Writer

Nel contesto del Linked Editing Academic Framework (LEAF)²² è stato sviluppato un *editor online*, LEAF - Writer*, per documenti XML e RDF. Oltre all'editing di documenti XML/TEI *ex novo* oppure a partire da template, LEAF-Writer permette di: estrarre automaticamente i riferimenti a nomi di luoghi, persone e organizzazioni (in inglese *named entities*, entità nominate), recuperando i rispettivi tag TEI presenti nel documento; cercare e selezionare gli identificatori per i tag delle *named entities*, da diversi servizi di Linked Open Data tra cui: DBPedia, Geonames e VIAF; generare annotazioni per i linked data in XML-RDF o JSON-LD conformi con il Web Annotation Data Model.

<i>Sito web</i>	https://leaf-writer.leaf-vre.org/
<i>Anno di realizzazione</i>	2022
<i>Descrizione</i>	LEAF-Writer provides: schema-aware document editing, including validation, against web-accessible schemas support for cascading stylesheets (CSS) to provide a WYSIWYG view of documents, as well as a view showing tags the ability to extract references to named entities (persons, places, or organizations) from already tagged XML references within a document in order to generate Web Annotation the ability to look up and select identifiers for named entity tags (persons, organizations, places, or titles) from the following Linked Open Data authorities: DBPedia, Geonames, Getty, LGPN, VIAF, and Wikidata. generation of Linked Data annotations corresponding to newly tagged named entities and document annotations (dates, notes, citations, corrections, links, keywords) in XML-RDF or JSON-LD conforming to the Web Annotation Data Model

²²LEAF è un *virtual research environment* sviluppato da Bucknell University, Newcastle University, University of Guelph per supportare progetti collaborativi di *digital humanities* di larga scala (per maggiori informazioni si rimanda a <https://www.leaf-vre.org/docs/about-leaf/about>).

	continuous XML validation Schema-constrained tagging options
<i>Creatore/i</i>	Project director: Susan Brown
<i>Ente/i partner</i>	Linked Editing Academic Framework (LEAF) Canadian Writing Research Collaboratory, Islandora Foundation, Bucknell University, Newcastle University, University of Guelph
<i>Formati di input</i>	XML-TEI, RDF, XML-RDF
<i>Formati di output</i>	XML-RDF, JSON-LD
<i>Tecnologie utilizzate per lo sviluppo</i>	TinyMCE editor
<i>Utilizzo</i>	Online Web-based
<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
Repository del codice sorgente	https://gitlab.com/calincs/cwrc/leaf-writer/leaf-writer/
<i>Licenza</i>	GLP-2.0
<i>Ultima versione rilasciata</i>	1.0.0

PIE

PIE è un *framework* per la lemmatizzazione e il *POS-tagging* (letteralmente ‘marcatura delle parti del discorso’),²³ liberamente disponibile in *open source*, che può essere utilizzato insieme a Pyrrha, applicativo web realizzato da alcuni sviluppatori dell’École Nationale des Chartes, ideato per velocizzare la correzione di corpora lemmatizzati e taggati a livello morfo-sintattico.

<i>Sito web</i>	https://zenodo.org/badge/latestdoi/131014015
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	PIE: A Framework for Joint Learning of Sequence Labeling Tasks PIE was primarily conceived to make experimentation on sequence labeling of variation-rich languages easy and user-friendly. PIE has been tested

²³ Con POS-tagging (letteralmente ‘marcatura delle parti del discorso’) si intende: “the process of identifying the grammatical type of words in a text” (<https://vocabs.dariah.eu/tadirah/en/page/posTagging>).

mostly for Lemmatization but other SoTA accuracies from other tasks like POS have been reproduced (cf. Plank et al). PIE is highly configurable both in terms of input preprocessing and model definition, in principle not requiring users to write any code (instead experiments are defined with json files). It is highly modular and therefore easy to extend. It includes transductive lemmatization as an additional sequence labeling task and, finally, it is reasonably fast and memory efficient.

<i>Creatore/i</i>	Enrique Manjavacas, Thibault Clérice, Mike Kestemont
<i>Formati di input</i>	Plain Text (.txt)
<i>Formati di output</i>	TXT
<i>Tecnologie utilizzate per lo sviluppo</i>	Python
<i>Utilizzo</i>	Desktop
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
Repository del codice sorgente	https://github.com/emanjavacas/pie/tree/v0.3.7c
<i>Licenza</i>	MIT
<i>Ultima versione rilasciata</i>	0.3.7

Pyrrha

Vedi PIE.

<i>Sito web</i>	https://dh.chartes.psl.eu/pyrrha/
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	Pyrrha is a simple Python Flask WebApp to fasten the post-correction of lemmatized and morpho-syntactic tagged corpora.
<i>Creatore/i</i>	Julien Pilla and Thibault Clérice
<i>Ente/i partner</i>	École nationale des chartes
<i>Tecnologie utilizzate per lo sviluppo</i>	Python Flask WebApp, JavaScript, HTML
<i>Utilizzo</i>	Web-based
<i>Consente il lavoro collaborativo</i>	No

Open source	Si
Repository <i>del codice sorgente</i>	https://github.com/hipster-philology/pyrrha
<i>Licenza</i>	MIT
<i>Ultima versione rilasciata</i>	3.0.0

TEI Critical Apparatus Toolbox

TEI Critical Apparatus Toolbox è uno strumento ideato per facilitare il lavoro di *editing* e di revisione di un'edizione in XML/TEI. È stato sviluppato quasi interamente da Marjorie Burghart, filologa mediolatina, per sopperire alla mancanza di strumenti per lavorare con XML/TEI che supportassero anche il modulo per l'apparato critico. L'applicativo consente di: revisionare le codifiche; visualizzare sinotticamente diverse versioni di un medesimo testo; stampare l'edizione trasformando i file XML in PDF mediante TEI-to-LateX; annotare le immagini, in modo tale da creare delle zone all'interno dei file TEI per preparare un'edizione facsimile o digital documentary edition; avere delle statistiche sui tag XML utilizzati nei propri documenti e il conteggio delle parole.

<i>Sito web</i>	http://teicat.huma-num.fr/index.php
<i>Anno di realizzazione</i>	2014
<i>Descrizione</i>	The TEI Critical Apparatus Toolbox is a tool for people preparing a natively digital TEI critical edition.
<i>Creatore/i</i>	Marjorie Burghart
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	PDF, LaTeX, HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	XSLT (TEI BoilerPlate stylesheets)
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	No
Open source	No

Tree Tagger

Tree Tagger è uno strumento per l'annotazione di testi e la lemmatizzazione, sviluppato intorno alla metà degli anni '90 da Helmut Schmid presso l'Istituto per la Linguistica Computazionale dell'Università di Stoccarda. È un software usato per numerose lingue diverse e può essere utilizzato potenzialmente per qualsiasi lingua se si dispone di un lessico e di un corpus di training taggato manualmente. Il software è liberamente disponibile se usato per scopi accademici e di ricerca.

<i>Sito web</i>	https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
<i>Anno di realizzazione</i>	1995
<i>Descrizione</i>	The TreeTagger is a tool for annotating text with part-of-speech and lemma information. It was developed by Helmut Schmid in the TC project at the Institute for Computational Linguistics of the University of Stuttgart. The TreeTagger has been successfully used to tag German, English, French, Italian, Danish, Swedish, Norwegian, Dutch, Spanish, Bulgarian, Russian, Portuguese, Belarusian, Ukrainian, Galician, Greek, Chinese, Swahili, Slovak, Slovenian, Latin, Estonian, Polish, Persian, Romanian, Czech, Coptic and old French texts and is adaptable to other languages if a lexicon and a manually tagged training corpus are available
<i>Creatore/i</i>	Helmut Schmid
<i>Ente/i partner</i>	Institute for Computational Linguistics of the University of Stuttgart
<i>Utilizzo</i>	Desktop PC-Linux, Windows, Mac-OS
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	No
<i>Licenza</i>	Gratuita
<i>Pubblicazioni</i>	Helmut Schmid (1995): Improvements in Part-of-Speech Tagging with an Application to German. Proceedings of the ACL SIGDAT-Workshop. Dublin, Ireland. Helmut Schmid (1994): Probabilistic Part-of-Speech Tagging Using Decision Trees. Proceedings of International Conference on New Methods in Language Processing, Manchester, UK.

Voyant Tools

Voyant Tools è un applicativo web che permette di analizzare un testo o un corpus di testi sotto vari punti di vista, per esempio quali sono le parole più frequenti, mediante diverse visualizzazioni grafiche *ad hoc*. È uno strumento pensato sia per ricercatori e studenti di *digital humanities* sia per un pubblico generico.

<i>Sito web</i>	https://voyant-tools.org/
<i>Anno di realizzazione</i>	2016
<i>Descrizione</i>	Voyant Tools is a web-based text reading and analysis environment. It is a scholarly project that is designed to facilitate reading and interpretive practices for digital humanities students and scholars as well as for the general public. You can find out more about Voyant or read the documentation on
<i>Creatore/i</i>	Stéfan Sinclair and Geoffrey Rockwell

<i>Formati di input</i>	TXT (Unicode), XML-TEI, HTML, XML, Microsoft Word (.doc, RTF, PDF, Excel
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	2.6.5
<i>Pubblicazioni</i>	Alhudithi, Ella. 2021. ‘Review of Voyant Tools: See through Your Text’. <i>Language Learning & Technology</i> 25 (3): 43–50.

Strumenti di visualizzazione

Per inquadrare gli strumenti di visualizzazione ci si può servire di due attività della tassonomia TADIRAH. Si può usare un generico “publishing”, definito come: “the activity of making any kind of object formally available to the wider public. This can involve objects of research, research data, research results, or tools and services. Publishing can be closed or open access / open source, and research results can be published in print or digital formats”.²⁴ Eventualmente si può specificare che si tratta di “academic publishing” e di “digital publishing”. Oppure si può usare l’attività “data visualization”, anche se la definizione fornita nella tassonomia potrebbe essere fuorviante rispetto a quanto si intende con “strumento di visualizzazione”: “creation and study of the visual representation of data”.²⁵ L’attività “publishing” e le sue sotto-attività sono da preferirsi, in quanto più coerenti.

CETEIcean

CETEIcean è una libreria JavaScript *open source*, ideata e sviluppata da Hugh Cayless (Duke University) e Raffaele Viglianti (Maryland Institute for Technology in the Humanities) nel 2017. In modo simile a TEIBoilerplate, dal quale è stato ispirato, il programma consente di visualizzare documenti in formato XML/TEI direttamente all’interno di un browser: il programma trasforma dinamicamente gli elementi XML/TEI in

²⁴ <https://vocabs.dariah.eu/tadirah/en/page/publishing>.

²⁵ <https://vocabs.dariah.eu/tadirah/en/page/dataVisualization>.

*custom elements*²⁶ nel momento in cui il documento XML/TEI viene caricato nel browser. CETEIcean consente un alto livello di personalizzazione, sia dell’aspetto grafico degli elementi sia del loro comportamento e modalità di interazione. Inoltre, si tratta di uno strumento concepito per realizzare edizioni critiche digitali che da un punto di vista tecnologico siano fondate sugli standard del web (HTML, CSS e JavaScript) e che al contempo si presentino come delle pagine web statiche, così da risultare più durevoli e facili da mantenere nel tempo. CETEIcean è stato adottato, per esempio, per realizzare le “micro-editions”,²⁷ ovvero delle edizioni volutamente contenute sia nei contenuti che nelle funzionalità tecnologiche, della rivista *Scholarly Editing*.²⁸

<i>Sito web</i>	https://github.com/TEIC/CETEIcean
<i>Anno di realizzazione</i>	2016
<i>Descrizione</i>	<p>CETEIcean is a Javascript library that allows TEI documents to be displayed in a web browser without first transforming them to HTML. It uses the emerging Web Components standards, especially Custom Elements. It works by loading the TEI file dynamically, renaming the elements to follow the Custom Elements conventions, and registering them with the browser. Browsers that support Web Components will use them to add the appropriate display and behaviors to the TEI elements; other browsers will use fallback methods to achieve the same result.</p> <p>Because it preserves the full structure and information from your TEI data model, CETEIcean allows you to build rich web applications from your source documents using standard tools like CSS and Javascript.</p>
<i>Creatore/i</i>	Hugh Cayless and Raffaele Viglianti
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	HTML5
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript
<i>Utilizzo</i>	Web-based
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://github.com/TEIC/CETEIcean
<i>Licenza</i>	BSD 2-Clause “Simplified”

²⁶ <https://html.spec.whatwg.org/multipage/custom-elements.html#custom-element>.

²⁷ Come esempio di “micro-edition” si rimanda a quella del testo teatrale *Columbus* di John Brougham, curata da Laura L. Mielke e Rachel Linnea Brown e pubblicata nell’ultimo numero della rivista: <https://scholarlyediting.org/issues/39/john-broughams-columbus-burlesque/>.

²⁸ Sito web dell’associazione *Scholarly Editing* e della relativa rivista: <https://scholarlyediting.org/>.

Ultima versione rilasciata 1.8.0

Edizioni Critiche Digitali di [Micro-Editions of the Scholarly Editing Journal](#), [The Early Modern Songscapes \(EMS\) project](#)

Pubblicazioni Cayless, Hugh. “Critical editions and the data model as interface.” *Digital Scholarly Editions as Interfaces* (2018): 249-263.

EVT (Edition Visualization Technology)

EVT è un software *open source* e *client only*²⁹ per la visualizzazione di edizioni in formato XML/TEI, ideato da Roberto Rosselli Del Turco e realizzato da un gruppo di studenti del corso di laurea in Informatica umanistica dell'Università di Pisa. Nato come strumento di visualizzazione *ad hoc* per l'edizione digitale del Vercelli Book,³⁰ EVT è diventato adattabile a edizioni diverse grazie alla collaborazione con altri progetti scientifici, in particolare il Codice Pelavicino Digitale,³¹ che se ne sono serviti e che hanno contribuito allo sviluppo. La prima versione del software è stata concepita per supportare edizioni di tipo diplomatico-interpretativo (quelle che sul web possono essere genericamente etichettate “digital documentary edition”), all'interno delle quali le riproduzioni fotografiche di un manoscritto o di un altro tipo di documento sono affiancate dalle rispettive trascrizioni, corredate da annotazioni editoriali. Questa prima versione, da un punto di vista tecnologico, si basa su una serie di trasformazioni XSLT che a partire da un documento XML/TEI genera una pagina HTML. La seconda versione del software, invece, è stata sviluppata mediante il *framework*³² AngularJS³³ ed è nata per supportare edizioni critiche, ovvero dotate di apparato critico. In un secondo momento questa versione è stata ampliata, in modo tale da supportare tutte le funzionalità che erano disponibili nella prima versione. Tra le edizioni realizzate con EVT 2 si rimanda come esempio all'edizione digitale della *Storia fiorentina* di Benedetto Varchi,³⁴ realizzata nell'ambito del /DH.arc (Digital Humanities Advanced Research Centre) dell'Università di Bologna, in collaborazione con la Concordia University di Montreal e CarisBo. La *release* della terza versione del software è in programma nei prossimi mesi e sarà un *reboot* della seconda versione, realizzata, tuttavia, con Angular,³⁵ un *framework* JavaScript più moderno.

Sito web <http://evt.labcd.unipi.it/>

²⁹ L'espressione *client only* (letteralmente “solo cliente”) individua gli applicativi che per funzionare non necessitano di rapportarsi con un *server*. Una volta che l'applicativo viene caricato nel *browser* dell'utente (cioè dal cliente), tutte le sue funzionalità non richiedono la comunicazione con un server per poter essere eseguite.

³⁰ Sito web dell'edizione digitale del Vercelli Book: <http://vbd.humnet.unipi.it/beta2/>.

³¹ Sito web dell'edizione digitale del Codice Pelavicino: <https://pelavicino.labcd.unipi.it/>.

³² In informatica con ‘framework’ (letteralmente ‘impalcatura’, ‘struttura’) si intende un insieme di librerie e/o di classi riutilizzabili per la scrittura di software, che può essere usato per velocizzare la produzione di nuovo codice, scrivendo *ex novo* solo il codice necessario per il software che si intende produrre.

³³ AngularJS è un *framework* per lo sviluppo dell'interfaccia grafica di applicativi web in JavaScript (<https://angularjs.org/>).

³⁴ Sito web dell'edizione digitale della *Storia fiorentina* di Benedetto Varchi: <https://dharc-org.github.io/progetto-vasto/index.html>.

³⁵ Angular è un framework per lo sviluppo dell'interfaccia grafica di applicativi web in JavaScript, realizzato dagli stessi autori di AngularJS come completa riscrittura. Rispetto alla versione precedente è più potente ed efficiente (<https://angular.io/>).

<i>Anno di realizzazione</i>	2013
<i>Descrizione</i>	A light-weight, open source tool specifically designed to create digital editions from XML-encoded texts, freeing the scholar from the burden of web programming and enabling the final user to browse, explore and study digital editions by means of a user-friendly interface.
<i>Creatore/i</i>	Roberto Rosselli Del Turco et al.
<i>Ente/i partner</i>	University of Pisa, University of Turin
<i>Formati di input</i>	XML-TEI, JPG
<i>Formati di output</i>	HTML, CSS, JS
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript, CSS, HTML, OpenSeadragon, VISCOLL. EVT 1: XSLT; EVT 2: AngularJS.
<i>Utilizzo</i>	Web-based client-only application
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
Repository del codice sorgente	https://github.com/evt-project/evt-viewer
<i>Licenza</i>	AGPL-3.0
<i>Ultima versione rilasciata</i>	EVT2 beta2
<i>Edizioni Critiche Digitali di esempio</i>	Codice Pelavicino Digitale, Tarsian. Digital Edition of an Ancient New Testament Arabic Manuscript, The Digital Vercelli Book, Petri de Ebulo De rebus Siculis carmen, “Progetto VaSto - Varchi, Storia Fiorentina Edizione Digitale”
<i>Pubblicazioni</i>	Del Turco, Roberto Rosselli, Chiara Di Pietro, and Chiara Martignano. “Progettazione e implementazione di nuove funzionalità per EVT 2: lo stato attuale dello sviluppo.” <i>Umanistica digitale</i> 7 (2019). Del Turco, Roberto Rosselli. “Designing an advanced software tool for Digital Scholarly Editions.” <i>Textual cultures</i> 12.2 (2019): 91-111. Del Turco, Roberto Rosselli. “After the editing is done: Designing a Graphic User Interface for digital editions.” <i>Digital Medievalist</i> 7 (2012).

Muruca

Muruca è un CMS (*content management system*) ideato dall'azienda italiana Net7 per creare collezioni online di oggetti culturali. Si tratta di uno strumento commerciale, disponibile anche in *open source*, che si rivolge a biblioteche, archivi e studiosi. Funziona in modo analogo a Omeka e, grazie all'integrazione di TEIPublisher, consente di pubblicare edizioni in formato XML/TEI.

<i>Sito web</i>	https://www.muruca.org/
<i>Anno di realizzazione</i>	2021
<i>Descrizione</i>	Le tue collezioni, online. Fai parte di un gruppo di ricerca, un museo, una fondazione, un'istituzione culturale? Muruca è la soluzione che ti consente di mettere online i tuoi contenuti in modo facile e veloce. Pubblica, condividi, promuovi cultura e ricerca, con Muruca.
<i>Creatore/i</i>	Net7 srl
<i>Tecnologie utilizzate per lo sviluppo</i>	TEI Publisher, Wordpress
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://github.com/net7
<i>Licenza</i>	A pagamento
<i>Pubblicazioni</i>	Aiola, Chiara, Giulio Andreini, Francesca Di Donato, and Tiziana Lombardo. 2021. 'Sharing Knowledge Digitally; the Muruca Case Study'. In AIUCD 2021, DH per la società: e-guaglianza, partecipazione, diritti e valori nell'era digitale. Pisa.

Omeka

Omeka è un CMS per la pubblicazione sul web di collezioni digitali. È stato sviluppato dal progetto Digital Scholar,³⁶ un gruppo di storici e sviluppatori software che hanno realizzato anche Zotero.

<i>Sito web</i>	https://omeka.org/
<i>Anno di realizzazione</i>	2008
<i>Descrizione</i>	Open-source web publishing platforms for sharing digital collections and creating media-rich online exhibits. Omeka S is a next-generation web publishing platform for institutions interested in connecting digital cultural heritage collections with other resources online. Omeka Classic is a web publishing platform for sharing digital collections and creating media-rich online exhibits.
<i>Creatore/i</i>	Digital Scholar

³⁶ <https://digitalscholar.org/>.

Alyssa Toby Fahringer, Jeremy Boggs, Jim Safley, John Flatness, Ken Albers, Kim Nguyen, Megan Brett, Patrick Murray-John, Sharon Leon, Sheila Brennan e Tom Scheinfeldt

<i>Ente/i partner</i>	Roy Rosenzweig Center for History and New Media
<i>Formati di input</i>	CSV, OAI-PMH, XML-TEI
<i>Tecnologie utilizzate per lo sviluppo</i>	PHP, HTML, SCSS, JavaScript TinyMCE
<i>Utilizzo</i>	Web-based
<i>Consente il lavoro collaborativo</i>	Si
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/omeka
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	Omeka S: 4.0.1; Omeka Classic: 3.1.1
<i>Pubblicazioni</i>	Leblanc, Elina. n.d. 'Omeka Classic. Un environnement de recherche pour les éditions scientifiques numériques'. RIDE (blog). Accessed 31 May 2023. https://ride.i-d-e.de/issues/issue-11/omeka/ .

TAPAS

TAPAS è uno strumento sviluppato con il supporto del consorzio TEI e un gruppo di università statunitensi, pensato per l'archiviazione e condivisione online di documenti marcati in XML/TEI. All'interno di TAPAS per ogni progetto l'utente può visualizzare sia il documento XML "nudo" sia le visualizzazioni di quel documento generate con fogli di stile sviluppati da TAPAS oppure con TEIBoilerplate. Per le edizioni grazie ai fogli di stile TAPAS è possibile generare *on the fly* una visualizzazione di livello diplomatico e una di livello interpretativo. Oltre ai documenti XML/TEI è possibile caricare dei file immagine.

<i>Sito web</i>	https://tapasproject.org/
<i>Anno di realizzazione</i>	2013
<i>Descrizione</i>	TAPAS is the TEI Archiving, Publishing, and Access Service hosted by Northeastern University Library's Digital Scholarship Group.
<i>Creatore/i</i>	Syd Bauman, TEI analyst Ash Clark, XML programmer Julia Flanders, TAPAS co-director Scott Hamlin, TAPAS co-director Patrick Murray-John, Associate Director for Systems, Digital Scholarship Group

<i>Ente/i partner</i>	Northeastern University Library's Digital Scholarship Group, TEI Consortium, Institute of Museum and Library Sciences (IMLS), and the National Endowment for the Humanities (NEH)
<i>Formati di input</i>	XML-TEI
<i>Tecnologie utilizzate per lo sviluppo</i>	TEIBoilerplate
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	Si
<i>Open source</i>	No
<i>Licenza</i>	Gratuita, a pagamento la versione con maggiori funzionalità
<i>Edizioni Critiche Digitali di esempio</i>	Digital Editions of Valentine and Orson

TEIBoilerplate

TEIBoilerplate, realizzato nel 2012 presso l'Università dell'Indiana, consente di visualizzare all'interno di un browser un documento XML/TEI, senza il bisogno di elaborazioni lato server. L'utente può personalizzare la formattazione grafica del documento risultante, inserendo delle indicazioni specifiche all'interno del file TEI sorgente oppure servendosi di fogli di stile CSS.

<i>Sito web</i>	http://teiboilerplate.org/
<i>Anno di realizzazione</i>	2012
<i>Descrizione</i>	TEI Boilerplate is a lightweight solution for publishing styled TEI (Text Encoding Initiative) P5 content directly in modern browsers. With TEI Boilerplate, TEI XML files can be served directly to the web without server-side processing or translation to HTML. Our TEI Boilerplate Demo illustrates many TEI features rendered by TEI Boilerplate.
<i>Creatore/i</i>	John Walsh, Grant Simpson and Saeed Moaddeli (Indiana University)
<i>Ente/i partner</i>	Indiana University
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	HTML5
<i>Tecnologie utilizzate per lo sviluppo</i>	XSLT, JavaScript, CSS, HTML
<i>Utilizzo</i>	Web-based

<i>Consente il lavoro collaborativo</i>	No
Open source	Si
Repository del codice sorgente	https://github.com/TEI-Boilerplate/TEI-Boilerplate
<i>Licenza</i>	Creative Commons Attribution 3.0 Unported License
<i>Edizioni Critiche Digitali di esempio</i>	Petrarche. An edition of Petrarch's songbook <i>Rerum vulgarium fragmenta</i>

TEIGarage

TEIGarage è la nuova versione di OxGarage, un applicativo web disponibile in *open access*, che consente di convertire documenti da TEI a numerosi altri formati e viceversa da numerosi formati diversi al formato TEI, semplicemente caricando il documento da convertire nella pagina web, selezionando il formato di output e scaricando il nuovo documento prodotto. Realizzato nel 2021, TEIGarage rispetto al suo predecessore permette anche di accedere al servizio di conversione tramite API.

<i>Sito web</i>	https://teigarage.tei-c.org/
<i>Anno di realizzazione</i>	2021
<i>Descrizione</i>	TEIGarage is a webservice and RESTful service to transform, convert and validate various formats, focussing on the TEI format. TEIGarage is based on the proven OxGarage.
<i>Creatore/i</i>	Anne Ferger, Peter Stadler, Daniel Röwenstrunk
<i>Ente/i partner</i>	TEI Consortium
<i>Formati di input</i>	Cocoa tagging, Compiled TEI ODD, DocBook, Markdown, Microsoft Word (.doc, .docx), ODD, OpenOffice (.sxw, .odt), Plain Text (.txt), Rich Text Format (.rtf), TCP XML, TEI P4 XML, TEI P5 XML, TEI Tite XML, WordPerfect (.wpd), Wordpress RSS feed of blog, xHTML
<i>Formati di output</i>	CSV, DocBook, ePub3, LaTeX, Markdown, Microsoft Word (.doc, .docx), National Library of Medicine (NLM) DTD 3.0, OpenOffice (.sxw, .odt), PDF, Plain text, RDF XML, Rich Text Format (.rtf), TEI Simple XML, VerbatimXML, xHTML, XML, XSL-FO
<i>Tecnologie utilizzate per lo sviluppo</i>	Java, HTML, XSLT, Docker, Maven
<i>Utilizzo</i>	Online RESTful API
<i>Consente il lavoro collaborativo</i>	No

Open source	Si
Repository <i>del codice sorgente</i>	https://github.com/TEIC/teigarage
Licenza	GPL-3.0
Ultima versione rilasciata	1.1.1
Pubblicazioni	Stadler, Peter, Anne Ferger, and Daniel Röwenstrunk. 2022. "From OxGarage to TEIGarage and MEIGarage". September 9. https://doi.org/10.5281/zenodo.7061525 .

TEIPublisher

Realizzato nel 2015 dall'azienda eXist Solutions³⁷ nel contesto dell'associazione svizzera e-codices,³⁸ TEIPublisher consente di pubblicare sul web file XML (prevalentemente secondo lo standard TEI), generando un sito web, oppure di convertire i documenti in altri formati, quali PDF e ePUB.³⁹ Da un punto di vista tecnologico, l'applicativo è basato su: il database per document XML eXist-db;⁴⁰ il TEI Processing Model,⁴¹ una sezione delle linee guida TEI che consente di fornire all'interno di un documento TEI le istruzioni per la sua elaborazione e visualizzazione in output; API (definite secondo le specifiche di Open API),⁴² per gestire la comunicazione tra *client* e *server*; infine, i *web components*,⁴³ dei componenti web personalizzati implementati sotto forma di tag HTML. TEIPublisher permette di personalizzare l'aspetto grafico e l'interattività dei diversi elementi contenuti in un documento TEI mediante un file TEI ODD.

<i>Sito web</i>	https://teipublisher.com/index.html
<i>Anno di realizzazione</i>	2015
<i>Descrizione</i>	TEI Publisher facilitates the integration of the TEI Processing Model into exist-db applications. The TEI Processing Model (PM) extends the TEI ODD specification format with a processing model for documents. That way intended processing for all elements can be expressed within the TEI vocabulary itself. It aims at the XML-savvy editor who is familiar with TEI but is not necessarily a developer.
<i>Creatore/i</i>	eXist Solutions

³⁷ <https://www.existsolutions.com/>.

³⁸ <https://www.e-editiones.org/>.

³⁹ Il formato ePUB (che sta per *electronic publication*) è uno standard aperto per e-book creato dall'International Digital Publishing Forum (IDPF). Negli anni è diventato il formato di riferimento per la pubblicazione di e-book e ha guadagnato popolarità in quanto è un formato basato su XML e libero, cioè non è un formato proprietario di aziende che producono e-reader e/o e-book. Per maggiori informazioni si rimanda alla seguente pagina nel sito del W3C: <https://www.w3.org/AudioVideo/ebook/>.

⁴⁰ <http://exist-db.org/exist/apps/homepage/index.html>.

⁴¹ <https://www.tei-c.org/release/doc/tei-p5-doc/en/html/TD.html#TDPMPM>.

⁴² <https://spec.openapis.org/oas/latest.html>.

⁴³ <https://www.webcomponents.org/>.

<i>Ente/i partner</i>	e-editiones
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	HTML, XSL-FO, LaTeX, ePUB
<i>Tecnologie utilizzate per lo sviluppo</i>	eXist-db, XSLT, ODD
<i>Utilizzo</i>	Online eXist-db application
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://github.com/eeditiones/tei-publisher-app
<i>Licenza</i>	GPL-3.0
<i>Ultima versione rilasciata</i>	8.0.0
<i>Edizioni Critiche Digitali di esempio</i>	Van Gogh Letters (demo) , The Bodleian First Folio (A digital facsimile of the First Folio of Shakespeare's plays, Bodleian Arch. G c.7.)
<i>Pubblicazioni</i>	

Versioning Machine

Versioning Machine* è un framework e interfaccia web che permette di visualizzare e confrontare molteplici versioni di un medesimo testo (per esempio i diversi testimoni di un'opera) codificati in XML/TEI. Ogni versione da confrontare viene visualizzata all'interno di un pannello le cui dimensioni possono essere regolate a piacere dall'utente e che può essere liberamente spostato nell'interfaccia. Versioning Machine consente anche il confronto tra un testo e un'immagine. I punti del testo in cui le diverse versioni differiscono tra loro sono evidenziati mediante riquadri, collegati fra di loro da linee, così da facilitare l'utente nella loro individuazione.

<i>Sito web</i>	http://v-machine.org/
<i>Anno di realizzazione</i>	2002
<i>Descrizione</i>	The Versioning Machine is a framework and an interface for displaying multiple versions of text encoded according to the Text Encoding Initiative (TEI) Guidelines, and is P5 compatible. While the VM provides for features typically found in critical editions, such as annotation and introductory material, it also takes advantage of the opportunities afforded by electronic publication to allow for the comparison diplomatic versions of witnesses,

and the ability to easily compare an image of the manuscript with a diplomatic version. VM 5.0 adds a number of new features, including the ability to resize and reorganize text panels, panning and zooming in the image viewer, and text-audio interlinking. The Versioning Machine’s underlying code has also been completely revised to support enhanced features.

The Versioning Machine is a useful tool for textual editors, providing an environment that allows editors to immediately see the consequences of their editorial decisions. The platform also has applications in teaching, translation, and digital publication. The many uses of the Versioning Machine are illustrated in the new VM IN USE section.

<i>Creatore/i</i>	Susan Schreibman
<i>Ente/i partner</i>	Maynooth University, Digital Humanities Observatory, University of Maryland Libraries, Maryland Institute for Technology in the Humanities, and New Jersey Institute of Technology
<i>Formati di input</i>	XML-TEI, TEI P5 XML
<i>Formati di output</i>	HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	XSLT
<i>Utilizzo</i>	Directory that can be installed on any hard drive. Optimized for use with Mozilla Firefox and Google Chrome
<i>Consente il lavoro collaborativo</i>	No
Open source	No
<i>Licenza</i>	Gratuita
<i>Ultima versione rilasciata</i>	5.0
<i>Pubblicazioni</i>	Schreibman, Susan, Amit Kumar, and Jarom McDonald. “The versioning machine.” <i>Literary and Linguistic Computing</i> 18.1 (2003): 101-107.

Visualizzazione di immagini

Per descrivere gli strumenti per la visualizzazione di immagini l’attività più adatta della tassonomia TaDiRAH può essere quella di “sharing” che, alla stessa stregua di “publishing”, rientra nella categoria di livello più alto “disseminating” ed è definita come: “the activity of making objects such as digital information, documents, files, software and other publically findable, accessible, interopera-

ble and reusable according to the fair principles and in the sense of open science through communication channels such as blogs, code sharing sites such as GitHUB, social media sites or other web-based services, collaborative and network applications”.⁴⁴

Nei casi in cui lo strumento consenta di realizzare delle edizioni in facsimile, come Manicule, potrebbe essere categorizzato con l’attività “publishing”.

Manicule

Manicule* è un applicativo web *standalone*⁴⁵ e *open source* per creare edizioni digitali in facsimile. Oltre a una funzionalità molto simile a quella offerta da VisColl mediante la quale è possibile visualizzare la struttura del libro a stampa o codice, Manicule consente di: creare dei tour guidati degli aspetti caratteristici del libro o codice; aggiungere delle note in margine alle pagine; categorizzare e distinguere per colore le pagine così da fornire all’utente una *bird’s eye view*⁴⁶ o mappa del libro/codice.

<i>Sito web</i>	https://digitalbookhistory.com/manicule/
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	Manicule is a standalone React/Redux web application for presenting unique printed books and manuscripts in digital facsimile. Manicule is a standalone React/Redux web application for presenting unique printed books and manuscripts in digital facsimile. It allows editors to: build guided tours through a book’s distinguishing features; annotate the edges of interesting pages with extra information; categorize and color-code each page in the facsimile, giving a bird’s-eye view of a book’s main elements; and visualize the book’s structure.
<i>Creatore/i</i>	Liza Daly and Whitney Trettien
<i>Ente/i partner</i>	Price Lab for Digital Humanities at the University of Pennsylvania
<i>Formati di input</i>	JPG, JSON, XML
<i>Tecnologie utilizzate per lo sviluppo</i>	TypeScript, CSS, HTML, JavaScript, React, Redux, VisColl
<i>Utilizzo</i>	Web-based
<i>Consente il lavoro collaborativo</i>	No

⁴⁴ <https://vocabs.dariah.eu/tadirah/en/page/sharing>.

⁴⁵ Nel campo dell’ingegneria del software con il termine ‘standalone’ (letteralmente ‘autonomo’) si intende un programma o un software che per essere eseguito non necessita dei servizi di un sistema operativo, dunque non deve essere installato.

⁴⁶ Letteralmente ‘vista a volo d’uccello’, indica la visualizzazione di un oggetto dall’alto.

Open source	Si
Repository <i>del codice sorgente</i>	https://github.com/wtrettien/manicule
<i>Licenza</i>	MIT
<i>Ultima versione rilasciata</i>	2.0.0

Mirador

OpenSeadragon consente di visualizzare l'immagine ad alti livelli di zoom, ma ha il limite di presentare una sola immagine alla volta e senza i relativi metadati; Mirador e UniversalViewer, invece, mostrano i metadati per la risorsa nella sua interezza e consentono altre funzionalità, quali il *download* e l'aggiunta di annotazioni al file *manifest* delle immagini.

<i>Sito web</i>	https://projectmirador.org/
<i>Anno di realizzazione</i>	2015
<i>Descrizione</i>	Open-source, web based, multi-window image viewing platform with the ability to zoom, display, compare and annotate images from around the world.
<i>Creatore/i</i>	Project Mirador
<i>Formati di input</i>	IIIF Manifest
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript, HTML
<i>Utilizzo</i>	JS library
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
Repository <i>del codice sorgente</i>	https://github.com/projectmirador/mirador
<i>Ultima versione rilasciata</i>	3.3.0

OpenSeadragon

Vd. Mirador.

<i>Sito web</i>	https://openseadragon.github.io/
<i>Anno di realizzazione</i>	2010
<i>Descrizione</i>	An open-source, web-based viewer for high-resolution zoomable images, implemented in pure JavaScript, for desktop and mobile.

<i>Creatore/i</i>	Open Collective
<i>Formati di input</i>	IIIF Manifest, TIFF, JPEG, PNG
<i>Tecnologie utilizzate per lo sviluppo</i>	JavaScript, HTML
<i>Utilizzo</i>	JS library Integrate via API
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
Repository del codice sorgente	https://github.com/openseadragon/openseadragon
<i>Licenza</i>	BSD-3-Clause license
<i>Ultima versione rilasciata</i>	4.1.0

UniversalViewer

Vd. Mirador.

<i>Sito web</i>	https://universalviewer.io/
<i>Anno di realizzazione</i>	2012
<i>Descrizione</i>	The Universal Viewer is a community-developed open source project on a mission to help you share your content with the world
<i>Creatore/i</i>	Created by Digirati, currently developed by a Steering Grop by Mnemo-scene
<i>Formati di input</i>	IIIF Manifest
<i>Tecnologie utilizzate per lo sviluppo</i>	TypeScript, JavaScript, Less, HTML, CSS
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
Repository del codice sorgente	https://github.com/universalviewer/universalviewer
<i>Licenza</i>	MIT
<i>Ultima versione rilasciata</i>	3.1.1

VisColl

VisColl è uno strumento che consente all'utente di modellare i fogli di un codice o di un libro manoscritto e come è stato fascicolato. Sulla base del modello, che è di fatto un documento XML, l'utente può derivare diverse visualizzazioni grafiche da integrare nelle proprie edizioni digitali o in cataloghi e archivi digitali. Per rendere più *user-friendly* il lavoro di scrittura del modello e di produzione delle visualizzazioni, il progetto ha realizzato rispettivamente l'*editor* apposito VCEditor⁴⁷ e l'applicativo web VisCodex.⁴⁸

<i>Sito web</i>	https://viscoll.org/
<i>Anno di realizzazione</i>	2016
<i>Descrizione</i>	VisColl is a system for building models of the physical collation of manuscripts, and then visualizing them in various ways. The core of VisColl is the data model, which defines the structure of individual manuscripts and which can be built and visualized by different tools.
<i>Creatore/i</i>	Dot Porter, Alberto Campagnolo
<i>Ente/i partner</i>	University of Toronto Libraries, Old Books New Science lab Schoenberg Institute for Manuscript Studies (SIMS)
<i>Formati di input</i>	XML
<i>Formati di output</i>	HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	XSLT, HTML, JavaScript, CSS, Shell
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Sì
<i>Repository del codice sorgente</i>	https://github.com/KislakCenter/VisColl/
<i>Licenza</i>	MIT
<i>Ultima versione rilasciata</i>	2.1.0
<i>Pubblicazioni</i>	Porter, Dot, Alberto Campagnolo, and Erin Connelly. 2017. 'VisColl: A New Collation Tool for Manuscript Studies'. In <i>Kodikologie und Paläographie im Digitalen Zeitalter 4 – Codicology and Palaeography in the Digital Age 4</i> , edited by Hannah Busch, Franz Fischer, and Patrick Sahle, pp. 81–100. Norderstedt: Books on Demand. http://nbn-resolving.de/urn:nbn:de:hbz:38-77742 .

⁴⁷ <https://viscoll.org/help/>.

⁴⁸ <https://viscodex.library.utoronto.ca/>.

Altri formati

LIFT (Linked data from TEI)

Per classificare questo strumento secondo la tassonomia TaDiRAH, oltre alle attività indicate a livello generico di strumento di visualizzazione, si può utilizzare anche “linked open data”: “linked data available under an open license”.⁴⁹

LIFT (Linked data from TEI)* è uno strumento, ideato da Francesca Giovannetti (Università di Bologna) nel 2018, basato sul linguaggio di programmazione Python, che permette di trasformare un’edizione in XML/TEI in un *knowledge graph*,⁵⁰ pronto per la pubblicazione sul web come *linked open data*.

<i>Sito web</i>	https://projects.dharc.unibo.it/lift/
<i>Anno di realizzazione</i>	2021
<i>Descrizione</i>	LIFT is a Python-based tool for transforming your TEI XML edition into a knowledge graph, ready for publication as linked open data on the web. LIFT comes with a thorough documentation, designed to help you understand and reuse the methodology and technology behind the tool.
<i>Creatore/i</i>	Francesca Giovannetti
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	RDF
<i>Tecnologie utilizzate per lo sviluppo</i>	Python, Jupyter Notebook, HTML
<i>Consente il lavoro collaborativo</i>	No
Open source	Sì
<i>Repository del codice sorgente</i>	https://github.com/fgiovannetti/lift
<i>Licenza</i>	Gratuita

⁴⁹ <https://vocabs.dariah.eu/tadirah/en/page/linkedOpenData>.

⁵⁰ Un *knowledge graph*, noto anche come ‘rete semantica’, rappresenta una rete di enti del mondo reale - oggetti, eventi, persone, concetti, ecc. - e illustra i loro rapporti reciproci. Si tratta, in altre parole, di un altro modo di organizzare la conoscenza, strutturando i dati partendo dalle loro relazioni.

Strumenti misti

Per categorizzare gli strumenti misti secondo la tassonomia TaDiRAH si possono utilizzare le stesse attività indicate per gli strumenti di produzione e di visualizzazione.

Classical Text Editor

Classical Text Editor è uno strumento che è stato largamente utilizzato nel campo della filologia *tout court*. Si tratta di un programma di videoscrittura ideato appositamente per realizzare edizioni critiche digitali, facilitando, quindi, la creazione di apparati critici articolati, l'affiancamento di immagini al testo o di più testi tra loro.

<i>Sito web</i>	https://cte.oeaw.ac.at/
<i>Anno di realizzazione</i>	1997
<i>Descrizione</i>	The word-processor for critical editions, commentaries and parallel texts. Any number of notes and apparatus - bidirectional text - Open-Type - sigla. The Classical Text Editor was designed to enable scholars working on a critical edition or on a text with commentary or translation to prepare a camera-ready copy or electronic publication without bothering much about making up and page proofs. Its features, formed in continuous discussion with editors using the program, meet the practical needs of the scholar concerning text constitution, entries to different apparatus and updating them when the text has been changed, as well as creating and redefining sigla. The possibility to search for and visualise manuscript constellations may be of considerable help in detecting affiliations between sources. It is the primary purpose of the Classical Text Editor to do the automatable work which consumes so much time and energy, and let the scholar concentrate on scientific issues.
<i>Creatore/i</i>	Austrian Academy of Sciences
<i>Ente/i partner</i>	Stefan Hagel
<i>Formati di input</i>	XML-TEI, Microsoft Word (.doc, Rich Text Format (.rtf), Plain Text (.txt)
<i>Formati di output</i>	PDF, XML-TEI, HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	Borland/GodeGear Delphi 10.2 (25.0)
<i>Utilizzo</i>	Desktop

<i>Consente il lavoro collaborativo</i>	No
Open source	No
<i>Licenza</i>	A pagamento
<i>Ultima versione rilasciata</i>	10.4

Ediarum

Ediarum è un insieme di componenti software che possono essere integrati nell'*editor* XML Oxygen, per poter editare documenti XML/TEI mediante un'interfaccia simile a quella dei *word processor*, creare un apparato critico e preparare l'edizione per la pubblicazione a stampa esportando un file PDF oppure per la pubblicazione in rete, realizzando un sito web basato su eXist-db, REST, XQuery e XSLT.

<i>Sito web</i>	https://www.ediarum.org/
<i>Anno di realizzazione</i>	2012
<i>Descrizione</i>	Die digitale Arbeitsumgebung ediarum ist eine aus mehreren Softwarekomponenten bestehende Lösung, die es Wissenschaftler*innen erlaubt, Transkriptionen von Manuskripten und Drucken in TEI-konformem XML zu bearbeiten, mit einem Text- und Sachapparat sowie Registern zu versehen und in Web und Druck zu veröffentlichen.
<i>Creatore/i</i>	Lead developers: Stefan Dumont, Martin Fechner, and Sascha Grabsch
<i>Ente/i partner</i>	TELOTA (The Electronic Life Of The Academy) initiative at the BBAW in Berlin
<i>Formati di input</i>	XML-TEI
<i>Formati di output</i>	PDF, HTML
<i>Tecnologie utilizzate per lo sviluppo</i>	eXist-db, Oxygen XML Author, REST, Xquery, XSLT, ConTeXt
<i>Utilizzo</i>	Extension for Oxygen and eXist-db
<i>Consente il lavoro collaborativo</i>	Sì
Open source	Sì
<i>Repository del codice sorgente</i>	https://github.com/ediarum
<i>Licenza</i>	GPL-3.0
<i>Edizioni Critiche Digitali di esempio</i>	Schleiermacher Digital , Ifflands Archiv , Edition Humboldt Digital

eLaborate

Nel 2003 è stato ideato l'ambiente di lavoro online eLaborate dall'istituto di ricerca Huygens. Gli utenti possono caricare in eLaborate le scansioni dei documenti, trascrivere e annotare i testi e pubblicare i risultati online come edizioni liberamente accessibili da chiunque sul web. Il codice sorgente dell'applicativo è disponibile in *open source* su GitHub. La quarta versione del software è stata pubblicata a marzo del 2014. Nonostante il software non sia più sviluppato, sembrerebbe essere ancora mantenuto.

<i>Sito web</i>	https://elaborate.huygens.knaw.nl/
<i>Anno di realizzazione</i>	2003
<i>Descrizione</i>	eLaborate is an online work environment in which scholars can upload scans, transcribe and annotate text, and publish the results as on online text edition which is freely available to all users. Short information about and a link to already published editions is presented on the page Editions under Published. Information about editions currently being prepared is posted on the page Ongoing projects.
<i>Creatore/i</i>	Huygens ING
<i>Ente/i partner</i>	Huygens Institute for the History of the Netherlands of the Royal Netherlands Academy of Arts and Sciences
<i>Tecnologie utilizzate per lo sviluppo</i>	TypeScript, Java
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	No
<i>Open source</i>	Si
<i>Repository del codice sorgente</i>	https://github.com/HuygensING
<i>Licenza</i>	GLP-3.0
<i>Ultima versione rilasciata</i>	4.0
<i>Edizioni Critiche Digitali di esempio</i>	Bartholomeus Engelsman, Van den proprieteyten der dinghen

TextGrid

TextGrid è un applicativo *open source* basato su exist-db per l'*editing* e la pubblicazione di risorse basate su TEI. È suddiviso in due parti: TextGrid Lab, per l'*editing* collaborativo o individuale sdi testi e TextGrid Repository, che funge da database e da piattaforma per la condivisione e la pubblicazione.

<i>Sito web</i>	https://textgrid.de/
-----------------	---

<i>Anno di realizzazione</i>	2006
<i>Descrizione</i>	TextGrid is a virtual research environment for the humanities that is optimised for working with TEI-coded resources and covers the entire research process up to publication. The service consists of the TextGrid Lab, a flexible application for individual and collective work on text resources, and the TextGrid Repository, a resource database and platform for exchange and publication.
<i>Creatore/i</i>	TextGrid Research Association
<i>Ente/i partner</i>	TextGrid was a project of ten partners, funded by the German Federal Ministry of Education and Research (BMBF) for the period from June 2012 to May 2015 (reference number: 01UG1203A). Since 2016, TextGrid is part of the DARIAH-DE Research Infrastructure.
<i>Formati di input</i>	XML-TEI, OAI-PMH, IIIF Manifest
<i>Tecnologie utilizzate per lo sviluppo</i>	eXist-db
<i>Consente il lavoro collaborativo</i>	No
Open source	Si
<i>Ultima versione rilasciata</i>	3.3.0
<i>Pubblicazioni</i>	Stefan E. Funk. “Elektronisches Publizieren von Digitalen Forschungsdaten am Beispiel des TextGrid Repositorys – Umsetzung von Digitalen Publikationsworkflows für die eHumanities.” Masterarbeit, Köln, 2018, doi:10.20375/0000-000B-D269-2.

Textual Communities

TextualCommunities è un ambiente online per la trascrizione collaborativa e la collazione automatica di testi, che permette anche di usare i materiali editi come base per costruire un sito web mediante un’API. L’utente può caricare le riproduzioni fotografiche dei manoscritti e trascriverli in formato XML/TEI attraverso un *editor* dedicato. Poi i diversi testi trascritti possono essere collazionati con una funzionalità interna a Textual Communities. Tra gli autori di Textual Communities vi è Peter Robinson. Il sito del software, infatti, ospita tra le “communities” di esempio una parte delle *Canterbury Tales* edite da Robinson stesso.

<i>Sito web</i>	https://textualcommunities.org/app/
<i>Anno di realizzazione</i>	2018
<i>Descrizione</i>	Gather materials Bring together images of manuscripts and other materials. Name and organize images. Display zoomable images page by page beside transcriptions

	Collaborate Invite people as co-leaders or transcribers. Assign transcription and other tasks. Monitor and check transcription and other tasks. Control who can see what, who can work on what.
	Edit View and edit transcriptions with full support for all TEI/XML encoding. Preview transcriptions without encoding. Transcribe page by page. Collate section by section, with full facilities for adjustment of the collation output.
	Publish and Share Build your own website, drawing all materials from the Textual Communities database. Allow others unlimited access to your editorial materials. Combine freely with other materials through an api.
<i>Creatore/i</i>	Peter Robinson
<i>Ente/i partner</i>	Canada Foundation for Innovation, the Social Sciences and Humanities Research Council of Canada, and the University of Saskatchewan
<i>Formati di input</i>	XML-TEI, IIIF Manifest, JPEG, XML, zip
<i>Formati di output</i>	XML-TEI
<i>Utilizzo</i>	Online
<i>Consente il lavoro collaborativo</i>	Sì
Open source	No
<i>Licenza</i>	Gratuita

TILE

TILE (The Text-Image Linking Environment) è un applicativo *web based* per la creazione e l'editing di edizioni "image-based" (edizioni che presentano le riproduzioni fotografiche dei testimoni, insieme a delle trascrizioni o al testo critico) e di archivi testuali. TILE contiene una funzionalità per il riconoscimento semi-automatico delle righe all'interno di un manoscritto, che marca le aree di testo nell'immagine. L'architettura a plugin dello strumento consente di espanderne le funzionalità. La prima e unica versione del software è stata pubblicata nel 2011, per cui quasi sicuramente il software non è più mantenuto.

<i>Sito web</i>	https://archive.mith.umd.edu/tile/index.html
<i>Anno di realizzazione</i>	2011
<i>Descrizione</i>	The Text-Image Linking Environment (TILE) is a web-based tool for creating and editing image-based electronic editions and digital archives of humanities texts. TILE features tools for importing and exporting transcript lines and images of text, an image markup tool, a semi-automated line recognizer that tags

regions of text within an image, and plugin architecture to extend the functionality of the software.

<i>Creatore/i</i>	Doug Reside, Dave Lester, Dot Porter and John Walsh
<i>Ente/i partner</i>	Maryland Institute for Technology in the Humanities (MITH), Indiana University Bloomington, The MAPAS Project, University of Oregon, The Homer Multitext Project, Harvard University
<i>Formati di input</i>	TEI P5 XML, JSON, XML
<i>Formati di output</i>	TEI, JSON, XML, HTML, TXT
<i>Utilizzo</i>	Web-based
<i>Consente il lavoro collaborativo</i>	No
Open source	No
<i>Licenza</i>	Gratuita
<i>Ultima versione rilasciata</i>	1.0

TUSTEP

TUSTEP fu ideato nel 1966 da Wilhelm Ott e Kuno Schälkle e consente di effettuare diverse operazioni, tra cui: trascrivere, collazionare, costituire apparati critici, generare indici e concordanze. Da un punto di vista tecnico, TUSTEP è costituito da diversi moduli, ciascuno ideato per eseguire un'operazione specifica, che l'utente può combinare a piacere in modo da creare flussi di lavoro personalizzati che rispondano alle proprie esigenze. Per utilizzare il programma è necessario lanciare dei comandi (in lingua tedesca o inglese) attraverso un'interfaccia a riga di comando non molto *user-friendly*. TUSTEP offre la possibilità di importare ed esportare testo in qualsiasi punto del flusso di lavoro. Gli output possono essere a stampa (PostScript) o per la pubblicazione elettronica (HTML, XML). Una nuova versione viene rilasciata circa ogni anno, l'ultima versione stabile è stata rilasciata nel 2018. Di recente è stata sviluppata una versione del software basata su XML chiamata TXSTEP,⁵¹ che però è ancora in fase sperimentale.

<i>Sito web</i>	https://www.tustep.uni-tuebingen.de/tustep_eng.html
<i>Anno di realizzazione</i>	1966
<i>Descrizione</i>	TUSTEP is a professional toolbox for scholarly processing textual data (including those in non-latin scripts) with a strong focus on humanities applications. It contains modules for all stages of scholarly text data processing, starting from data capture and including information retrieval, text collation, text analysis, sorting and ordering, rule-based text manipulation, and

⁵¹ <http://www.txstep.de/>.

output in electronic or conventional form (including typesetting in professional quality).

<i>Creatore/i</i>	Wilhelm Ott and Kuno Schälkle
<i>Ente/i partner</i>	Computing Center of the University of Tübingen
<i>Formati di input</i>	XML-TEI, Plain Text (.txt)
<i>Formati di output</i>	XML-TEI, TXT, PostScript, HTML
<i>Consente il lavoro collaborativo</i>	No
Open source	No
<i>Licenza</i>	BSD
<i>Ultima versione rilasciata</i>	TUSTEP 2023
<i>Pubblicazioni</i>	Griesinger, Christian. 2020. «Review of the Tübinger System von Textverarbeitungs-Programmen (TUSTEP)». RIDE (blog). gennaio 2020. https://doi.org/10.18716/ride.a.11.2 .

Appendice B

Definizioni inserite nell'ontologia

Uno degli aspetti della presente ricerca più rilevanti da un punto di vista scientifico è la definizione delle entità e delle loro reciproche relazioni modellate all'interno dell'ontologia. Tali definizioni sono la base su cui fondare un *ubiquitous language* (Evans 2004) tra i filologi, gli “esperti del dominio”, e gli sviluppatori informatici, che aiuti i secondi a realizzare strumenti informatici capaci di soddisfare le esigenze dei primi.

In questa appendice si presentano in modo più leggibile le etichette e le descrizioni complete fornite nell'ontologia, che sono state elaborate da chi scrive, in modo da consentire una verifica più agevole della validità e dell'efficacia delle descrizioni proposte. Le descrizioni degli elementi di ontologie esistenti sono state omesse dalla presente appendice, in quanto già presentate e discusse all'interno del Capitolo 3. Eventualmente, possono essere consultate anche all'interno del codice dell'ontologia all'appendice C.

Per ogni classe, proprietà e *named individual* sono state fornite:

- due etichette, una in inglese e una in italiano, inserite nell'ontologia grazie all'elemento `<rdfs:label>`;
- una descrizione in inglese e la traduzione in italiano, inserite nell'ontologia con l'elemento `<rdfs:comment>`.

Le descrizioni sono state scritte in modo tale da risultare chiare, ma concise.

Per alcune classi e *named individual* si presentano anche le definizioni riprese dai manuali utilizzati come riferimento teorico (vd. 3.3) e i riferimenti alle fonti bibliografiche utilizzate.

L'appendice è suddivisa in quattro parti, che ricalcano la struttura del modello concettuale: i concetti basilari per inquadrare il significato di edizione critica e di edizione scientifica in generale; i concetti che caratterizzano la fase di *restitutio textus*; i componenti comuni a diverse tipologie di edizione scientifica; i componenti specifici di un'edizione critica (o ricostruttiva), inquadrati secondo

l'approccio neo-lachmanniano. Ciascuna parte è suddivisa a sua volta in più sezioni: la prima dedicata alle classi, una seconda dedicata ai *named individual* e l'ultima per le proprietà. All'interno di ogni sezione gli elementi sono organizzati in ordine alfabetico, in base all'etichetta in inglese. L'etichetta in inglese è seguita subito sotto dalla traduzione in italiano. A seguire ci sono le descrizioni in inglese e in italiano, presentate una accanto all'altra per favorire il confronto, ed eventualmente le definizioni fornite da altri studiosi con le rispettive sigle bibliografiche. L'etichetta delle proprietà inverse è presentata tra parentesi tonde accanto alla proprietà di cui sono l'inverso.

Concetti basilari

Classi

Critical Edition

Edizione critica

A type of [scholarly edition](#), created by applying the principles of textual criticism, with the aim of reconstructing the original form or more original forms of the text of a work.

Un tipo di edizione scientifica, realizzata applicando i principi della critica testuale, allo scopo di ricostruire la forma originaria o le forme originarie del testo di un'opera.

“La conclusione naturale di uno studio di critica testuale è la realizzazione di un'edizione critica. Si tratta di un'edizione ‘scientifica’ dell'opera, tale cioè che possa essere utilizzata dal lettore come testo ‘ufficiale’ e affidabile, e nella quale si affrontino i problemi posti da quello specifico testo in ordine al suo stato di conservazione e, ove necessario, alla sua ricostruzione. Essa può consistere nella riproduzione dell'originale, se è conservato; o in un'ipotesi di ricostruzione dell'originale, se esso non è conservato; o ancora nella pubblicazione comparativa di testi diversi, ognuno dei quali gode della qualifica di originale o è comunque importante ad illustrare il processo di sviluppo testuale.”

Chiesa 2002, p. 24

“A critical edition is an edition in which the text has been constituted on the basis of more than one source according to the genealogical principle. As it uses more than one source, in this respect it is an eclectic edition. What sets the critical edition apart is that it is based on a strict recension of the manuscript sources and the attempt to edit the archetypal text (possibly with some changes where the archetype is clearly faulty), and it is thus closely associated with the Lachmannian tradition of textual editing. Editors

Voce “edition, critical” del PLS

who do not agree with this reconstructive tradition, usually refer to their editions in other terms than critical.”

Voce “edition (critical)” del LexiconSE.¹

Editor

Editore

Person that created a scholarly edition. The editor must possess philological skills and textual criticism skills. The identity of the editor must be known, in order to make their work citable.

Persona che realizza un’edizione scientifica. L’editore deve possedere competenze di filologia e di critica testuale. L’identità dell’editore deve essere nota, per rendere il suo lavoro citabile.

Voce “editor (scholarly)” del LexiconSE.²

Voce “editor” del LexiconSE.³

Editors Group

Gruppo di editori

A group of [editors](#) that collaboratively create a scholarly edition.

Un gruppo di editori che ha realizzato collaborativamente un’edizione scientifica.

Reader

Lettore

Person that reads the scholarly edition.

Persona che legge l’edizione scientifica.

Note: the term “reads” was chosen to sum up the different ways in which a critical edition can be used.

Nota: con un generico “legge” si riassumono le diverse modalità in cui un’edizione critica può essere fruita.

¹ Dato che nel LexiconSE per ogni voce vengono riportate due o più definizioni, in questa appendice si citano solo quelle che sono state prese direttamente a riferimento per elaborare la definizione dell’elemento dell’ontologia. Per le altre definizioni si rimanda alla rispettiva pagina online. <https://lexiconse.uantwerpen.be/lexicon/editionCritical.html>.

² <https://lexiconse.uantwerpen.be/lexicon/editorScholarly.html>.

³ <https://lexiconse.uantwerpen.be/lexicon/editor.html>.

Scholarly Edition

Edizione scientifica

The content of a publication that makes the text of a literary work or work of historical-cultural interest, mostly of the past, otherwise inaccessible, accessible to the public. The edition is called “scholarly” as it is created by a scholar or a group of scholars, applying shared methodologies in their own philological field.

Il contenuto di una pubblicazione che rende accessibile al pubblico il testo di un’opera letteraria o di interesse storico-culturale, perlopiù del passato, altrimenti inaccessibile. L’edizione è detta “scientifica” in quanto è realizzata da uno studioso o da un gruppo di studiosi, applicando delle metodologie condivise nel proprio ambito filologico.

“l’edizione è l’atto con il quale si pubblica un testo del passato, cioè lo si rende di nuovo pubblico, accessibile alla lettura.” Leonardi 2022, p. 132

“Scholarly editing [...] is used to describe any sort of edition prepared by a person claiming to be a scholar. [...] The term should be reserved for editorial efforts designed to make available for scholarly use works not ordinarily available or available only in corrupt or inadequate forms.” Shillingsburg 1996

Voce “edition (scholarly)” nel LexiconSE.⁴

Scholarly Edition Publication

Pubblicazione dell’edizione scientifica

The publication of a scholarly edition, which comprises both the content (the scholarly edition itself), its presentation and the material support.

La pubblicazione dell’edizione scientifica, che comprende non solo il suo contenuto (l’edizione scientifica in sé) ma anche il modo in cui si presenta e il suo supporto fisico.

Textual Tradition

Tradizione testuale

Set of witnesses that preserve a work.

Insieme dei testimoni che tramandano un’opera.

“insieme dei testimoni di un’opera; o anche insieme dei fenomeni che si realizzano nella sua trasmissione nel tempo” Leonardi 2022, p. 197

⁴ <https://lexiconse.uantwerpen.be/lexicon/editionScholarly.html>.

“The tradition of a work is made up of the texts in which this work has been transmitted, as attested by its witnesses. The tradition is said to be ‘direct’ if the witnesses are ‘direct’, i.e. if they are assumed to be copies (manuscripts) or prints (*editio princeps*, *incunabula*) of the work itself. Part of the tradition can be ‘indirect’, consisting of indirect witnesses, i.e. excerpts or citations from the work in other works or translations of the work into another language. [...]”⁵

Voce “tradition”
del PLS

Voce “textual transmission” del LexiconSE.⁶

Witness

Testimone

Copy of the text of a work.

Copia del testo di un’opera.

“copia di un testo, manoscritta o a stampa”

Leonardi 2022,
p. 197

“A witness is an existing instance of a text (see also copy). The witnesses form the evidence for the textual tradition of a given text, and are the basis from which stemmata are constructed. Witnesses may be carried in any physical medium; normally these will be manuscripts or printed books, but can also include oral recordings and the like in the case of folk tales and other oral traditions. A single manuscript is often witness to multiple texts, as manuscripts generally carry more than one text”⁷

Voce “witness”
del PLS

Voce “witness” del LexiconSE.⁸

Witness Carrier

Supporto del testimone

The physical support of a witness, usually a manuscript or a printed book.

Il supporto materiale di un testimone, solitamente un manoscritto o un libro a stampa.

⁵ <https://wiki.helsinki.fi/display/stemmatology/Tradition>.

⁶ <https://lexiconse.uantwerpen.be/lexicon/textualTransmission.html>.

⁷ <https://wiki.helsinki.fi/display/stemmatology/Witness>.

⁸ <https://lexiconse.uantwerpen.be/lexicon/witness.html>.

Proprietà

created (was created by)

ha creato (è stata creata da)

This property associates an instance of [Editor](#) or [EditorsGroup](#) with an instance of [Scholarly Edition](#) they created.

Questa proprietà associa un'istanza di Editor o di EditorsGroup a un'istanza di Scholarly Edition da loro creata.

has part (is part of)

ha una parte (è parte di)

This property associates an instance of [Textual Transmission](#) to an instance of [Witness](#) and represents the relationship between a witness and the textual tradition to which it belongs.

Questa proprietà associa un'istanza di Textual Transmission a un'istanza di Witness e rappresenta il rapporto tra un testimone e la tradizione testuale cui appartiene.

is edited in (is edition of)

è edito in (è l'edizione di)

This property associates an instance of F1 Work with an instance of [Scholarly Edition](#). A scholarly edition is one of the realizations of a work.

Questa proprietà associa un'istanza di F1 Work a un'istanza di Scholarly Edition. Un'edizione scientifica è una delle realizzazioni di un'opera.

preserves (is preserved in)

conserva (è conservato in)

This property associates a physical support, an instance of [Witness Carrier](#), with the witness it preserves, an instance of [Witness](#). Several witnesses can be preserved within the same physical carrier.

Questa proprietà associa un supporto fisico, un'istanza di Witness Carrier, al testimone che conserva al proprio interno, un'istanza di Witness. All'interno del medesimo supporto fisico possono essere conservati più testimoni.

publishes (is published in)

pubblica (è pubblicata in)

This property associates an instance of [Scholarly Edition](#) with the instance of [Scholarly Edition Publication](#) within which it is published. The same instance of Scholarly Edition can be published in different publications. At the same time, a publication can contain several scholarly editions.

Questa proprietà associa un'istanza di Scholarly Edition all'istanza di Scholarly Edition Publication all'interno della quale è pubblicata. La medesima istanza di Scholarly Edition può essere pubblicata in pubblicazioni diverse. Al contempo, una pubblicazione può contenere più edizioni scientifiche.

reads (is read by)

legge (è letta da)

This property associates an instance of [Reader](#) with the instance of [Scholarly Edition Publication](#) they read.

Questa proprietà associa un'istanza di Reader all'istanza di Scholarly Edition Publication che legge.

transmits (is transmitted in)

tramanda (è tramandata da)

This property associates an instance of Witness with the instance of F1 Work it transmits. An instance of F1 Work is handed down by at least one witness and a variable number of witnesses can hand down the same work.

Questa proprietà associa un'istanza di Witness all'istanza di F1 Work che tramanda. Un'istanza di F1 Work è tramandata da almeno un testimone e un numero variabile di testimoni può tramandare la stessa opera.

Restitutio textus: rapporto tra il testo critico e la tradizione

Classi

Base Witness

Testimone di base

Witness used as a basis to establish the critical text.

Testimone adottato come base per l'allestimento del testo critico.

“manoscritto il cui testo è adottato come base per l'edizione”

Leonardi 2022,
p. 194

“A manuscript on which an edition is based (Greg 1959: 19), usually the supposedly best manuscript. In English also called the base text (but there may be a subtle difference, cf. collation), sometimes the German expression *Leithandschrift* is also used in English. The details of the usage vary to some point (cf. Sahle 2013: 171). Altick and Wright define the copy text as (1971: 134): “The text of a work, in print or in manuscript, from which a new edition is set. More narrowly, the edition or manuscript which is closest to the author’s intention and which is used as the basis for a critical edition”. Greg saw that a copy text should only be used for accidental readings (cf. variant reading) whereas the Lachmannian method should be used for substantial ones.”⁹

Voce “copy text” del PLS

Critical Intervention

Intervento critico

During the preparation of the critical text, it is the interpretative act through which the editor establishes or reconstructs the portion of text to be included in the critical text and the readings to be presented in the critical apparatus, starting from the readings present in the textual transmission.

In fase di allestimento del testo critico, è l’atto interpretativo mediante il quale l’editore stabilisce o ricostruisce la porzione di testo da accogliere nel testo critico e le lezioni da presentare nell’apparato critico, partendo dalle lezioni presenti nella tradizione testuale.

Voce “editing (critical)” del LexiconSE.¹⁰

Critical Intervention Criterion

Criterio dell’intervento critico

The criterion or more broadly the reason why the editor has deemed a critical intervention necessary.

Il criterio o più in generale la motivazione che ad avviso dell’editore hanno reso necessario l’intervento critico.

Critical Intervention Type

Tipo di intervento critico

The type of critical intervention made by the editor, for example: the correction of an evident error, the

La tipologia dell’intervento critico effettuato dall’editore, per esempio: la correzione di un errore

⁹ <https://wiki.helsinki.fi/display/stemmatology/Copy+text>.

¹⁰ <https://lexiconse.uantwerpen.be/lexicon/editingCritical.html>.

selection of a variant (*selectio*), the conjecture (*divinatio*), etc.

evidente, la scelta di una variante (*selectio*), la congettura (*divinatio*), ecc.

Critical Text

Testo critico

A type of edited text presented in a critical edition and edited for the purpose of reconstructing the original form or one of the original forms of the text of a work.

Un tipo di testo edito presentato all'interno di un'edizione critica e allestito allo scopo di ricostruire la forma originaria o una delle forme originarie del testo di un'opera.

Voce "text (critical)" del LexiconSE.¹¹

Critical Text Passage

Passo del testo critico

A portion of variable size of the critical text, made up of characters, words or sentences.

Una porzione di dimensione variabile del testo critico, costituita da caratteri, parole o frasi.

External Critical Intervention Criterion

Criterio esterno dell'intervento critico

A criterion for carrying out the critical intervention based on the genealogy of the textual transmission.

Un criterio per effettuare l'intervento critico basato sulla genealogia della tradizione testuale.

"Criteri di valutazione delle varianti in base alla loro posizione nella genealogia della tradizione" Leonardi 2022, p. 190

Gap

Passo mancante

Missing passage within a witness. It may be due to a voluntary or involuntary omission on the part of the copyist, a mechanical gap, an erasure or the state of conservation of the witness.

Passo mancante all'interno di un testimone. Può essere dovuto a un'omissione volontaria o involontaria da parte del copista, a una lacuna meccanica, a

¹¹ <https://lexiconse.uantwerpen.be/lexicon/textCritical.html>.

una cancellatura o allo stato di conservazione del testimone.

“**lacuna** omissione di una parte di testo. La lacuna è detta meccanica quando dipende dalla perdita fisica di una parte del manoscritto” Leonardi 2022, p. 193

“A lacuna, literally in Latin a ‘hole, gap’, in a manuscript is a gap of missing text of varying length. ‘Lacuna’ is generally used with the physical object in mind (see material accidents), whereas ‘omission’ refers only to the text.” Voce “lacuna” del PLS

“An omission is any segment of text that a copyist does not reproduce in the copied text, but that is present in the exemplar. In the process of collating and editing the term omission is a relative one which only indicates that a segment of text which is present in the base-text is lacking in some witness(es), without making a judgement whether the omission is secondary or not. One may distinguish omissions intentionally introduced by the scribe from unintentional ones.” Voce “omission” del PLS

“(gap) indicates a point where material has been omitted in a transcription, whether for editorial reasons described in the TEI header, as part of sampling practice, or because the material is illegible, invisible, or inaudible.”¹² TEI P5 Guidelines

Voce “omission” del LexiconSE.¹³

Internal Critical Intervention Criterion

Criterio interno dell'intervento critico

A criterion for carrying out the critical intervention based on the intrinsic plausibility of the variants. | Un criterio per effettuare l'intervento critico basato sulla plausibilità intrinseca delle varianti.

“Criteri di valutazione delle varianti in base alla loro plausibilità intrinseca” Leonardi 2022, p. 191

Reading

Lezione

¹² <https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-gap.html>.

¹³ <https://lexiconse.uantwerpen.be/lexicon/omission.html>.

A sequence of characters, words, or phrases that can be read in a preserved or lost witness.	Una sequenza di caratteri, parole o frasi che può essere letta in un testimone conservato o andato perduto.
--	---

“ciò che si legge in un manoscritto”	Leonardi 2022, p. 194
--------------------------------------	-----------------------

“le forme presenti nei testimoni (tecnicamente chiamate lezioni, ‘ciò che si legge’).”	Chiesa 2002, p. 14
--	--------------------

“leggere 1. detto di scelta critica, adottare (come testo critico) 2. detto di testimone, >avere” ¹⁴	Glossario della terminologia filologica
---	---

“A reading is a short piece of text, often a single word or phrase, which typically varies between manuscripts, and for this reason will often be synonymous with a variant reading. The fact that a segment of text is called a reading may imply that it may be open to interpretation, i.e. that the word(s) in question can be read in more than one way.”	Voce “reading” del PLS
--	------------------------

Voce “reading (variant)” del LexiconSE.¹⁵

Reading In Apparatus

Lezione in apparato

Reading presented in a critical apparatus entry. A reading in the critical apparatus may correspond to two or more readings of different witnesses. The reading in the critical apparatus can be presented in a diplomatic (according to a particular witness) or interpretative form.	Lezione presentata in una voce dell’apparato critico. Una lezione nell’apparato critico può corrispondere in realtà a due o più lezioni di testimoni diversi. La lezione nell’apparato critico può essere riportata in veste diplomatica (secondo la lettera di un particolare testimone) o in veste interpretativa.
--	--

“A reading is a scholarly claim about or interpretation of a particular textual fragment contained in a witness. Multiple readings may be given for the same textual fragment.” ¹⁶	CAO
---	-----

“Copy of one or more words or phrases of an original or witness.” ¹⁷	SE
---	----

¹⁴ <http://www.filologiadautore.it/wp/glossario-della-terminologia-filologica/>.

¹⁵ <https://lexiconse.uantwerpen.be/lexicon/readingVariant.html>.

¹⁶ <https://w3id.org/cao/Reading>.

¹⁷ <https://github.com/nie-ine/e-editiones/blob/master/ontology/scholarly-editing.rdf>.

Reading In Apparatus Cause

Causa della lezione in apparato

The type of error made by the copyist during the act of copying or the factor that may have led the copyist to produce an erroneous or innovative reading.	Il tipo di errore commesso dal copista durante l'operazione di copia o il fattore che può aver indotto il copista a produrre una lezione erronea o innovativa.
--	--

“This class comprises reading causes. A reading cause is the cause for a variant reading (e.g. homeoteleuton, dittography, interpolation).”¹⁸ CAO

“@cause classifies the cause for the variant reading, according to any appropriate typology of possible origins”¹⁹ TEI P5 Guidelines

Reading In Apparatus Type

Tipo della lezione in apparato

Category that the editor assigns to a lesson to classify it and make their evaluation explicit.	Categoria che l'editore attribuisce a una lezione per classificarla ed esplicitare la propria valutazione.
---	--

“This class comprises reading types. A reading type is the category for a variant reading (e.g. omission, conjecture, addition).”²⁰ CAO

“@type classifies the reading according to some useful typology.”²¹ TEI P5 Guidelines

Surface Witness

Testimone di superficie

Witness used as a reference to establish the linguistic style of the critical text.	Testimone adottato come riferimento per stabilire la forma linguistica del testo critico.
---	---

“**manoscritto di superficie** manoscritto il cui testo è seguito nell'edizione per stabilire la superficie o forma linguistica del testo” Leonardi 2022, p. 194

¹⁸ <https://w3id.org/cao/ReadingCause>.

¹⁹ https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-att.textCritical.html#tei_att.cause.

²⁰ <https://w3id.org/cao/ReadingType>.

²¹ https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-att.textCritical.html#tei_att.type.

Unclear Passage

Luogo incerto del testo

Witness passage of variable size, which the editor cannot read and report with certainty due to how it appears within its material support.

Passo del testimone di dimensione variabile, che l'editore non può leggere e riportare con certezza per via di come si presenta all'interno del proprio supporto materiale.

“(unclear) contains a word, phrase, or passage which cannot be transcribed with certainty because it is illegible or inaudible in the source.”²²

TEI P5 Guidelines

Voce “apparatus (critical)” del LexiconSE.²³

Named Individual

Reading In Apparatus Type

Diversi *named individual* della classe Reading In Apparatus Type sono stati ripresi dalla CAO, tuttavia la definizione di cao:addition, cao:deletion e cao:conjecture non è stata ritenuta sufficientemente chiara e descrittiva. Per cui all'interno della presente ontologia sono state elaborate delle nuove definizioni.

addition

aggiunta

An addition is a text string that the copyist inserts into the text but is not present in the antigraph, or a text string omitted by the copyist and added later elsewhere in the witness.

Un'aggiunta è una stringa testuale che il copista inserisce nel testo ma non è presente nel modello oppure una stringa di testo omessa dal copista e aggiunta in un secondo momento in un altro punto del testimone.

“1) stringa testuale che il copista inserisce nel testo ma non è presente nel modello 2) Leonardi 2022 stringa di testo inizialmente omessa dal copista e aggiunta in un secondo momento in un altro punto del documento”

²² <https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-unclear.html>.

²³ <https://lexiconse.uantwerpen.be/lexicon/addition.html>.

“An addition is any segment of text that a copyist introduces into the copied text (one that is not present in the exemplar). In the process of collating and editing the term addition is a relative one which only indicates that a segment of text which is lacking in the base-text is present in some witness(es), without making a judgement whether the addition is secondary or not. Cf. its contrary omission, and types of errors.”²⁴

Voce “addition”
del PLS

Voce “addition” del LexiconSE.²⁵

characteristic reading

lezione caratteristica

Non erroneous reading common to several witnesses, which can confirm their common origin if this is demonstrated by conjunctive errors.	Lezione non erronea comune a più testimoni, che può confermare la loro comune origine se questa è dimostrata da errori congiuntivi.
---	---

“lezione non erronea comune a più manoscritti, che può confermare la loro comune origine se questa è dimostrata da errori congiuntivi” Leonardi 2022

conjecture

congettura

Quando l’editore offre una lezione congetturale, la lezione deve essere rappresentata come un’istanza di Critical Text Passage e/o di Base Reading In Apparatus, prodotta da un intervento critico avente associato come tipo di intervento critico “emendatio ope ingenii”. Quando l’editore presenta in apparato una lezione congetturata da un editore esterno all’edizione o da un copista, occorre un’istanza della classe Reading In Apparatus, avente associato come tipo “conjecture”.

Conjectural reading proposed by an editor external to the critical edition or by the copyist of a witness.	Lezione congetturale proposta da un editore esterno all’edizione critica oppure dal copista di un testimone.
--	--

“A conjecture is the introduction into the text of a reading without a basis in the witnesses available to the person doing the conjecture. One should differentiate between conjectures by scribes (see scribal conjecture) as opposed to conjectures by modern editors (which are done by *divinatio*, cf. also *emendatio*)”.²⁶

Voce “con-
jecture” del PLS

²⁴ <https://wiki.helsinki.fi/display/stemmatology/Addition>.

²⁵ <https://lexiconse.uantwerpen.be/lexicon/apparatusCritical.html>.

²⁶ <https://wiki.helsinki.fi/display/stemmatology/Conjecture>.

conjunctive error

errore congiuntivo

Error present in two or more witnesses, which allows to deduce that they are copies of each other or copies of the same lost witness.	Errore presente in due o più testimoni, che consente di dedurre che essi sono copie l'uno dell'altro o copie di uno stesso testimone perduto.
---	---

“errore presente in due o più codici, che consente di dedurre che essi sono copie l'uno dell'altro o copie di uno stesso codice perduto” Leonardi 2022

“Conjunctive errors are errors that show that two witnesses can be grouped together against other witnesses. For an error to be conjunctive, it must be highly unlikely that it arises from polygenesis. Because these errors are used in the construction of the stemma, they must be significant”.²⁷ Voce “conjunctive error” del PLS

Voce “error (conjunctive)” del LexiconSE.²⁸

deletion

cancellatura

String of text that the copyist somehow designated to be removed.	Stringa di testo che il copista ha in qualche modo designato come da rimuovere.
---	---

“stringa di testo che il copista ha in qualche modo designato come da rimuovere” Leonardi 2022

Voce “cancellation” del LexiconSE.²⁹

Voce “elimination” del LexiconSE.³⁰

deliberate omission

omissione volontaria

Text string deliberately omitted by the copyist in the act of copying the text.	Stringa di testo volutamente omessa dal copista nell'atto di copiare il testo.
---	--

²⁷ <https://wiki.helsinki.fi/display/stemmatology/Error%2C+conjunctive>.

²⁸ <https://lexiconse.uantwerpen.be/lexicon/errorConjunctive.html>.

²⁹ <https://lexiconse.uantwerpen.be/lexicon/cancellation.html>.

³⁰ <https://lexiconse.uantwerpen.be/lexicon/elimination.html>.

diffraction

diffrazione

The phenomenon whereby tradition presents multiple alternative and distinct variants for a single textual passage, resulting from the scribes' reaction to a difficulty in the original text.

Fenomeno per cui la tradizione presenta più varianti alternative e diverse tra loro per un singolo luogo testuale, in seguito alla reazione dei copisti di fronte a una difficoltà del testo di partenza.

“fenomeno per cui la tradizione presenta più varianti alternative per un singolo luogo testuale, in seguito alla reazione dei copisti di fronte a una difficoltà del testo di partenza” Leonardi 2022

“The term ‘diffraction’ (also ‘multiple innovation’) refers to the substitution of the original [reading], when particularly difficult, infrequent or rare, with several innovative (but trivial) readings or attempts of clarification by the scribes.” Voce “diffraction” del PLS

Voce “diffraction” del LexiconSE.³¹

diffraction in absentia

diffrazione in assenza (della lezione ricevibile)

The same as diffraction, but with the peculiarity that none of the variants attested within the textual transmission is considered to be the origin of the phenomenon by the editor.

Lo stesso che diffrazione, ma con la particolarità che nessuna delle varianti attestate dalla tradizione è considerata all'origine del fenomeno dall'editore.

“Quando la tradizione è indubbiamente corrotta nel suo insieme, ma il guasto non è omogeneo, perché i diversi testimoni hanno errori differenti, si può supporre che le varie forme presenti nella tradizione abbiano surrogato una concorrente “più difficile”, perpetuata fino a un gradino superiore a quello dei testimoni superstiti. Il Maas parla di varianti “che si possono interpretare come errori che sono stati determinati dalla medesima lezione dell'archetipo”. Questo fenomeno, secondo una proposta del Contini, si può chiamare diffrazione, e più esattamente diffrazione in assenza (della lezione ricevibile)” Brambilla Ageno 1984

diffraction in presentia

diffrazione in presenza (della lezione ricevibile)

³¹ <https://lexiconse.uantwerpen.be/lexicon/diffraction.html>.

The same as diffraction, but with the peculiarity that one of the variants attested within the textual transmission is considered to be the origin of the phenomenon by the editor.

Lo stesso che diffrazione, ma con la particolarità che una delle varianti attestate dalla tradizione è considerata all'origine del fenomeno dall'editore.

“Accade che non si debba scegliere fra due sole varianti, ma che la tradizione presenti una serie di lezioni divergenti. Questo fenomeno, che si chiama diffrazione, postula una difficoltà (o un errore) nell'esemplare comune al gruppo di manoscritti in cui si verifica. Ci limitiamo qui ad esaminare il caso della diffrazione in presenza (della lezione ricevibile): a volte l'esame attento delle varie lezioni offerte dalla tradizione dimostra che una di esse è ricevibile, ed è insieme tale da spiegare tutte le altre, in quanto contiene una singolarità o una difficoltà che può avere indotto i singoli copisti a tentativi di correzione o a banalizzazioni.”

Leonardi 2022

easily corrigible error

errore facilmente correggibile

Obviously erroneous reading and therefore easily corrigible.

Lezione evidentemente erronea e quindi facilmente correggibile.

error

errore

Reading that is considered not to be in the original form of the text as it is incongruous with respect to the grammatical, semantic, metric, narrative coherence of the text.

Lezione che si ritiene non ascrivibile alla forma originaria del testo in quanto incongrua rispetto alla coerenza grammaticale, semantica, metrica, narrativa del testo.

“lezione che si ritiene non attribuibile all'autore in quanto incongrua rispetto alla coerenza grammaticale, semantica, metrica, narrativa del testo”

Leonardi 2022

indifferent variant

variante adiàfora

Variant for which according to the editor it is not possible to express a preference with respect to the others equally attributable to the original form of the text.

Variante per cui secondo l'editore non è possibile esprimere una preferenza rispetto alle altre altrettanto attribuibili alla forma originaria del testo.

“le varianti si definiscono adiafore quando non si trovano argomenti che rendono l’una preferibile all’altra” Leonardi 2022

Voce “variant (indifferent)” del LexiconSE.³²

innovation

innovazione

Reading that alters the text of the antigraph and therefore cannot be attributed to the original form of the text.	Lezione che modifica il testo dell’antigrafo e quindi non attribuibile alla forma originaria del testo.
--	---

“lezione che modifica il testo dell’antigrafo”

Leonardi 2022

“In textual criticism an innovation is a change introduced at some point into a textual tradition. It is basically a positive expression for the Lachmannian notion of error, cf. also secondary reading. As it is more positive it will be used more for conscious and goal-driven interventions by scribes into texts (which can only be called errors by stretching the normal use of this word quite far).”	Voce “innovation” del PLS
---	---------------------------

Voce “innovation” del LexiconSE.³³

inter scribendum correction

correzione nel corso della scrittura

Error present in two or more witnesses, which allows to deduce that they are copies of each other or copies of the same lost witness.	Correzione di un errore da parte dello stesso copista, eseguita immediatamente, prima di continuare la scrittura del testo.
---	---

“correzione di un errore da parte dello stesso copista, eseguita immediatamente, prima di continuare la scrittura del testo” Leonardi 2022

Voce “error (conjunctive)” del LexiconSE.³⁴

lectio difficilior

lezione più difficile

³² <https://lexiconse.uantwerpen.be/lexicon/variantIndifferent.html>.

³³ <https://lexiconse.uantwerpen.be/lexicon/innovation.html>.

³⁴ <https://lexiconse.uantwerpen.be/lexicon/errorConjunctive.html>.

More complex or rare or refined reading compared to the alternative variants, and therefore probably original.	Lezione più complessa o rara o ricercata rispetto alle varianti alternative, e quindi verosimilmente originaria.
--	--

“lezione più complessa o rara o ricercata rispetto alle varianti alternative, e quindi verosimilmente originaria” Leonardi 2022

“A *lectio difficilior* is a more difficult reading and as such often a more original reading. When scribes met with difficult passages in the text, they often simplified the text in order to make it understandable to them. For this reason, the more difficult reading is often the better (older) one. The traditional rule therefore states that *lectio difficilior potior* ‘the more difficult reading is the stronger one’. Its contrary is the ‘easier reading’ or *lectio faciliior*, which is also known as banalisation or trivialisation of the text”.³⁵

Voce “lectio difficilior, lectio potior” del PLS

Voce “lectio difficilior” del LexiconSE.³⁶

lectio faciliior

lezione più facile (o banalizzazione)

Less complex or rare or refined reading compared to the alternative variants, and therefore probably not original.	Lezione meno complessa o rara o ricercata rispetto alle varianti alternative, e quindi non verosimilmente originaria.
--	---

“nell’atto di copia, innovazione che rende il testo più banale, più facile” Leonardi 2022

lectio singularis

lezione isolata

Reading transmitted by a single witness or a single sub-family of witnesses who are isolated in the stemma with respect to the tradition as a whole.	Lezione trasmessa da un solo testimone o una sola sotto-famiglia di testimoni che risultano isolati nello stemma rispetto all’insieme della tradizione.
--	---

“lezione trasmessa da un solo testimone o una sola sotto-famiglia che risultano isolati nello stemma rispetto all’insieme della tradizione” Leonardi 2022

“A reading (*lectio*) transmitted in a single (*singularis*) witness, generally due to either scribal inattention or arbitrary innovation. In the majority of cases, *lectiones singulares* are late, secondary readings (see reading, secondary) that do not find place in the reconstructed text (cf. also *eliminatio lectionum singularium*). One possible exception is

Voce “lectio singularis” del PLS

³⁵ <https://wiki.helsinki.fi/display/stemmatology/Lectio+difficilior%2C+lectio+potior>.

³⁶ <https://lexiconse.uantwerpen.be/lexicon/lectioDifficilior.html>.

represented by *lectiones singulares difficiliore*s (Contini 1986, 101) which may be primary (see Reading, primary)".³⁷

mechanic lacuna

lacuna meccanica

Omission due to the physical loss of part of the witness.		Omissione dovuta alla perdita fisica di una parte del testimone.
---	--	--

“La lacuna è detta meccanica quando dipende dalla perdita fisica di una parte del manoscritto” Leonardi 2022

“A *lacuna*, literally in Latin a ‘hole, gap’, in a manuscript is a gap of missing text of varying length. ‘Lacuna is generally used with the physical object in mind (see material accidents), whereas ‘omission’ refers only to the text”.³⁸ Voce “lacuna” del PLS

monogenetic error

errore monogenetico

An error which is unlikely to have been committed by several copyists independently of each other.		Errore che è improbabile sia stato commesso da più copisti indipendentemente l’uno dall’altro.
--	--	--

“errore che è improbabile sia stato commesso da più copisti indipendentemente l’uno dall’altro” Leonardi 2022

Voce “error (monogenetic) del LexiconSE.³⁹

orthographic variant

variante ortografica

Reading that differs on a formal level (graphic and/or phonetic) from the reading considered original by the publisher.		Lezione che si differenzia su un piano formale (grafico e/o fonetico) rispetto alla lezione considerata originaria dall’editore.
---	--	--

“**forma** veste linguistica di un testo, negli aspetti più facilmente riconducibili da ogni copista al proprio sistema; le modifiche di questo livello sono dette varianti formali. Lo stesso che superficie (vedi anche pàtina); contrapposto a sostanza.” Leonardi 2022

³⁷ <https://wiki.helsinki.fi/display/stemmatology/Lectio+singularis>.

³⁸ <https://wiki.helsinki.fi/display/stemmatology/Lacuna>.

³⁹ <https://lexiconse.uantwerpen.be/lexicon/errorMonogenetic.html>.

“As Gaston Paris stated for the first time, in textual criticism a clear distinction should be made between ‘the analysis of readings’ (i.e. the substance of the text, in Contini’s terminology) and the ‘analysis of forms’ (i.e. the linguistic features of the text): ‘Les leçons sont établies sur la classification des manuscrits, les formes sont restituées d’après l’appréciation critique de la langue du poète’ [...] A further distinction should be made, within the analysis of forms, between what Trovato calls ‘core’ (‘the original language of a text’) and ‘patina’ (‘the linguistic sedimentation that is certainly due to the copyists’) (Trovato 2014, 231; see also Leonardi 2014, 45-52). Cf. vulgarisation”.⁴⁰

Voce “analysis of forms” del PLS

polygenetic error

errore poligenetico

An error that is likely to have been committed by several copyists independently of each other.	Errore che è probabile sia stato commesso da più copisti indipendentemente l’uno dall’altro.
---	--

“errore che è probabile sia stato commesso da più copisti indipendentemente l’uno dall’altro” Leonardi 2022

Voce “error (polygenetic)” del LexiconSE.⁴¹

separative error

errore separativo

Error present in a witness A and absent in a witness B, which allows to deduce that B does not derive from A.	Errore presente in un codice A e assente in un codice B, che consente di dedurre che B non deriva da A.
---	---

“errore presente in un codice A e assente in un codice B, che consente di dedurre che B non deriva da A” Leonardi 2022

Voce “error (separative)” del LexiconSE.⁴²

significant error

errore significativo (o errore-guida)

Conjunctive or separative error.	Errore congiuntivo o separativo.
----------------------------------	----------------------------------

⁴⁰ <https://wiki.helsinki.fi/display/stemmatology/Analysis+of+forms>.

⁴¹ <https://lexiconse.uantwerpen.be/lexicon/errorPolygenetic.html>.

⁴² <https://lexiconse.uantwerpen.be/lexicon/errorSeparative.html>.

“errore congiuntivo o separativo”

Leonardi 2022

substantive variant

variante sostanziale

Reading that differs on a semantic level from the reading considered original by the editor.

Lezione che si differenzia su un piano semantico rispetto alla lezione considerata originaria dall'editore.

“**sostanza** livello del testo meno soggetto alle modifiche superficiali dei copisti, in quanto più carico di contenuti semantici; le modifiche di questo livello sono dette varianti sostanziali. Contrapposto a forma.” Leonardi 2022

variant

variante

Reading of one witness different from that of another, without one of them necessarily being erroneous.

Lezione di un testimone diversa da quella di un altro, senza che una di esse sia necessariamente erronea.

“lezione di un testimone diversa da quella di un altro, senza che una di esse sia erronea” Leonardi 2022

“A variant in a witness (manuscript, edition...) is a reading which is different from other readings in other witnesses to the same text on the same variant location.” Voce “reading, variant” del PLS

Voce “reading (variant)” del LexiconSE.⁴³

Critical Intervention Type

emendatio

correction / correzione

A critical intervention can be classified as ‘emendatio’ when the editor corrects a passage in the text of the work.

Un intervento critico è classificabile come ‘emendatio’ quando l’editore corregge un luogo del testo dell’opera.

⁴³ <https://lexiconse.uantwerpen.be/lexicon/readingVariant.html>.

“A volte il più antico stadio del testo che sia possibile ricostruire in base alla tradizione coincide o può coincidere in tutti i suoi punti con l’originale; a volte restano dei passi evidentemente guasti, cioè vi è stato fra l’originale e la più antica ramificazione della tradizione, un intermediario, che si chiama archètipo, già inquinato da errori. Dove ciò si verifica, bisogna cercare di risalire al di là dell’archètipo, cioè di migliorare la lezione tramandata mediante congettura (*divinatio*): tale operazione si chiama emendatio.” Brambilla Ageno 1984

Voce “emendation” del LexiconSE.⁴⁴

emendatio ex fontibus

correction based on sources / correzione in base alle fonti

A critical intervention can be classified as ‘emendatio ex fontibus’ when the editor corrects a passage in the text, based on the sources they consider to have been used in writing the text of the work.

Un intervento critico è classificabile come ‘emendatio ex fontibus’ quando l’editore sana un luogo del testo, basandosi sulle fonti che ritiene che siano state utilizzate in fase di scrittura del testo dell’opera.

“lat. correzione in base alle fonti”

Leonardi 2022

emendatio ope codicum

correction based on witnesses / correzione in base ai testimoni

A critical intervention can be classified as ‘emendatio ope codicum’ when the editor corrects a passage in the text, resorting to a good reading transmitted by one or more witnesses.

Un intervento critico è classificabile come ‘emendatio ope codicum’ quando l’editore sana un luogo del testo, facendo ricorso a una lezione buona trasmessa da uno o più testimoni.

“correzione di un errore tramite il ricorso a una lezione buona trasmessa da uno o più altri codici” Leonardi 2022

emendatio ope ingenii

conjecture / congettura

A critical intervention can be classified as ‘emendatio ope ingenii’ when the editor corrects a place in the text, proposing a conjectural reading, not attested by any witness.

Un intervento critico è classificabile come ‘emendatio ope ingenii’ quando l’editore sana un luogo del testo, proponendo una lezione congetturale, non attestata in alcun testimone.

⁴⁴ <https://lexiconse.uantwerpen.be/lexicon/emendation.html>.

“lo stesso che congettura”

Leonardi 2022

“A conjecture is the introduction into the text of a reading without a basis in the witnesses available to the person doing the conjecture. One should differentiate between conjectures by scribes (see scribal conjecture) as opposed to conjectures by modern editors (which are done by *divinatio*, cf. also *emendatio*).”

Voce “conjecture” del PLS

Voce “conjectural” del LexiconSE.⁴⁵

selectio

selection / selezione

A critical intervention can be classified as ‘selectio’ when the editor constitutes the passage of the critical text by choosing among the variants.

Un intervento critico è classificabile come ‘*selectio*’ quando l’editore costituisce il luogo del testo critico scegliendo tra le varianti.

“scelta tra le varianti per la costituzione del testo critico”

Leonardi 2022

“*Selectio* consists in the choice between readings bearing the same stemmatic value (IT: varianti adiafore) on the basis of internal criteria such as usus scribendi or lectio difficilior. It is one of the three main operations that characterise emendatio, along with *divinatio* and *combinatio*”.⁴⁶

Voce “selectio” del PLS

Voce “apparatus (critical)” del LexiconSE.⁴⁷

Proprietà

has intervention criterion

ha il criterio di intervento

This property associates an instance of [Critical Intervention](#) with one or more instances of [Critical Intervention Criterion](#).

Questa proprietà associa un’istanza di Critical Intervention a una o più istanze di Critical Intervention Criterion.

⁴⁵ <https://lexiconse.uantwerpen.be/lexicon/conjectural.html>.

⁴⁶ <https://wiki.helsinki.fi/display/stemmatology/Analysis+of+forms>.

⁴⁷ <https://lexiconse.uantwerpen.be/lexicon/apparatusCritical.html>.

has intervention type

ha il tipo di intervento

This property associates an instance of [Critical Intervention](#) with an instance of [Critical Intervention Type](#).

Questa proprietà associa un'istanza di Critical Intervention a un'istanza di Critical Intervention Type.

has reading (is reading of)

ha la lezione (è la lezione di)

This property associates an instance of Witness with an instance of Reading. A reading is attested to by a single witness. A reading presented in the critical apparatus, on the other hand, can be common to two or more witnesses.

Questa proprietà associa un'istanza di Witness a un'istanza di Reading. Una lezione è attestata da un solo testimone. Una lezione presentata nell'apparato critico, invece, può essere comune a due o più testimoni.

is carried out on

è condotto su

This property associates an instance of [Critical Intervention](#) with one or more instances of [Reading](#) (or its subclasses) upon which the critical intervention is based.

Questa proprietà associa un'istanza di Critical Intervention a una o più istanze di Reading (o le sue sottoclassi) su cui è basato l'intervento critico.

is linguistic basis for (is linguistically based on)

è la base linguistica per (è linguisticamente basato su)

This property associates an instance of [Critical Text](#) with only one instance of [Surface Witness](#).

Questa proprietà associa un'istanza di Critical Text a una sola istanza di Surface Witness.

is main basis for (is mainly based on)

è la base principale per (è principalmente basato su)

This property associates an instance of [Critical Text](#) with only one instance of [Base Witness](#).

Questa proprietà associa un'istanza di Critical Text a una sola istanza di Base Witness.

performed by

effettuato da

This property associates an instance of [Critical Intervention](#) with an instance of [Editor](#).

Questa proprietà associa un'istanza di Critical Intervention a un'istanza di Editor.

prepared by

preparata da

This property associates an instance of [Reading In Apparatus](#) with an instance of [Critical Intervention](#). With the same critical intervention, the editor prepares multiple readings to be presented in the critical apparatus.

Questa proprietà associa un'istanza di Reading In Apparatus a un'istanza di Critical Intervention. Con il medesimo intervento critico l'editore prepara molteplici lezioni da presentare nell'apparato critico.

set by

stabilito da

This property associates an instance of [Critical Text Passage](#) with an instance of [Critical Intervention](#). With a critical intervention the editor establishes only one lesson to be included in the critical text.

Questa proprietà associa un'istanza di Critical Text Passage a un'istanza di Critical Intervention. Con un intervento critico l'editore stabilisce una sola lezione da includere nel testo critico.

Componenti di un'edizione scientifica

Classi

Accompanying Material

Materiale di corredo

Component of a scholarly edition of a textual or other nature that accompanies the edited text and is subordinate to it.

Componente di un'edizione scientifica di natura testuale o di altro tipo che accompagna il testo edito e ad esso è subordinato.

“Un’edizione critica [...], oltre al testo, fornisce in genere uno o più strumenti complementari” Leonardi 2022, p.133

Apparatus

Apparato

Component of a scholarly edition that accompanies the edited text and consists of a set of annotations that the editor has appended to edited text for a particular scientific purpose.	Componente di un’edizione scientifica che accompagna un testo edito ed è costituito da un insieme di annotazioni che l’editore ha apposto al testo edito per un particolare obiettivo scientifico.
---	--

“Editorial compositional structure in a scholarly edition to analyse text along certain criteria.” SE

“The word is derived from the Latin *apparātus* (gen. *apparātūs*) ‘preparation, equipment, instruments etc.’ which is an abstract noun to the verb *ad-parō* ‘to prepare, provide, furnish’. An *apparatus* (pl. *apparatus* or *apparatuses*) is a critical tool that complements the text of the editor and elucidates one or more of the following: the editorial (re)construction of the text (*critical apparatus*), the witnesses in which the current portion of the text is present, and the identification of source material referenced within a text. The term is most commonly used for the *critical apparatus*. ” Voce “apparatus” del PLS

Apparatus Entry

Voce d’apparato

An annotation presented within an apparatus.	Un’annotazione presentata all’interno di un apparato.
--	---

“Entry of an editing apparatus.” SE

Apparatus Entry Content

Contenuto della voce d’apparato

Textual content of an apparatus entry.	Contenuto testuale di una voce d’apparato.
--	--

Edited Text

Testo edito

Text of a work prepared by an editor or a group of editors within a scholarly edition.	Testo di un'opera allestito da un editore o da un gruppo di editori presentato all'interno di un'edizione scientifica.
--	--

“Text edited in a scholarly way.”

SE

Voce “text (edited)” del LexiconSE.⁴⁸

Edited Text Passage

Passo del testo edito

A portion of variable size of the edited text, made up of characters, words or sentences.

Una porzione di dimensione variabile del testo edito, costituita da caratteri, parole o frasi.

Introduction

Introduzione

Text of variable size which introduces the edited text within a scholarly edition, presenting the work itself and the work carried out by the editor. The introduction may be organized in different parts and may contain lists, pictures, and other materials.

Testo di dimensione variabile che all'interno di un'edizione scientifica introduce il testo edito, presentando l'opera e il lavoro svolto dall'editore. L'introduzione può essere articolata in più parti e può contenere elenchi, immagini e altri tipi di materiali.

Introduction Part

Parte dell'introduzione

A part of the introduction.

Una parte dell'introduzione.

Passage Selector

Selettore del passo

Sequence of characters that allows to identify exactly a passage within a text. There are several types of selector. Often it is a reference to the location of the passage, for example the line or verse number, or a copy of the passage concerned.

Sequenza di caratteri che permette di individuare esattamente un passo all'interno di un testo. Ci sono diversi tipi di selettore. Spesso si tratta di un riferi-

⁴⁸ <https://lexiconse.uantwerpen.be/lexicon/textEdited.html>.

mento alla posizione del passo, per esempio il numero di riga o verso, o di una copia del passo interessato.

“A resource which describes the segment of interest in a representation of a Source resource”⁴⁹ Web Annotation Vocabulary

Scholarly Edition Component

Componente di un'edizione scientifica

A component of a scholarly edition. Usually the component of a scholarly edition is a text, but it can also be an image, a multimedia object or another kind of object. Among the main components of a scholarly edition are the edited text and the materials accompanying the edited text.

Un componente di un'edizione scientifica. Solitamente il componente di un'edizione scientifica è un testo, ma può essere anche un'immagine, un oggetto multimediale o un altro tipo di oggetto. Tra i principali componenti di un'edizione scientifica ci sono il testo edito e i materiali di corredo al testo edito.

Proprietà

accompanies (is accompanied by)

correda (è corredato da)

This property associates an instance of [Accompanying Material](#) with the instance of [Edited Text](#) it accompanies. An edited text can have multiple accompanying materials.

Questa proprietà associa un'istanza di Accompanying Material all'istanza di Edited Text che correda. Un testo edito può avere molteplici materiali di corredo.

apparatus entry has content (is content of apparatus entry)

la voce d'apparato ha il contenuto (è il contenuto della voce d'apparato)

This property associates an apparatus entry, i.e. an instance of [Apparatus Entry](#), with its content, an instance of [Apparatus Entry Content](#).

Questa proprietà associa una voce d'apparato, ovvero un'istanza di Apparatus Entry, al suo contenuto, istanza di Apparatus Entry Content.

⁴⁹ <http://www.w3.org/ns/oa#Selector>.

apparatus entry refers to (is referred to by apparatus entry)

la voce d'apparato si riferisce a (è riferito dalla voce d'apparato)

This property associates an [Apparatus Entry](#) instance with the [Edited Text Passage](#) instance it refers to.

Questa proprietà associa un'istanza di Apparatus Entry all'istanza di Edited Text Passage cui è riferita.

edition has component (is component of edition)

l'edizione ha il componente (è il componente dell'edizione)

This property associates an instance of [Scholarly Edition](#) with an instance of [Scholarly Edition Component](#).

Questa proprietà associa un'istanza di Scholarly Edition a un'istanza di Scholarly Edition Component.

has apparatus (is apparatus of)

ha l'apparato (è l'apparato di)

This property associates an [Edited Text](#) instance with an [Apparatus](#) instance. Multiple apparatuses can be associated with the same edited text.

Questa proprietà associa un'istanza di Edited Text a un'istanza di Apparatus. Più apparati possono essere associati al medesimo testo edito.

has entry (is entry of)

ha la voce (è la voce di)

This property associates an instance of [Apparatus](#) instance with an instance of [Apparatus Entry](#). An apparatus contains at least one apparatus entry. Several apparatus entries are linked to an apparatus.

Questa proprietà associa un'istanza di Apparatus a un'istanza di Apparatus Entry. Un apparato contiene almeno una voce d'apparato. A un apparato sono collegate più voci d'apparato.

has passage (is passage of)

ha il passo (è il passo di)

This property associates an edited text, an instance of [Edited Text](#), with a passage contained within it, represented by an instance of [Edited Text Passage](#).

Questa proprietà associa un testo edito, istanza di Edited Text, a un passo contenuto al suo interno, rappresentato da un'istanza di Edited Text Passage.

is based on (is basis for)

è basato su (è la base per)

This property associates an instance of [Edited Text](#) with at least one instance of [Witness](#), which represents the witness from which it is derived.

Questa proprietà associa un'istanza di Edited Text ad almeno un'istanza di Witness, che rappresenta il testimone da cui è derivato.

passage has selector (is selector of passage)

il passo ha il selettore (è il selettore del passo)

This property associates a passage of the edited text, represented by an instance of [Edited Text Passage](#), to the selector that identifies it, represented by an instance of [Passage Selector](#).

Questa proprietà associa un passo del testo edito, rappresentato da un'istanza di Edited Text Passage, al selettore che lo individua, rappresentato da un'istanza di Passage Selector.

selector is shown in

il selettore è presentato in

This property associates the selector of a passage of the edited text, an instance of [Passage Selector](#), with the content of an apparatus entry, an instance of [Apparatus Entry Content](#), within which it is shown.

Questa proprietà associa il selettore di un passo del testo edito, istanza di Passage Selector, al contenuto di una voce d'apparato, istanza di Apparatus Entry Content, all'interno del quale viene mostrato.

Componenti di un'edizione critica

Classi

Base Reading In Apparatus

Lezione di base nell'apparato

The reading chosen or proposed by the editor, presented within an entry of the critical apparatus for the dual purpose of indicating the passage in the text to which the entry refers and to allow the reader to

La lezione scelta o proposta dall'editore, presentata all'interno di una voce dell'apparato critico al duplice scopo di esplicitare il punto del testo cui la voce fa riferimento e di consentire al lettore di metterla

compare it more easily with the other recorded readings. | più facilmente a confronto con le altre lezioni registrate.

“This class comprises special types of readings, i.e. the one reading within the same unit of variation preferred by the editor over the others.” CAO

“Editorial structure in an apparatus entry with a copy of one or more words or phrases referring to its edition text, and as appearing in the original or a witness, or created by an editor, and judged by the editor to be authentic, accurate, and/or authoritative [, and possibly compared with variants referring to other witnesses]” SE

“(lemma) contains the lemma, or base text, of a textual variation.”⁵⁰ TEI P5 Guidelines

Voce “lemma” del LexiconSE.⁵¹

Commentary

Commento

A type of apparatus that the editor appends to the critical text in order to offer more information on the textual transmission and to argue their editorial choices. | Un tipo di apparato che l'editore appone al testo critico per offrire maggiori informazioni sulla tradizione testuale e argomentare le proprie scelte editoriali.

Critical Apparatus

Apparato critico (o apparato delle varianti)

An apparatus through which the editor shows the differences between the critical text and the textual transmission and possibly explains their choices. | Apparato mediante il quale l'editore illustra le differenze tra il testo critico e la tradizione, motivando eventualmente le proprie scelte.

“settore dell'edizione in cui si registrano le differenze fra il testo critico e i manoscritti” Leonardi 2022, p. 189

Voce “apparatus (critical)” del LexiconSE.⁵²

⁵⁰ <https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-lem.html>.

⁵¹ <https://lexiconse.uantwerpen.be/lexicon/lemma.html>.

⁵² <https://lexiconse.uantwerpen.be/lexicon/apparatusCritical.html>.

Critical Apparatus Entry

Voce dell'apparato critico

An annotation to the critical text that is part of the critical apparatus, within which the editor presents the variants and, more generally, the readings attested in the textual tradition, comparing them with the relative passage in the critical text.

Un'annotazione al testo critico che fa parte dell'apparato critico, all'interno della quale l'editore presenta le varianti e, più in generale, le lezioni attestate nella tradizione, mettendole a confronto con il relativo passaggio nel testo critico.

Critical Apparatus Entry Content

Contenuto della voce dell'apparato critico

The content of a critical apparatus entry.

Il contenuto di una voce dell'apparato critico.

Critical Apparatus Entry Note

Nota della voce dell'apparato critico

Note that the editor inserts in an entry of the critical apparatus to specify some philological aspects of a reading, of a group of readings, of the chosen or proposed reading or, in general, of the textual tradition at that point of the text.

Nota che l'editore inserisce in una voce dell'apparato critico per specificare alcuni aspetti filologici di una lezione, di un gruppo di lezioni, della lezione scelta o proposta o, in generale, della tradizione testuale in quel punto del testo.

Critical Apparatus Group

Fascia dell'apparato critico

Structure in which the critical apparatus can be divided, in order to distinguish the different types of information it contains and facilitate its reading. A critical apparatus group may contain a particular type of readings (for example substantive variants or orthographic variants) or it may group together the information about the variation in a given set of witnesses (for example a family of witnesses).

Struttura in cui l'apparato critico può essere suddiviso, al fine di distinguere le diverse tipologie di informazioni contenute al suo interno e di facilitarne la consultazione. Una fascia dell'apparato critico può contenere una certa tipologia di lezioni (per esempio le varianti formali o le varianti sostanziali) oppure raggruppare le informazioni riguardanti alcuni sottogruppi della tradizione (per esempio una sotto-famiglia di testimoni).

Note

Nota

An annotation to the critical text, which is part of the commentary.	Un'annotazione al testo critico, che fa parte del commento.
--	---

Note Content

Contenuto della nota

Content of a note.	Contenuto di una nota.
--------------------	------------------------

Passage Position Selector

Selettore del passaggio in base alla posizione

Type of selector of a passage of the critical text which indicates where the passage is positioned, usually through the number of the structural element in which the passage is contained (for example the verse, the line, the paragraph, etc.) or through the cross reference to an alphanumeric exponent or other graphic sign.	Tipo di selettore di un passo del testo critico che indica dove il passo è posizionato, solitamente attraverso il numero dell'elemento strutturale in cui il passo è contenuto (per esempio il verso, la riga, il paragrafo, ecc.) oppure attraverso il rimando a un esponente alfanumerico o a un altro segno grafico.
---	---

“The TextPositionSelector describes a range of text by recording the start and end positions of the selection in the stream. Position 0 would be immediately before the first character, position 1 would be immediately before the second character, and so on.”⁵³

Web Annotation Vocabulary

Passage Quote Selector

Selettore del passaggio mediante la citazione

Type of selector of a passage of the critical text which indicates the passage by copying it in its entirety or in abbreviated form.	Tipo di selettore di un passo del testo critico che indica il passo copiandolo integralmente o in forma abbreviata.
--	---

⁵³ <https://www.w3.org/TR/annotation-vocab/#textpositionselector>.

“The TextQuoteSelector describes a range of text by copying it, and including some of the text immediately before (a prefix) and after (a suffix) it to distinguish between multiple copies of the same sequence of characters.”⁵⁴ Web Annotation Vocabulary

Reading Group

Gruppo di lezioni

A set of readings within an entry of the critical apparatus, grouped to highlight some peculiarities of the textual tradition or to facilitate consultation. A group of readings is graphically distinct from the other contents of the critical apparatus entry.	Un insieme di lezioni all'interno di una voce dell'apparato critico, raggruppate per mettere in evidenza alcune peculiarità della tradizione o per facilitare la consultazione. Un gruppo di lezioni è graficamente distinto dagli altri contenuti della voce dell'apparato critico.
---	--

Siglum

Sigla del testimone

Alphanumeric string that identifies a witness within a critical edition.	Stringa alfanumerica che identifica un testimone all'interno di un'edizione critica.
--	--

“stringa alfanumerica con cui si identifica un manoscritto” Leonardi 2022, p. 196

“A siglum (plural sigla) is a letter (or other character) used to designate any number of the following: a witness or a group (or a family) of witnesses, a vertex/node in a graph, or the archetype and hyparchetypes in a textual tradition or manuscript tradition.” Voce “siglum” del PLS

“One or more symbols as identifier of an expression as original or witness in a scholarly edition.” SE

Voce “siglum” del LexiconSE.⁵⁵

Siglum Reference

Rimando alla sigla del testimone

⁵⁴ <https://www.w3.org/TR/annotation-vocab/#textquoteselector>.

⁵⁵ <https://lexiconse.uantwerpen.be/lexicon/siglum.html>.

Reference to a witness siglum, used within the critical apparatus (and in other apparatuses), next to readings and optionally to reconstructed passage references.

Rimando alla sigla di un testimone, utilizzato nell'apparato critico (e anche negli altri apparati), accanto alle lezioni e opzionalmente ai rimandi al passo ricostruito.

“Reference to a siglum as expression identifier.”

SE

Stemma Codicum

(idem)

Graph that represents the genealogical tree of the textual transmission of a work.

Grafo che rappresenta l'albero genealogico della tradizione di un'opera.

“(lat. ‘albero dei codici’, anche soltanto stemma) grafico che rappresenta l'albero genealogico della tradizione di un testo”

Leonardi 2022, p. 196

Voce “stemma” del LexiconSE.⁵⁶

Witness Description

Descrizione del testimone

Text that can be provided within the witnesses list, to describe a witness in more or less depth and dealing with different aspects: the state of conservation, the material aspect, the palaeographic analysis, etc.

Testo che può essere fornito all'interno dell'elenco dei testimoni, per descrivere un testimone in modo più o meno approfondito e trattando diversi aspetti: lo stato di conservazione, l'aspetto materiale, l'analisi paleografica, ecc.

Witnesses List

Elenco dei testimoni (o testimoniale)

List of the witnesses that preserve the text of a work. The list is presented within a critical edition, usually inside the introduction.

Elenco dei testimoni che tramandano il testo di un'opera. L'elenco è presentato all'interno di un'edizione critica, solitamente nell'introduzione.

⁵⁶ <https://lexiconse.uantwerpen.be/lexicon/stemma.html>.

Proprietà

commentary has note (is note of commentary)

Il commento ha la nota (è la nota del commento)

This property associates a philological comment, instance of [Commentary](#), with a note it contains, instance of [Note](#). There is at least one note within a commentary.

Questa proprietà associa un commento filologico, istanza di [Commentary](#), a una nota che contiene, istanza di [Note](#). All'interno di un commento c'è almeno una nota.

critical apparatus entry has content (is content of critical apparatus entry)

la voce dell'apparato critico ha il contenuto (è il contenuto della voce dell'apparato critico)

This property associates an instance of [Critical Apparatus Entry](#) with its own content, instance of [Critical Apparatus Entry Content](#). Within the same critical apparatus entry there are contents of a different nature, represented by instances of the [Critical Apparatus Entry Content](#) subclasses.

Questa proprietà associa un'istanza di [Critical Apparatus Entry](#) al proprio contenuto, istanza di [Critical Apparatus Entry Content](#). All'interno della medesima voce dell'apparato critico ci sono contenuti di natura diversa, rappresentati da istanze delle sottoclassi di [Critical Apparatus Entry Content](#).

critical apparatus entry refers to (is referred to by critical apparatus entry)

la voce dell'apparato critico si riferisce a (è riferito dalla voce dell'apparato critico)

This property associates an entry of the critical apparatus, an instance of [Critical Apparatus Entry](#), with the passage of the critical text to which it refers, an instance of [Critical Text Passage](#).

Questa proprietà associa una voce dell'apparato critico, istanza di [Critical Apparatus Entry](#), al passo del testo critico cui è riferita, un'istanza di [Critical Text Passage](#).

critical apparatus has entry (is entry of critical apparatus)

l'apparato critico ha la voce (è voce dell'apparato critico)

This property associates an instance of [Critical Apparatus](#) with an instance of [Critical Apparatus Entry](#). A critical apparatus contains at least one entry.

Questa proprietà associa un'istanza di [Critical Apparatus](#) a un'istanza di [Critical Apparatus Entry](#). Un apparato critico contiene almeno una voce.

entry content has note

Il contenuto della voce dell'apparato critico ha una nota

This property associates an instance of [Critical Apparatus Entry Note](#) with an instance of [Critical Apparatus Entry Content](#) or one of its subclasses.

Questa proprietà associa un'istanza di Critical Apparatus Entry Note a un'istanza di Critical Apparatus Entry Content o di una delle sue sottoclassi.

entry has reading

La voce dell'apparato critico ha la lezione

This property associates an instance of [Critical Apparatus Entry](#) with one or more instances of [Reading In Apparatus](#). At least one reading is presented in each entry of the critical apparatus.

Questa proprietà associa un'istanza di Critical Apparatus Entry a un'istanza o più di Reading In Apparatus. In ogni voce dell'apparato critico è presentata almeno una lezione.

critical apparatus has group

l'apparato critico ha una fascia

This property associates an instance of [Critical Apparatus](#), with the groups into which it can be subdivided, that are instances of [Critical Apparatus Group](#).

Questa proprietà associa un apparato critico, istanza di Critical Apparatus, alle fasce in cui può essere suddiviso, istanze di Critical Apparatus Group.

has formal variant (is formal variant of)

ha la variante formale (è variante formale di)

This property associates an instance of [Base Reading In Apparatus](#) with instances of [Reading In Apparatus](#), which are formal variants of it.

Questa proprietà associa un'istanza di Base Reading In Apparatus, alle istanze di Reading In Apparatus, che sono delle sue varianti formali.

has indifferent variant (is indifferent variant of)

ha la variante adiafora (è variante adiafora di)

This property associates an instance of [Base Reading In Apparatus](#) with instances of [Reading In Apparatus](#), which are indifferent variants of it.

Questa proprietà associa un'istanza di Base Reading In Apparatus, alle istanze di Reading In Apparatus, che sono delle sue varianti adiafore.

has reading group

ha il gruppo di lezioni

This property associates an instance of [Critical Apparatus Entry](#) with one or more instances of [Reading Group](#), which it may contain.

Questa proprietà associa un'istanza di Critical Apparatus Entry a una o più istanze di Reading Group, che può presentare al proprio interno.

introduction has part (is part of introduction)

l'introduzione ha la parte (è parte dell'introduzione)

This property associates an instance of [Introduction](#) with one or more instances of [Introduction Part](#) it may contain.

Questa proprietà associa un'istanza di Introduction a una o più istanze di Introduction Part che può contenere.

is part of critical apparatus group

è parte di una fascia dell'apparato critico

This property associates an instance of [Critical Apparatus Entry](#) with an instance of [Critical Apparatus Group](#). Within a group of the critical apparatus there is at least one entry.

Questa proprietà associa un'istanza di Critical Apparatus Entry a un'istanza di Critical Apparatus Group. All'interno di una fascia dell'apparato critico c'è almeno una voce.

is part of reading group

è parte di un gruppo di lezioni

This property associates an instance of [Reading In Apparatus](#) with an instance of [Reading Group](#).

Questa proprietà associa un'istanza di Reading In Apparatus a un'istanza di Reading Group.

reading is witnessed by

la lezione è testimoniata da (oppure 'la lezione è attestata in')

This property associates an instance of [Reading In Apparatus](#) or [Base Reading In Apparatus](#) with one or more instances of [Siglum Reference](#).

Questa proprietà associa un'istanza di Reading In Apparatus o di Base Reading In Apparatus a una o più istanze di Siglum Reference.

refers to siglum

si riferisce alla sigla

This property associates the siglum of a witness, an instance of [Siglum](#), to its references contained within the entries of the critical apparatus, represented by instances of [Siglum Reference](#).

Questa proprietà associa la sigla di un testimone, un'istanza di Siglum, ai suoi rimandi contenuti all'interno delle voci dell'apparato critico, rappresentati da istanze di Siglum Reference.

siglum identifies (witness is identified by)

la sigla identifica (il testimone è identificato da)

This property associates an instance of [Witness](#) with the instance of [Siglum](#) which identifies it within the critical edition.

Questa proprietà associa un'istanza di Witness all'istanza di Siglum che lo identifica all'interno dell'edizione critica.

witness is described by

il testimone è descritto in

This property associates an instance of [Witness](#) with the instance of [Witness Description](#) in which it is described.

Questa proprietà associa un'istanza di Witness con l'istanza di Witness Description in cui è descritto.

witness is presented in

il testimone è presentato in

This property associates an instance of [Witness](#) with the instance of [Witnesses List](#) in which it is presented.

Questa proprietà associa un'istanza di Witness con l'istanza di Witnesses List in cui è presentato.

Appendice C

Critical Edition Ontology

```
1 <?xml version="1.0"?>
2 <rdf:RDF xmlns="http://purl.org/critical-edition-ontology#"
3   xml:base="http://purl.org/critical-edition-ontology"
4   xmlns:dc="http://purl.org/dc/elements/1.1/"
5   xmlns:owl="http://www.w3.org/2002/07/owl#"
6   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
7   xmlns:xml="http://www.w3.org/XML/1998/namespace"
8   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
9   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
10  xmlns:terms="http://purl.org/dc/terms/">
11   <owl:Ontology rdf:about="http://purl.org/critical-edition-ontology#"
12     <owl:versionIRI rdf:resource="http://purl.org/critical-edition-on-
13 tology"/>
14     <dc:date>28.06.2023</dc:date>
15     <dc:title>Critical Edition Ontology (CEO)</dc:title>
16     <terms:creator>Chiara Martignano</terms:creator>
17     <terms:license xml:lang="en">This work is distributed under a Crea-
18 tive Commons Attribution License (http://creativecommons.org/li-
19 censes/by/3.0/).</terms:license>
20     <owl:versionInfo
21 rdf:datatype="http://www.w3.org/2001/XMLSchema#decimal">1.0</owl:version-
22 Info>
23   </owl:Ontology>
24
25
26 <!--////////////////////////////////////
27 //
28 // Annotation properties
29 //
30 //////////////////////////////////-->
31
32
33 <!-- http://purl.org/dc/elements/1.1/date -->
34
35   <owl:AnnotationProperty rdf:about="http://purl.org/dc/ele-
36 ments/1.1/date"/>
37
38
39 <!-- http://purl.org/dc/elements/1.1/title -->
40
41   <owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/ti-
42 tle"/>
43
44
45 <!-- http://purl.org/dc/terms/creator -->
46
47   <owl:AnnotationProperty rdf:about="http://purl.org/dc/terms/creator"/>
48
49
50 <!-- http://purl.org/dc/terms/license -->
51
52   <owl:AnnotationProperty rdf:about="http://purl.org/dc/terms/license"/>
```

```

53
54
55
56 <!--////////////////////////////////////
57 //
58 // Object Properties
59 //
60 //////////////////////////////////----->
61
62
63 <!-- http://purl.org/critical-edition-ontology#R15_has_fragment -->
64
65     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
66 ogy#R15_has_fragment">
67         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
68 tology#R15_is_fragment_of"/>
69         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
70 ogy#F2_Expression"/>
71         <rdfs:range rdf:resource="http://www.cidoc-crm.org/cidoc-
72 crm/E90_Symbolic_Object"/>
73         <rdfs:comment xml:lang="en">This property associates an instance of
74 E90 Symbolic Object with an instance of F2 Expression of which it is a
75 fragment. The fragment is not itself an instance of F2 Expression as it
76 does not express any instance of F1 Work. When the fragment consists of in-
77 telligible words it is an instance of E33 Linguistic Object.</rdfs:comment>
78         <rdfs:label xml:lang="en">R15 has fragment</rdfs:label>
79         <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
80 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
81         </owl:ObjectProperty>
82
83
84 <!-- http://purl.org/critical-edition-ontology#R15_is_fragment_of -->
85
86     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
87 ogy#R15_is_fragment_of">
88         <rdfs:domain rdf:resource="http://www.cidoc-crm.org/cidoc-
89 crm/E90_Symbolic_Object"/>
90         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
91 ogy#F2_Expression"/>
92         <rdfs:label xml:lang="en">R15 is fragment of</rdfs:label>
93         <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
94 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
95         </owl:ObjectProperty>
96
97
98 <!-- http://purl.org/critical-edition-ontology#R3_is_realised_in -->
99
100     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
101 ogy#R3_is_realised_in">
102         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
103 tology#R3_realises"/>
104         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
105 ogy#F1_Work"/>
106         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
107 ogy#F2_Expression"/>
108         <rdfs:label xml:lang="en">R3 is realised in</rdfs:label>
109         <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
110 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
111         </owl:ObjectProperty>
112
113

```

```

114 <!-- http://purl.org/critical-edition-ontology#R3_realises -->
115
116   <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
117 ogy#R3_realises">
118     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
119 ogy#F2_Expression"/>
120     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
121 ogy#F1_Work"/>
122     <rdfs:comment xml:lang="en">This property associates an instance of
123 F2 Expression with an instance of F1 Work.</rdfs:comment>
124     <rdfs:label xml:lang="en">R3 realises</rdfs:label>
125     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
126 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
127   </owl:ObjectProperty>
128
129
130 <!-- http://purl.org/critical-edition-ontology#R4_embodies -->
131
132   <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
133 ogy#R4_embodies">
134     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
135 tology#R4_is_embodied_in"/>
136     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
137 tology#R5_is_component_of"/>
138     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
139 ogy#F3_Manifestation"/>
140     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
141 ogy#F2_Expression"/>
142     <rdfs:comment xml:lang="en">This property associates an instance of
143 F3 Manifestation with one or more instances of F2 Expression which are ren-
144 dered by this instance of F3 Manifestation. The manifestation formats the
145 expression(s) in the way they are to be presented to some public, including
146 specifying the intended sensory impression (such as visual appearance or
147 audio rendition).</rdfs:comment>
148     <rdfs:label xml:lang="en">R4 embodies</rdfs:label>
149     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
150 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
151   </owl:ObjectProperty>
152
153
154 <!-- http://purl.org/critical-edition-ontology#R4_is_embodied_in -->
155
156   <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
157 ogy#R4_is_embodied_in">
158     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
159 ogy#F2_Expression"/>
160     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
161 ogy#F3_Manifestation"/>
162     <rdfs:label xml:lang="en">R4 is embodied in</rdfs:label>
163     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
164 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
165   </owl:ObjectProperty>
166
167
168 <!-- http://purl.org/critical-edition-ontology#R5_has_component -->
169
170   <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
171 ogy#R5_has_component">
172     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
173 ogy#F2_Expression"/>

```

```

174     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
175 ogy#F2_Expression"/>
176     <rdfs:comment xml:lang="en">This property associates an instance of
177 an F2 Expression X with a structural component Y that conveys a part of the
178 overall work realized by X, such as volumes, chapters, or sections. This
179 property is transitive, asymmetric and irreflexive.</rdfs:comment>
180     <rdfs:label xml:lang="en">R5 has component</rdfs:label>
181     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
182 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
183     </owl:ObjectProperty>
184
185
186 <!-- http://purl.org/critical-edition-ontology#R5_is_component_of -->
187
188     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
189 ogy#R5_is_component_of">
190     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
191 ogy#F2_Expression"/>
192     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
193 ogy#F2_Expression"/>
194     <rdfs:label xml:lang="en">R5 is component of</rdfs:label>
195     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
196 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
197     </owl:ObjectProperty>
198
199
200 <!-- http://purl.org/critical-edition-ontology#R76_has_derivative -->
201
202     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
203 ogy#R76_has_derivative">
204     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-ontol-
205 ogy#R76_is_derivative_of"/>
206     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
207 ogy#F2_Expression"/>
208     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
209 ogy#F2_Expression"/>
210     <rdfs:comment xml:lang="en">This property associates an instance of
211 F2 Expression with another instance of F2 Expression which was its source
212 or one of its sources. This property is not transitive. It is asymmetric
213 and irreflexive.</rdfs:comment>
214     <rdfs:label xml:lang="en">R76 has derivative</rdfs:label>
215     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
216 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
217     </owl:ObjectProperty>
218
219
220 <!-- http://purl.org/critical-edition-ontology#R76_is_derivative_of -->
221
222     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
223 ogy#R76_is_derivative_of">
224     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
225 ogy#F2_Expression"/>
226     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
227 ogy#F2_Expression"/>
228     <rdfs:label xml:lang="en">R76 is derivative of</rdfs:label>
229     <rdfs:seeAlso>https://www.cidoc-crm.org/frbroo/sites/de-
230 fault/files/LRMoo_V0.9%28draft%20for%20WLIC%202022%29.pdf</rdfs:seeAlso>
231     </owl:ObjectProperty>
232
233

```



```

234 <!-- http://purl.org/critical-edition-ontology#accompanies -->
235
236 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
237 ogy#accompanies">
238 <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
239 tology#isAccompaniedBy"/>
240 <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
241 ogy#AccompanyingMaterial"/>
242 <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
243 ogy#EditedText"/>
244 <rdfs:comment xml:lang="en">This property associates an instance of
245 Accompanying Material with the instance of Edited Text it accompanies. An
246 edited text can have multiple accompanying materials.</rdfs:comment>
247 <rdfs:label xml:lang="en">accompanies</rdfs:label>
248 </owl:ObjectProperty>
249
250
251 <!--http://purl.org/critical-edition-ontology#apparatusEntryHasContent-->
252
253 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
254 ogy#apparatusEntryHasContent">
255 <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
256 tology#isContentOfApparatusEntry"/>
257 <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
258 ogy#ApparatusEntry"/>
259 <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
260 ogy#ApparatusEntryContent"/>
261 <rdfs:comment xml:lang="en">This property associates an apparatus
262 entry, i.e. an instance of Apparatus Entry, with its content, an instance
263 of Apparatus Entry Content.</rdfs:comment>
264 <rdfs:label xml:lang="en">apparatus entry has content</rdfs:label>
265 </owl:ObjectProperty>
266
267
268 <!-- http://purl.org/critical-edition-ontology#apparatusEntryRefersTo -->
269
270 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
271 ogy#apparatusEntryRefersTo">
272 <rdfs:subPropertyOf rdf:resource="http://www.cidoc-crm.org/cidoc-
273 crm/P67_refers_to"/>
274 <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
275 tology#isReferredToByApparatusEntry"/>
276 <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
277 ogy#ApparatusEntry"/>
278 <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
279 ogy#EditedTextPassage"/>
280 <rdfs:comment xml:lang="en">This property associates an Apparatus
281 Entry instance with the Edited Text Passage instance it refers
282 to.</rdfs:comment>
283 <rdfs:label xml:lang="en">apparatus entry refers to</rdfs:label>
284 </owl:ObjectProperty>
285
286
287 <!-- http://purl.org/critical-edition-ontology#commentaryHasNote -->
288
289 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
290 ogy#commentaryHasNote">
291 <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
292 ontology#hasEntry"/>
293 <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
294 tology#isNoteOfCommentary"/>

```

```

295     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
296 ogy#Commentary"/>
297     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
298 ogy#Note"/>
299     <rdfs:comment xml:lang="en">This property associates a philological
300 comment, instance of Commentary, with a note it contains, instance of Note.
301 There is at least one note within a commentary.</rdfs:comment>
302     <rdfs:label xml:lang="en">commentary has note</rdfs:label>
303     </owl:ObjectProperty>
304
305
306 <!-- http://purl.org/critical-edition-ontology#created -->
307
308     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
309 ogy#created">
310         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
311 tology#wasCreatedBy"/>
312         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
313 ogy#Editor"/>
314         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
315 ogy#ScholarlyEdition"/>
316         <rdfs:comment xml:lang="en">This property associates an instance of
317 Editor or EditorsGroup with an instance of Scholarly Edition they cre-
318 ated.</rdfs:comment>
319         <rdfs:label xml:lang="en">created</rdfs:label>
320     </owl:ObjectProperty>
321
322
323 <!-- http://purl.org/critical-edition-ontology#criticalApparatusEntry-
324 HasContent -->
325
326     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
327 ogy#criticalApparatusEntryHasContent">
328         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
329 ontology#apparatusEntryHasContent"/>
330         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
331 tology#isContentOfCriticalApparatusEntry"/>
332         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
333 ogy#CriticalApparatusEntry"/>
334         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
335 ogy#CriticalApparatusEntryContent"/>
336         <rdfs:comment xml:lang="en">This property associates an instance of
337 Critical Apparatus Entry with its own content, instance of Critical Appa-
338 ratus Entry Content. Within the same critical apparatus entry there are
339 contents of a different nature, represented by instances of the Critical
340 Apparatus Entry Content subclasses.</rdfs:comment>
341         <rdfs:label xml:lang="en">critical apparatus entry has con-
342 tent</rdfs:label>
343     </owl:ObjectProperty>
344
345
346 <!-- http://purl.org/critical-edition-ontology#criticalApparatusEntryRe-
347 fersTo -->
348
349     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
350 ogy#criticalApparatusEntryRefersTo">
351         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
352 ontology#apparatusEntryRefersTo"/>
353         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
354 tology#isReferredToByCriticalApparatusEntry"/>

```

```

355     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
356 ogy#CriticalApparatusEntry"/>
357     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
358 ogy#CriticalTextPassage"/>
359     <rdfs:comment xml:lang="en">This property associates an entry of
360 the critical apparatus, an instance of Critical Apparatus Entry, with the
361 passage of the critical text to which it refers, an instance of Critical
362 Text Passage.</rdfs:comment>
363     <rdfs:label xml:lang="en">critical apparatus entry refers
364 to</rdfs:label>
365     </owl:ObjectProperty>
366
367
368 <!--http://purl.org/critical-edition-ontology#criticalApparatusHasEntry -->
369
370     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
371 ogy#criticalApparatusHasEntry">
372     <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
373 ontology#hasEntry"/>
374     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
375 tology#isEntryOfCriticalApparatus"/>
376     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
377 ogy#CriticalApparatus"/>
378     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
379 ogy#CriticalApparatusEntry"/>
380     <rdfs:comment xml:lang="en">This property associates an instance of
381 Critical Apparatus with an instance of Critical Apparatus Entry. A critical
382 apparatus contains at least one entry.</rdfs:comment>
383     <rdfs:label xml:lang="en">critical apparatus has entry</rdfs:label>
384     </owl:ObjectProperty>
385
386
387 <!--http://purl.org/critical-edition-ontology#criticalApparatusHasGroup -->
388
389     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
390 ogy#criticalApparatusHasGroup">
391     <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
392 ontology#R5_has_component"/>
393     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
394 ogy#CriticalApparatus"/>
395     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
396 ogy#CriticalApparatusGroup"/>
397     <rdfs:comment xml:lang="en">This property associates an instance of
398 Critical Apparatus, with the groups into which it can be subdivided, that
399 are instances of Critical Apparatus Group.</rdfs:comment>
400     <rdfs:label xml:lang="en">critical apparatus has group</rdfs:label>
401     </owl:ObjectProperty>
402
403
404 <!-- http://purl.org/critical-edition-ontology#editionHasComponent -->
405
406     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
407 ogy#editionHasComponent">
408     <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
409 ontology#R5_has_component"/>
410     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
411 tology#isComponentOfEdition"/>
412     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
413 ogy#ScholarlyEdition"/>
414     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
415 ogy#ScholarlyEditionComponent"/>

```

```

416         <rdfs:comment xml:lang="en">This property associates an instance of
417 Scholarly Edition with an instance of Scholarly Edition Compo-
418 nent.</rdfs:comment>
419         <rdfs:label xml:lang="en">edition has component</rdfs:label>
420         </owl:ObjectProperty>
421
422
423 <!-- http://purl.org/critical-edition-ontology#entryContentHasNote -->
424
425         <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
426 ogy#entryContentHasNote">
427             <rdfs:subPropertyOf rdf:resource="http://www.cidoc-crm.org/cidoc-
428 crm/P3_has_note"/>
429             <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
430 ogy#CriticalApparatusEntryContent"/>
431             <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
432 ogy#CriticalApparatusEntryNote"/>
433             <rdfs:comment xml:lang="en">This property associates an instance of
434 Critical Apparatus Entry Note with an instance of Critical Apparatus Entry
435 Content or one of its subclasses.</rdfs:comment>
436             <rdfs:label xml:lang="en">entry content has note</rdfs:label>
437         </owl:ObjectProperty>
438
439
440 <!-- http://purl.org/critical-edition-ontology#entryHasReading -->
441
442         <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
443 ogy#entryHasReading">
444             <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
445 ogy#CriticalApparatusEntry"/>
446             <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
447 ogy#ReadingInApparatus"/>
448             <rdfs:comment xml:lang="en">This property associates an instance of
449 Critical Apparatus Entry with one or more instances of Reading In Appa-
450 ratus. At least one reading is presented in each entry of the critical ap-
451 paratus.</rdfs:comment>
452             <rdfs:label xml:lang="en">entry has reading</rdfs:label>
453         </owl:ObjectProperty>
454
455
456 <!-- http://purl.org/critical-edition-ontology#follows -->
457
458         <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
459 ogy#follows">
460             <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
461 ogy#ReadingInApparatus"/>
462             <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
463 ogy#ReadingInApparatus"/>
464             <rdfs:label xml:lang="en">follows</rdfs:label>
465         </owl:ObjectProperty>
466
467
468 <!-- http://purl.org/critical-edition-ontology#hasApparatus -->
469
470         <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
471 ogy#hasApparatus">
472             <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
473 ontology#isAccompaniedBy"/>
474             <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
475 tology#isApparatusOf"/>

```

```

476     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
477 ogy#EditedText"/>
478     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
479 ogy#Apparatus"/>
480     <rdfs:comment xml:lang="en">This property associates an Edited Text
481 instance with an Apparatus instance. Multiple apparatuses can be associated
482 with the same edited text.</rdfs:comment>
483     <rdfs:label xml:lang="en">has apparatus</rdfs:label>
484     </owl:ObjectProperty>
485
486
487 <!-- http://purl.org/critical-edition-ontology#hasEntry -->
488
489     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
490 ogy#hasEntry">
491         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
492 ontology#R5_has_component"/>
493         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
494 tology#isEntryOf"/>
495         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
496 ogy#Apparatus"/>
497         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
498 ogy#ApparatusEntry"/>
499         <rdfs:comment xml:lang="en">This property associates an instance of
500 Apparatus instance with an instance of Apparatus Entry. An apparatus con-
501 tains at least one apparatus entry. Several apparatus entries are linked to
502 an apparatus.</rdfs:comment>
503         <rdfs:label xml:lang="en">has entry</rdfs:label>
504     </owl:ObjectProperty>
505
506
507 <!-- http://purl.org/critical-edition-ontology#hasFormalVariant -->
508
509     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
510 ogy#hasFormalVariant">
511         <rdfs:subPropertyOf rdf:resource="https://w3id.org/cao#hasVari-
512 ant"/>
513         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
514 tology#isFormalVariantOf"/>
515         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
516 ogy#BaseReadingInApparatus"/>
517         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
518 ogy#ReadingInApparatus"/>
519         <rdfs:comment xml:lang="en">This property associates an instance of
520 Base Reading In Apparatus with instances of Reading In Apparatus, which are
521 formal variants of it.</rdfs:comment>
522         <rdfs:label xml:lang="en">has formal variant</rdfs:label>
523     </owl:ObjectProperty>
524
525
526 <!-- http://purl.org/critical-edition-ontology#hasIndifferentVariant -->
527
528     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
529 ogy#hasIndifferentVariant">
530         <rdfs:subPropertyOf rdf:resource="https://w3id.org/cao#hasVari-
531 ant"/>
532         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
533 tology#isIndifferentVariantOf"/>
534         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
535 ogy#BaseReadingInApparatus"/>

```

```

536     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
537 ogy#ReadingInApparatus"/>
538     <rdfs:comment xml:lang="en">This property associates an instance of
539 Base Reading In Apparatus with instances of Reading In Apparatus, which are
540 indifferent variants of it.</rdfs:comment>
541     <rdfs:label xml:lang="en">has indifferent variant</rdfs:label>
542     </owl:ObjectProperty>
543
544
545 <!-- http://purl.org/critical-edition-ontology#hasInterventionCriterion -->
546
547     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
548 ogy#hasInterventionCriterion">
549     <rdfs:subPropertyOf rdf:resource="http://purl.org/emmedi/hico#has-
550 InterpretationCriterion"/>
551     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
552 ogy#CriticalIntervention"/>
553     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
554 ogy#CriticalInterventionCriterion"/>
555     <rdfs:comment xml:lang="en">This property associates an instance of
556 Critical Intervention with one or more instances of Critical Intervention
557 Criterion.</rdfs:comment>
558     <rdfs:label xml:lang="en">has intervention criterion</rdfs:label>
559     </owl:ObjectProperty>
560
561
562 <!-- http://purl.org/critical-edition-ontology#hasInterventionType -->
563
564     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
565 ogy#hasInterventionType">
566     <rdfs:subPropertyOf rdf:resource="http://purl.org/emmedi/hico#has-
567 InterpretationType"/>
568     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
569 ogy#CriticalIntervention"/>
570     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
571 ogy#CriticalInterventionType"/>
572     <rdfs:comment xml:lang="en">This property associates an instance of
573 Critical Intervention with an instance of Critical Intervention
574 Type.</rdfs:comment>
575     <rdfs:label xml:lang="en">has intervention type</rdfs:label>
576     </owl:ObjectProperty>
577
578
579 <!-- http://purl.org/critical-edition-ontology#hasPart -->
580
581     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
582 ogy#hasPart">
583     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
584 tology#isPartOf"/>
585     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
586 ogy#TextualTradition"/>
587     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
588 ogy#Witness"/>
589     <rdfs:comment xml:lang="en">This property associates an instance of
590 Textual Transmission to an instance of Witness and represents the relation-
591 ship between a witness and the textual tradition to which it be-
592 longs.</rdfs:comment>
593     <rdfs:label xml:lang="en">has part</rdfs:label>
594     </owl:ObjectProperty>
595
596

```

```

597 <!-- http://purl.org/critical-edition-ontology#hasPassage -->
598
599     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
600 ogy#hasPassage">
601         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
602 ontology#R15_has_fragment"/>
603         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
604 tology#isPassageOf"/>
605         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
606 ogy#EditedText"/>
607         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
608 ogy#EditedTextPassage"/>
609         <rdfs:comment xml:lang="en">This property associates an edited
610 text, an instance of Edited Text, with a passage contained within it, rep-
611 resented by an instance of Edited Text Passage.</rdfs:comment>
612         <rdfs:label xml:lang="en">has passage</rdfs:label>
613     </owl:ObjectProperty>
614
615
616 <!-- http://purl.org/critical-edition-ontology#hasReading -->
617
618     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
619 ogy#hasReading">
620         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
621 ontology#R15_has_fragment"/>
622         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
623 tology#isReadingOf"/>
624         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
625 ogy#Witness"/>
626         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
627 ogy#Reading"/>
628         <rdfs:comment xml:lang="en">This property associates an instance of
629 Witness with an instance of Reading. A reading is attested to by a single
630 witness. A reading presented in the critical apparatus, on the other hand,
631 can be common to two or more witnesses.</rdfs:comment>
632         <rdfs:label xml:lang="en">has reading</rdfs:label>
633     </owl:ObjectProperty>
634
635
636 <!-- http://purl.org/critical-edition-ontology#hasReadingGroup -->
637
638     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
639 ogy#hasReadingGroup">
640         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
641 ogy#CriticalApparatusEntry"/>
642         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
643 ogy#ReadingGroup"/>
644         <rdfs:comment xml:lang="en">This property associates an instance of
645 Critical Apparatus Entry with one or more instances of Reading Group, which
646 it may contain.</rdfs:comment>
647         <rdfs:label xml:lang="en">has reading group</rdfs:label>
648     </owl:ObjectProperty>
649
650
651 <!-- https://www.w3.org/ns/oa#hasSelector -->
652
653     <owl:ObjectProperty rdf:about="https://www.w3.org/ns/oa#hasSelector">
654         <owl:equivalentProperty rdf:resource="http://purl.org/critical-edition-
655 ontology#passageHasSelector"/>
656     </owl:ObjectProperty>
657

```

```

658
659     <!-- http://purl.org/critical-edition-ontology#introductionHasPart -->
660
661     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
662 ogy#introductionHasPart">
663         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
664 ontology#R5_has_component"/>
665         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
666 tology#isPartOfIntroduction"/>
667         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
668 ogy#Introduction"/>
669         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
670 ogy#IntroductionPart"/>
671         <rdfs:comment xml:lang="en">This property associates an instance of
672 Introduction with one or more instances of Introduction Part it may con-
673 tain.</rdfs:comment>
674         <rdfs:label xml:lang="en">introduction has part</rdfs:label>
675     </owl:ObjectProperty>
676
677
678 <!-- http://purl.org/critical-edition-ontology#isAccompaniedBy -->
679
680     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
681 ogy#isAccompaniedBy">
682         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
683 ogy#EditedText"/>
684         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
685 ogy#AccompanyingMaterial"/>
686         <rdfs:label xml:lang="en">is accompanied by</rdfs:label>
687     </owl:ObjectProperty>
688
689
690 <!-- http://purl.org/critical-edition-ontology#isApparatusOf -->
691
692     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
693 ogy#isApparatusOf">
694         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
695 ontology#accompanies"/>
696         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
697 ogy#Apparatus"/>
698         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
699 ogy#EditedText"/>
700         <rdfs:label xml:lang="en">is apparatus of</rdfs:label>
701     </owl:ObjectProperty>
702
703
704 <!-- http://purl.org/critical-edition-ontology#isBasedOn -->
705
706     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
707 ogy#isBasedOn">
708         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
709 ontology#R76_is_derivative_of"/>
710         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
711 tology#isBasisFor"/>
712         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
713 ogy#EditedText"/>
714         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
715 ogy#Witness"/>
716         <rdfs:comment xml:lang="en">This property associates an instance of
717 Edited Text with at least one instance of Witness, which represents the
718 witness from which it is derived.</rdfs:comment>

```



```

719     <rdfs:label xml:lang="en">is based on</rdfs:label>
720 </owl:ObjectProperty>
721
722
723 <!-- http://purl.org/critical-edition-ontology#isBasisFor -->
724
725     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
726 ogy#isBasisFor">
727         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
728 ontology#R76_has_derivative"/>
729         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
730 ogy#Witness"/>
731         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
732 ogy#EditedText"/>
733         <rdfs:label xml:lang="en">is basis for</rdfs:label>
734     </owl:ObjectProperty>
735
736
737 <!-- http://purl.org/critical-edition-ontology#isCarriedOutOn -->
738
739     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
740 ogy#isCarriedOutOn">
741         <rdfs:subPropertyOf rdf:resource="http://purl.org/emmedi/hico#isEx-
742 tractedFrom"/>
743         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
744 ogy#CriticalIntervention"/>
745         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
746 ogy#Reading"/>
747         <rdfs:comment xml:lang="en">This property associates an instance of
748 Critical Intervention with one or more instances of Reading (or its sub-
749 classes) upon which the critical intervention is based.</rdfs:comment>
750         <rdfs:label xml:lang="en">is carried out on</rdfs:label>
751     </owl:ObjectProperty>
752
753
754 <!-- http://purl.org/critical-edition-ontology#isComponentOfEdition -->
755
756     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
757 ogy#isComponentOfEdition">
758         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
759 ontology#R5_is_component_of"/>
760         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
761 ogy#ScholarlyEditionComponent"/>
762         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
763 ogy#ScholarlyEdition"/>
764         <rdfs:label xml:lang="en">is component of edition</rdfs:label>
765     </owl:ObjectProperty>
766
767
768 <!--http://purl.org/critical-edition-ontology#isContentOfApparatusEntry -->
769
770     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
771 ogy#isContentOfApparatusEntry">
772         <owl:equivalentProperty rdf:re-
773 source="http://www.w3.org/ns/oa#hasBody"/>
774         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
775 ogy#ApparatusEntryContent"/>
776         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
777 ogy#ApparatusEntry"/>
778         <rdfs:label xml:lang="en">is content of apparatus entry</rdfs:la-
779 bel>

```

```

780     </owl:ObjectProperty>
781
782
783 <!-- http://purl.org/critical-edition-ontology#isContentOfCriticalApparat-
784 usEntry -->
785
786     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
787 ogy#isContentOfCriticalApparatusEntry">
788         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
789 ontology#isContentOfApparatusEntry"/>
790         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
791 ogy#CriticalApparatusEntryContent"/>
792         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
793 ogy#CriticalApparatusEntry"/>
794         <rdfs:label xml:lang="en">is content of critical apparatus en-
795 try</rdfs:label>
796     </owl:ObjectProperty>
797
798
799 <!-- http://purl.org/critical-edition-ontology#isEditedIn -->
800
801     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
802 ogy#isEditedIn">
803         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
804 ontology#R3_is_realised_in"/>
805         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
806 tology#isEditionOf"/>
807         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
808 ogy#F1_Work"/>
809         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
810 ogy#ScholarlyEdition"/>
811         <rdfs:comment xml:lang="en">This property associates an instance of
812 F1 Work with an instance of Scholarly Edition. A scholarly edition is one
813 of the realizations of a work.</rdfs:comment>
814         <rdfs:label xml:lang="en">is edited in</rdfs:label>
815     </owl:ObjectProperty>
816
817
818 <!-- http://purl.org/critical-edition-ontology#isEditionOf -->
819
820     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
821 ogy#isEditionOf">
822         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
823 ontology#R3_realises"/>
824         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
825 ogy#ScholarlyEdition"/>
826         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
827 ogy#F1_Work"/>
828         <rdfs:label xml:lang="en">is edition of</rdfs:label>
829     </owl:ObjectProperty>
830
831
832 <!-- http://purl.org/critical-edition-ontology#isEntryOf -->
833
834     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
835 ogy#isEntryOf">
836         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
837 ontology#R5_is_component_of"/>
838         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
839 ogy#ApparatusEntry"/>

```

```

840     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
841 ogy#Apparatus"/>
842     <rdfs:label xml:lang="en">is entry of</rdfs:label>
843 </owl:ObjectProperty>
844
845
846 <!--http://purl.org/critical-edition-ontology#isEntryOfCriticalApparatus-->
847
848     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
849 ogy#isEntryOfCriticalApparatus">
850         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
851 ontology#isEntryOf"/>
852         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
853 ogy#CriticalApparatusEntry"/>
854         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
855 ogy#CriticalApparatus"/>
856         <rdfs:label xml:lang="en">is entry of critical apparatus</rdfs:la-
857 bel>
858     </owl:ObjectProperty>
859
860
861 <!-- http://purl.org/critical-edition-ontology#isFormalVariantOf -->
862
863     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
864 ogy#isFormalVariantOf">
865         <rdfs:subPropertyOf rdf:resource="https://w3id.org/cao#isVarian-
866 tOf"/>
867         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
868 ogy#ReadingInApparatus"/>
869         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
870 ogy#BaseReadingInApparatus"/>
871         <rdfs:label xml:lang="en">is formal variant of</rdfs:label>
872     </owl:ObjectProperty>
873
874
875 <!-- http://purl.org/critical-edition-ontology#isIndifferentVariantOf -->
876
877     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
878 ogy#isIndifferentVariantOf">
879         <rdfs:subPropertyOf rdf:resource="https://w3id.org/cao#isVarian-
880 tOf"/>
881         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
882 ogy#ReadingInApparatus"/>
883         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
884 ogy#BaseReadingInApparatus"/>
885         <rdfs:label xml:lang="en">is indifferent variant of</rdfs:label>
886     </owl:ObjectProperty>
887
888
889 <!-- http://purl.org/critical-edition-ontology#isLinguisticBasisFor -->
890
891     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
892 ogy#isLinguisticBasisFor">
893         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
894 ontology#R76_has_derivative"/>
895         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
896 tology#isLinguisticallyBasedOn"/>
897         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
898 ogy#SurfaceWitness"/>
899         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
900 ogy#CriticalText"/>

```

```

901         <rdfs:comment xml:lang="en">This property associates an instance of
902 Critical Text with only one instance of Surface Witness.</rdfs:comment>
903         <rdfs:label xml:lang="en">is linguistic basis for</rdfs:label>
904     </owl:ObjectProperty>
905
906
907     <!-- http://purl.org/critical-edition-ontology#isLinguisticallyBasedOn -->
908
909     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
910 ogy#isLinguisticallyBasedOn">
911         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
912 ontology#R76_is_derivative_of"/>
913         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
914 ogy#CriticalText"/>
915         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
916 ogy#SurfaceWitness"/>
917         <rdfs:label xml:lang="en">is linguistically based on</rdfs:label>
918     </owl:ObjectProperty>
919
920
921     <!-- http://purl.org/critical-edition-ontology#isMainBasisFor -->
922
923     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
924 ogy#isMainBasisFor">
925         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
926 ontology#R76_has_derivative"/>
927         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
928 tology#isMainlyBasedOn"/>
929         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
930 ogy#BaseWitness"/>
931         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
932 ogy#CriticalText"/>
933         <rdfs:comment xml:lang="en">This property associates an instance of
934 Critical Text with only one instance of Base Witness.</rdfs:comment>
935         <rdfs:label xml:lang="en">is main basis for</rdfs:label>
936     </owl:ObjectProperty>
937
938
939     <!-- http://purl.org/critical-edition-ontology#isMainlyBasedOn -->
940
941     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
942 ogy#isMainlyBasedOn">
943         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
944 ontology#R76_is_derivative_of"/>
945         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
946 ogy#CriticalText"/>
947         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
948 ogy#BaseWitness"/>
949         <rdfs:label xml:lang="en">is mainly based on</rdfs:label>
950     </owl:ObjectProperty>
951
952
953     <!-- http://purl.org/critical-edition-ontology#isNoteOfCommentary -->
954
955     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
956 ogy#isNoteOfCommentary">
957         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
958 ontology#isEntryOf"/>
959         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
960 ogy#Note"/>

```

```

961         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
962 ogy#Commentary"/>
963         <rdfs:label xml:lang="en">is note of commentary</rdfs:label>
964     </owl:ObjectProperty>
965
966
967 <!-- http://purl.org/critical-edition-ontology#isPartOf -->
968
969     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
970 ogy#isPartOf">
971         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
972 ogy#Witness"/>
973         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
974 ogy#TextualTradition"/>
975         <rdfs:label xml:lang="en">is part of</rdfs:label>
976     </owl:ObjectProperty>
977
978
979 <!-- http://purl.org/critical-edition-ontology#isPartOfCriticalAppa-
980 ratusGroup -->
981
982     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
983 ogy#isPartOfCriticalApparatusGroup">
984         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
985 ontology#R5_is_component_of"/>
986         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
987 ogy#CriticalApparatusEntry"/>
988         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
989 ogy#CriticalApparatusGroup"/>
990         <rdfs:comment xml:lang="en">This property associates an instance of
991 Critical Apparatus Entry with an instance of Critical Apparatus Group.
992 Within a group of the critical apparatus there is at least one en-
993 try.</rdfs:comment>
994         <rdfs:label xml:lang="en">is part of critical apparatus
995 group</rdfs:label>
996     </owl:ObjectProperty>
997
998
999 <!-- http://purl.org/critical-edition-ontology#isPartOfIntroduction -->
1000
1001     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1002 ogy#isPartOfIntroduction">
1003         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1004 ontology#R5_is_component_of"/>
1005         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1006 ogy#IntroductionPart"/>
1007         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1008 ogy#Introduction"/>
1009         <rdfs:label xml:lang="en">is part of introduction</rdfs:label>
1010     </owl:ObjectProperty>
1011
1012
1013 <!-- http://purl.org/critical-edition-ontology#isPartOfReadingGroup -->
1014
1015     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1016 ogy#isPartOfReadingGroup">
1017         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1018 ontology#R5_is_component_of"/>
1019         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1020 ogy#ReadingInApparatus"/>

```

```

1021     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1022 ogy#ReadingGroup"/>
1023     <rdfs:comment xml:lang="en">This property associates an instance of
1024 Reading In Apparatus with an instance of Reading Group.</rdfs:comment>
1025     <rdfs:label xml:lang="en">is part of reading group</rdfs:label>
1026     </owl:ObjectProperty>
1027
1028
1029 <!-- http://purl.org/critical-edition-ontology#isPassageOf -->
1030
1031     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1032 ogy#isPassageOf">
1033         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1034 ontology#R15_is_fragment_of"/>
1035         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1036 ogy#EditedTextPassage"/>
1037         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1038 ogy#EditedText"/>
1039         <rdfs:label xml:lang="en">is passage of</rdfs:label>
1040     </owl:ObjectProperty>
1041
1042
1043 <!-- http://purl.org/critical-edition-ontology#isPreservedIn -->
1044
1045     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1046 ogy#isPreservedIn">
1047         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1048 ontology#R4_is_embodied_in"/>
1049         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
1050 tology#preserves"/>
1051         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1052 ogy#Witness"/>
1053         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1054 ogy#WitnessCarrier"/>
1055         <rdfs:label xml:lang="en">is preserved in</rdfs:label>
1056     </owl:ObjectProperty>
1057
1058
1059 <!-- http://purl.org/critical-edition-ontology#isPublishedIn -->
1060
1061     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1062 ogy#isPublishedIn">
1063         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1064 ontology#R4_is_embodied_in"/>
1065         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
1066 tology#publishes"/>
1067         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1068 ogy#ScholarlyEdition"/>
1069         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1070 ogy#ScholarlyEditionPublication"/>
1071         <rdfs:label xml:lang="en">is published in</rdfs:label>
1072     </owl:ObjectProperty>
1073
1074
1075 <!-- http://purl.org/critical-edition-ontology#isReadBy -->
1076
1077     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1078 ogy#isReadBy">
1079         <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
1080 tology#reads"/>

```

```

1081     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1082 ogy#ScholarlyEditionPublication"/>
1083     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1084 ogy#Reader"/>
1085     <rdfs:label xml:lang="en">is read by</rdfs:label>
1086     </owl:ObjectProperty>
1087
1088
1089 <!-- http://purl.org/critical-edition-ontology#isReadingOf -->
1090
1091     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1092 ogy#isReadingOf">
1093         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1094 ontology#R15_is_fragment_of"/>
1095         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1096 ogy#Reading"/>
1097         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1098 ogy#Witness"/>
1099         <rdfs:label xml:lang="en">is reading of</rdfs:label>
1100         </owl:ObjectProperty>
1101
1102
1103 <!--http://purl.org/critical-edition-ontology#isReferredToByApparatusEntry-
1104 ->
1105
1106     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1107 ogy#isReferredToByApparatusEntry">
1108         <rdfs:subPropertyOf rdf:resource="http://www.cidoc-crm.org/cidoc-
1109 crm/P67_is_referred_to_by"/>
1110         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1111 ogy#EditedTextPassage"/>
1112         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1113 ogy#ApparatusEntry"/>
1114         <rdfs:label xml:lang="en">is referred to by apparatus en-
1115 try</rdfs:label>
1116         </owl:ObjectProperty>
1117
1118
1119 <!-- http://purl.org/critical-edition-ontology#isReferredToByCriticalAppa-
1120 ratusEntry -->
1121
1122     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1123 ogy#isReferredToByCriticalApparatusEntry">
1124         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1125 ontology#isReferredToByApparatusEntry"/>
1126         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1127 ogy#CriticalTextPassage"/>
1128         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1129 ogy#CriticalApparatusEntry"/>
1130         <rdfs:label xml:lang="en">is referred to by critical apparatus en-
1131 try</rdfs:label>
1132         </owl:ObjectProperty>
1133
1134
1135 <!-- http://purl.org/critical-edition-ontology#isSelectorOfPassage -->
1136
1137     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1138 ogy#isSelectorOfPassage">
1139         <rdfs:subPropertyOf rdf:resource="http://www.cidoc-crm.org/cidoc-
1140 crm/P1i_identifies"/>

```

```

1141     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
1142 tology#passageHasSelector"/>
1143     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1144 ogy#PassageSelector"/>
1145     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1146 ogy#EditedTextPassage"/>
1147     <rdfs:label xml:lang="en">is selector of passage</rdfs:label>
1148 </owl:ObjectProperty>
1149
1150
1151 <!-- http://purl.org/critical-edition-ontology#isTransmittedIn -->
1152
1153     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1154 ogy#isTransmittedIn">
1155     <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1156 ontology#R3_is_realised_in"/>
1157     <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
1158 tology#transmits"/>
1159     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1160 ogy#F1_Work"/>
1161     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1162 ogy#Witness"/>
1163     <rdfs:label xml:lang="en">is transmitted in</rdfs:label>
1164 </owl:ObjectProperty>
1165
1166
1167 <!-- http://purl.org/critical-edition-ontology#passageHasSelector -->
1168
1169     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1170 ogy#passageHasSelector">
1171     <rdfs:subPropertyOf rdf:resource="http://www.cidoc-crm.org/cidoc-
1172 crm/P1_is_identified_by"/>
1173     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1174 ogy#EditedTextPassage"/>
1175     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1176 ogy#PassageSelector"/>
1177     <rdfs:comment xml:lang="en">This property associates a passage of
1178 the edited text, represented by an instance of Edited Text Passage, to the
1179 selector that identifies it, represented by an instance of Passage Selec-
1180 tor.</rdfs:comment>
1181     <rdfs:label xml:lang="en">passage has selector</rdfs:label>
1182 </owl:ObjectProperty>
1183
1184
1185 <!-- http://purl.org/critical-edition-ontology#performedBy -->
1186
1187     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1188 ogy#performedBy">
1189     <rdfs:subPropertyOf rdf:resource="http://www.w3.org/ns/prov#wasAs-
1190 sociatedWith"/>
1191     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1192 ogy#CriticalIntervention"/>
1193     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1194 ogy#Editor"/>
1195     <rdfs:comment xml:lang="en">This property associates an instance of
1196 Critical Intervention with an instance of Editor.</rdfs:comment>
1197     <rdfs:label xml:lang="en">performed by</rdfs:label>
1198 </owl:ObjectProperty>
1199
1200
1201

```



```

1202 <!-- http://purl.org/critical-edition-ontology#preparedBy -->
1203
1204     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1205 ogy#preparedBy">
1206         <rdfs:subPropertyOf rdf:resource="http://www.w3.org/ns/prov#wasGen-
1207 eratedBy"/>
1208         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1209 ogy#ReadingInApparatus"/>
1210         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1211 ogy#CriticalIntervention"/>
1212         <rdfs:comment xml:lang="en">This property associates an instance of
1213 Reading In Apparatus with an instance of Critical Intervention. With the
1214 same critical intervention, the editor prepares multiple readings to be
1215 presented in the critical apparatus.</rdfs:comment>
1216         <rdfs:label xml:lang="en">prepared by</rdfs:label>
1217     </owl:ObjectProperty>
1218
1219
1220 <!-- http://purl.org/critical-edition-ontology#preserves -->
1221
1222     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1223 ogy#preserves">
1224         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1225 ontology#R4_embodies"/>
1226         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1227 ogy#WitnessCarrier"/>
1228         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1229 ogy#Witness"/>
1230         <rdfs:comment xml:lang="en">This property associates a physical
1231 support, an instance of Witness Carrier, with the witness it preserves, an
1232 instance of Witness. Several witnesses can be preserved within the same
1233 physical carrier.</rdfs:comment>
1234         <rdfs:label xml:lang="en">preserves</rdfs:label>
1235     </owl:ObjectProperty>
1236
1237
1238 <!-- http://purl.org/critical-edition-ontology#publishes -->
1239
1240     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1241 ogy#publishes">
1242         <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1243 ontology#R4_embodies"/>
1244         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1245 ogy#ScholarlyEditionPublication"/>
1246         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1247 ogy#ScholarlyEdition"/>
1248         <rdfs:comment xml:lang="en">This property associates an instance of
1249 Scholarly Edition with the instance of Scholarly Edition Publication within
1250 which it is published. The same instance of Scholarly Edition can be pub-
1251 lished in different publications. At the same time, a publication can con-
1252 tain several scholarly editions.</rdfs:comment>
1253         <rdfs:label xml:lang="en">publishes</rdfs:label>
1254     </owl:ObjectProperty>
1255
1256
1257 <!-- http://purl.org/critical-edition-ontology#readingIsWitnessedBy -->
1258
1259     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1260 ogy#readingIsWitnessedBy">
1261         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1262 ogy#ReadingInApparatus"/>

```

```

1263     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1264 ogy#SiglumReference"/>
1265     <rdfs:comment xml:lang="en">This property associates an instance of
1266 Reading In Apparatus or Base Reading In Apparatus with one or more in-
1267 stances of Siglum Reference.</rdfs:comment>
1268     <rdfs:label xml:lang="en">reading is witnessed by</rdfs:label>
1269     </owl:ObjectProperty>
1270
1271
1272 <!-- http://purl.org/critical-edition-ontology#reads -->
1273
1274     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1275 ogy#reads">
1276         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1277 ogy#Reader"/>
1278         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1279 ogy#ScholarlyEditionPublication"/>
1280         <rdfs:comment xml:lang="en">This property associates an instance of
1281 Reader with the instance of Scholarly Edition Publication they
1282 read.</rdfs:comment>
1283         <rdfs:label xml:lang="en">reads</rdfs:label>
1284         </owl:ObjectProperty>
1285
1286
1287 <!-- http://purl.org/critical-edition-ontology#refersToSiglum -->
1288
1289     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1290 ogy#refersToSiglum">
1291         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1292 ogy#SiglumReference"/>
1293         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1294 ogy#Siglum"/>
1295         <rdfs:comment xml:lang="en">This property associates the siglum of
1296 a witness, an instance of Siglum, to its references contained within the
1297 entries of the critical apparatus, represented by instances of Siglum Ref-
1298 erence.</rdfs:comment>
1299         <rdfs:label xml:lang="en">refers to siglum</rdfs:label>
1300         </owl:ObjectProperty>
1301
1302
1303 <!-- http://purl.org/critical-edition-ontology#selectorIsShownIn -->
1304
1305     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1306 ogy#selectorIsShownIn">
1307         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1308 ogy#PassageSelector"/>
1309         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1310 ogy#ApparatusEntryContent"/>
1311         <rdfs:comment xml:lang="en">This property associates the selector
1312 of a passage of the edited text, an instance of Passage Selector, with the
1313 content of an apparatus entry, an instance of Apparatus Entry Content,
1314 within which it is shown.</rdfs:comment>
1315         <rdfs:label xml:lang="en">selector is shown in</rdfs:label>
1316         </owl:ObjectProperty>
1317
1318
1319 <!-- http://purl.org/critical-edition-ontology#setBy -->
1320
1321     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1322 ogy#setBy">

```

```

1323     <rdfs:subPropertyOf rdf:resource="http://www.w3.org/ns/prov#wasGen-
1324 eratedBy"/>
1325     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1326 ogy#CriticalTextPassage"/>
1327     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1328 ogy#CriticalIntervention"/>
1329     <rdfs:comment xml:lang="en">This property associates an instance of
1330 Critical Text Passage with an instance of Critical Intervention. With a
1331 critical intervention the editor establishes only one lesson to be included
1332 in the critical text.</rdfs:comment>
1333     <rdfs:label xml:lang="en">set by</rdfs:label>
1334 </owl:ObjectProperty>
1335
1336 <!-- http://purl.org/critical-edition-ontology#siglumIdentifies -->
1337
1338 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1339 ogy#siglumIdentifies">
1340 <owl:inverseOf rdf:resource="http://purl.org/critical-edition-on-
1341 tology#witnessIsIdentifiedBy"/>
1342 <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1343 ogy#Siglum"/>
1344 <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1345 ogy#Witness"/>
1346 <rdfs:comment xml:lang="en">This property associates an instance of
1347 Witness with the instance of Siglum which identifies it within the critical
1348 edition.</rdfs:comment>
1349 <rdfs:label xml:lang="en">siglum identifies</rdfs:label>
1350 </owl:ObjectProperty>
1351
1352
1353 <!-- http://purl.org/critical-edition-ontology#transmits -->
1354
1355 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1356 ogy#transmits">
1357 <rdfs:subPropertyOf rdf:resource="http://purl.org/critical-edition-
1358 ontology#R3_realises"/>
1359 <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1360 ogy#Witness"/>
1361 <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1362 ogy#F1_Work"/>
1363 <rdfs:comment xml:lang="en">This property associates an instance of
1364 Witness with the instance of F1 Work it transmits. An instance of F1 Work
1365 is handed down by at least one witness and a variable number of witnesses
1366 can hand down the same work.</rdfs:comment>
1367 <rdfs:label xml:lang="en">transmits</rdfs:label>
1368 </owl:ObjectProperty>
1369
1370
1371 <!-- http://purl.org/critical-edition-ontology#wasCreatedBy -->
1372
1373 <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1374 ogy#wasCreatedBy">
1375 <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1376 ogy#ScholarlyEdition"/>
1377 <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1378 ogy#Editor"/>
1379 <rdfs:label xml:lang="en">was created by</rdfs:label>
1380 </owl:ObjectProperty>
1381
1382
1383

```

```

1384
1385     <!-- http://purl.org/critical-edition-ontology#wasRevisionOf -->
1386
1387     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1388 ogy#wasRevisionOf">
1389         <owl:equivalentProperty rdf:re-
1390 source="http://www.w3.org/ns/prov#wasRevisionOf"/>
1391         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1392 ogy#ReadingInApparatus"/>
1393         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1394 ogy#ReadingInApparatus"/>
1395         <rdfs:comment xml:lang="en"></rdfs:comment>
1396         <rdfs:label xml:lang="en"></rdfs:label>
1397     </owl:ObjectProperty>
1398
1399
1400     <!-- http://purl.org/critical-edition-ontology#witnessIsDescribedBy -->
1401
1402     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1403 ogy#witnessIsDescribedBy">
1404         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1405 ogy#Witness"/>
1406         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1407 ogy#WitnessDescription"/>
1408         <rdfs:comment xml:lang="en">This property associates an instance of
1409 Witness with the instance of Witness Description in which it is de-
1410 scribed.</rdfs:comment>
1411         <rdfs:label xml:lang="en">witness is described by</rdfs:label>
1412     </owl:ObjectProperty>
1413
1414
1415     <!-- http://purl.org/critical-edition-ontology#witnessIsIdentifiedBy -->
1416
1417     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1418 ogy#witnessIsIdentifiedBy">
1419         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1420 ogy#Witness"/>
1421         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1422 ogy#Siglum"/>
1423         <rdfs:label xml:lang="en">witness is identified by</rdfs:label>
1424     </owl:ObjectProperty>
1425
1426
1427     <!-- http://purl.org/critical-edition-ontology#witnessIsPresentedIn -->
1428
1429     <owl:ObjectProperty rdf:about="http://purl.org/critical-edition-ontol-
1430 ogy#witnessIsPresentedIn">
1431         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1432 ogy#Witness"/>
1433         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1434 ogy#WitnessesList"/>
1435         <rdfs:comment xml:lang="en">This property associates an instance of
1436 Witness with the instance of Witnesses List in which it is pre-
1437 sented.</rdfs:comment>
1438         <rdfs:label xml:lang="en">witness is presented in</rdfs:label>
1439     </owl:ObjectProperty>
1440
1441
1442     <!-- http://purl.org/emmedi/hico#hasInterpretationCriterion -->
1443

```

```

1444     <owl:ObjectProperty rdf:about="http://purl.org/emmedi/hico#hasInterpre-
1445 tationCriterion"/>
1446
1447
1448     <!-- http://purl.org/emmedi/hico#hasInterpretationType -->
1449
1450     <owl:ObjectProperty rdf:about="http://purl.org/emmedi/hico#hasInterpre-
1451 tationType"/>
1452
1453
1454     <!-- http://purl.org/emmedi/hico#isExtractedFrom -->
1455
1456     <owl:ObjectProperty rdf:about="http://purl.org/emmedi/hico#isExtract-
1457 edFrom"/>
1458
1459
1460 <!--http://www.cidoc-crm.org/cidoc-crm/P107_has_current_or_former_member-->
1461
1462     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1463 crm/P107_has_current_or_former_member">
1464         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1465 ogy#EditorsGroup"/>
1466         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1467 ogy#Editor"/>
1468     </owl:ObjectProperty>
1469
1470
1471 <!--http://www.cidoc-crm.org/cidoc-crm/P107_is_current_or_former_member -->
1472
1473     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1474 crm/P107_is_current_or_former_member">
1475         <owl:inverseOf rdf:resource="http://www.cidoc-crm.org/cidoc-
1476 crm/P107_has_current_or_former_member"/>
1477         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1478 ogy#Editor"/>
1479         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1480 ogy#EditorsGroup"/>
1481     </owl:ObjectProperty>
1482
1483
1484     <!-- http://www.cidoc-crm.org/cidoc-crm/P1_is_identified_by -->
1485
1486     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1487 crm/P1_is_identified_by"/>
1488
1489
1490     <!-- http://www.cidoc-crm.org/cidoc-crm/P1i_identifies -->
1491
1492     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1493 crm/P1i_identifies">
1494         <owl:inverseOf rdf:resource="http://www.cidoc-crm.org/cidoc-
1495 crm/P1_is_identified_by"/>
1496     </owl:ObjectProperty>
1497
1498
1499     <!-- http://www.cidoc-crm.org/cidoc-crm/P3_has_note -->
1500
1501     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1502 crm/P3_has_note"/>
1503
1504

```

```

1505     <!-- http://www.cidoc-crm.org/cidoc-crm/P67_is_referred_to_by -->
1506
1507     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1508     crm/P67_is_referred_to_by"/>
1509
1510
1511     <!-- http://www.cidoc-crm.org/cidoc-crm/P67_refers_to -->
1512
1513     <owl:ObjectProperty rdf:about="http://www.cidoc-crm.org/cidoc-
1514     crm/P67_refers_to">
1515         <owl:inverseOf rdf:resource="http://www.cidoc-crm.org/cidoc-
1516     crm/P67_is_referred_to_by"/>
1517     </owl:ObjectProperty>
1518
1519
1520     <!-- http://www.w3.org/ns/oa#hasBody -->
1521
1522     <owl:ObjectProperty rdf:about="http://www.w3.org/ns/oa#hasBody">
1523         <rdfs:label xml:lang="en">has body</rdfs:label>
1524         <rdfs:comment xml:lang="en">The object of the relationship is a re-
1525     source that is a body of the Annotation.</rdfs:comment>
1526     </owl:ObjectProperty>
1527
1528
1529     <!-- http://www.w3.org/ns/prov#wasAssociatedWith -->
1530
1531     <owl:ObjectProperty rdf:about="http://www.w3.org/ns/prov#wasAssociat-
1532     edWith"/>
1533
1534
1535     <!-- http://www.w3.org/ns/prov#wasDerivedFrom -->
1536
1537     <owl:ObjectProperty
1538     rdf:about="http://www.w3.org/ns/prov#wasDerivedFrom"/>
1539
1540
1541     <!-- http://www.w3.org/ns/prov#wasGeneratedBy -->
1542
1543     <owl:ObjectProperty rdf:about="http://www.w3.org/ns/prov#wasGenerat-
1544     edBy"/>
1545
1546
1547     <!-- http://www.w3.org/ns/prov#wasRevisionOf -->
1548
1549     <owl:ObjectProperty rdf:about="http://www.w3.org/ns/prov#wasRevi-
1550     sionOf"/>
1551
1552
1553     <!-- https://w3id.org/cao#hasAttributedHand -->
1554
1555     <owl:ObjectProperty rdf:about="https://w3id.org/cao#hasAttributedHand">
1556         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1557     ogy#ReadingInApparatus"/>
1558         <rdfs:range rdf:resource="http://www.w3.org/ns/prov#Agent"/>
1559     </owl:ObjectProperty>
1560
1561
1562     <!-- https://w3id.org/cao#hasBaseReading -->
1563
1564     <owl:ObjectProperty rdf:about="https://w3id.org/cao#hasBaseReading">

```

```

1565     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1566 ogy#CriticalApparatusEntryContent"/>
1567     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1568 ogy#ReadingInApparatusCause"/>
1569     </owl:ObjectProperty>
1570
1571
1572     <!-- https://w3id.org/cao#hasReading -->
1573
1574     <owl:ObjectProperty rdf:about="https://w3id.org/cao#hasReading">
1575         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1576 ogy#CriticalApparatusEntryContent"/>
1577         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1578 ogy#ReadingInApparatus"/>
1579     </owl:ObjectProperty>
1580
1581
1582     <!-- https://w3id.org/cao#hasReadingCause -->
1583
1584     <owl:ObjectProperty rdf:about="https://w3id.org/cao#hasReadingCause">
1585         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1586 ogy#ReadingInApparatus"/>
1587         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1588 ogy#ReadingInApparatusCause"/>
1589     </owl:ObjectProperty>
1590
1591
1592     <!-- https://w3id.org/cao#hasReadingType -->
1593
1594     <owl:ObjectProperty rdf:about="https://w3id.org/cao#hasReadingType">
1595         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1596 ogy#ReadingInApparatus"/>
1597         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1598 ogy#ReadingInApparatusType"/>
1599     </owl:ObjectProperty>
1600
1601
1602     <!-- https://w3id.org/cao#hasVariant -->
1603
1604     <owl:ObjectProperty rdf:about="https://w3id.org/cao#hasVariant">
1605         <owl:inverseOf rdf:resource="https://w3id.org/cao#isVariantOf"/>
1606         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1607 ogy#BaseReadingInApparatus"/>
1608         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1609 ogy#ReadingInApparatus"/>
1610     </owl:ObjectProperty>
1611
1612
1613     <!-- https://w3id.org/cao#isVariantOf -->
1614
1615     <owl:ObjectProperty rdf:about="https://w3id.org/cao#isVariantOf">
1616         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1617 ogy#ReadingInApparatus"/>
1618         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1619 ogy#BaseReadingInApparatus"/>
1620     </owl:ObjectProperty>
1621
1622
1623     <!-- https://w3id.org/cao#wasCorrectedTo -->
1624
1625     <owl:ObjectProperty rdf:about="https://w3id.org/cao#wasCorrectedTo">

```

```

1626     <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1627 ogy#ReadingInApparatus"/>
1628     <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1629 ogy#ReadingInApparatus"/>
1630     </owl:ObjectProperty>
1631
1632
1633     <!-- https://w3id.org/cao#wasDerivedFrom -->
1634
1635     <owl:ObjectProperty rdf:about="https://w3id.org/cao#wasDerivedFrom">
1636         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1637 ogy#ReadingInApparatus"/>
1638         <rdfs:range rdf:resource="http://purl.org/critical-edition-ontol-
1639 ogy#ReadingInApparatus"/>
1640     </owl:ObjectProperty>
1641
1642
1643     <!--//////////////////////////////////////
1644     //
1645     // Data properties
1646     //
1647     ////////////////////////////////////////-->
1648
1649
1650
1651     <!-- http://purl.org/critical-edition-ontology#isNegative -->
1652
1653     <owl:DatatypeProperty rdf:about="http://purl.org/critical-edition-on-
1654 tology#isNegative">
1655         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1656 ogy#CriticalApparatus"/>
1657     </owl:DatatypeProperty>
1658
1659
1660     <!-- http://purl.org/critical-edition-ontology#isPositive -->
1661
1662     <owl:DatatypeProperty rdf:about="http://purl.org/critical-edition-on-
1663 tology#isPositive">
1664         <rdfs:domain rdf:resource="http://purl.org/critical-edition-ontol-
1665 ogy#CriticalApparatus"/>
1666     </owl:DatatypeProperty>
1667
1668
1669     <!-- ////////////////////////////////////////
1670     //
1671     // Classes
1672     //
1673     //////////////////////////////////////// -->
1674
1675
1676
1677     <!-- http://purl.org/critical-edition-ontology#AccompanyingMaterial -->
1678
1679     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Accom-
1680 panyingMaterial">
1681         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1682 tology#ScholarlyEditionComponent"/>
1683         <rdfs:comment xml:lang="en">Component of a scholarly edition of a
1684 textual or other nature that accompanies the edited text and is subordinate
1685 to it.</rdfs:comment>
1686         <rdfs:label xml:lang="en">Accompanying Material</rdfs:label>

```



```

1687     </owl:Class>
1688
1689
1690     <!-- https://www.w3.org/ns/oa#Annotation -->
1691
1692     <owl:Class rdf:about="https://www.w3.org/ns/oa#Annotation">
1693         <owl:equivalentClass rdf:resource="http://purl.org/critical-edi-
1694 tion-ontology#ApparatusEntry"/>
1695         <rdfs:comment xml:lang="en">The class for Web Annota-
1696 tions.</rdfs:comment>
1697         <rdfs:label xml:lang="en">Annotation</rdfs:label>
1698     </owl:Class>
1699
1700
1701     <!-- http://purl.org/critical-edition-ontology#Apparatus -->
1702
1703     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Appa-
1704 ratus">
1705         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1706 tology#AccompanyingMaterial"/>
1707         <rdfs:comment xml:lang="en">Component of a scholarly edition that
1708 accompanies the edited text and consists of a set of annotations that the
1709 editor has appended to edited text for a particular scientific pur-
1710 pose.</rdfs:comment>
1711         <rdfs:label xml:lang="en">Apparatus</rdfs:label>
1712     </owl:Class>
1713
1714
1715     <!-- http://purl.org/critical-edition-ontology#ApparatusEntry -->
1716
1717     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Appa-
1718 ratusEntry">
1719         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1720 tology#AccompanyingMaterial"/>
1721         <rdfs:comment xml:lang="en">An annotation presented within an appa-
1722 ratus.</rdfs:comment>
1723         <rdfs:label xml:lang="en">Apparatus Entry</rdfs:label>
1724     </owl:Class>
1725
1726
1727     <!-- http://purl.org/critical-edition-ontology#ApparatusEntryContent --
1728 >
1729
1730     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Appa-
1731 ratusEntryContent">
1732         <owl:equivalentClass rdf:resource="http://www.w3.org/ns/oa#Textual-
1733 Body"/>
1734         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1735 tology#AccompanyingMaterial"/>
1736         <rdfs:comment xml:lang="en">Textual content of an apparatus en-
1737 try.</rdfs:comment>
1738         <rdfs:label xml:lang="en">Apparatus Entry Content</rdfs:label>
1739     </owl:Class>
1740
1741
1742     <!-- http://purl.org/critical-edition-ontology#BaseReadingInApparatus -
1743 ->
1744
1745     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Bas-
1746 eReadingInApparatus">

```

```

1747     <owl:equivalentClass rdf:resource="https://w3id.org/cao#BaseRead-
1748     ing"/>
1749     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1750     tology#ReadingInApparatus"/>
1751     <rdfs:comment xml:lang="en">The reading chosen or proposed by the
1752     editor, presented within an entry of the critical apparatus for the dual
1753     purpose of indicating the passage in the text to which the entry refers and
1754     to allow the reader to compare it more easily with the other recorded read-
1755     ings.</rdfs:comment>
1756     <rdfs:label xml:lang="en">Base Reading In Apparatus</rdfs:label>
1757     </owl:Class>
1758
1759
1760     <!-- http://purl.org/critical-edition-ontology#BaseWitness -->
1761
1762     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Base-
1763     Witness">
1764     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1765     tology#Witness"/>
1766     <rdfs:comment xml:lang="en">Witness used as a basis to establish
1767     the critical text.</rdfs:comment>
1768     <rdfs:label xml:lang="en">Base Witness</rdfs:label>
1769     </owl:Class>
1770
1771
1772     <!-- http://purl.org/critical-edition-ontology#Commentary -->
1773
1774     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Commen-
1775     tary">
1776     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1777     tology#Apparatus"/>
1778     <rdfs:comment xml:lang="en">A type of apparatus that the editor ap-
1779     pends to the critical text in order to offer more information on the tex-
1780     tual transmission and to argue their editorial choices.</rdfs:comment>
1781     <rdfs:label xml:lang="en">Commentary</rdfs:label>
1782     </owl:Class>
1783
1784
1785     <!-- http://purl.org/critical-edition-ontology#CriticalApparatus -->
1786
1787     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Critic-
1788     alApparatus">
1789     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1790     tology#Apparatus"/>
1791     <rdfs:comment xml:lang="en">An apparatus through which the editor
1792     shows the differences between the critical text and the textual transmis-
1793     sion and possibly explains their choices.</rdfs:comment>
1794     <rdfs:label xml:lang="en">Critical Apparatus</rdfs:label>
1795     </owl:Class>
1796
1797
1798     <!-- http://purl.org/critical-edition-ontology#CriticalApparatusEntry -
1799     ->
1800
1801     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Critic-
1802     alApparatusEntry">
1803     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1804     tology#ApparatusEntry"/>
1805     <rdfs:comment xml:lang="en">An annotation to the critical text that
1806     is part of the critical apparatus, within which the editor presents the

```

```

1807 variants and, more generally, the readings attested in the textual tradi-
1808 tion, comparing them with the relative passage in the critical
1809 text.</rdfs:comment>
1810     <rdfs:label xml:lang="en">Critical Apparatus Entry</rdfs:label>
1811     </owl:Class>
1812
1813
1814     <!-- http://purl.org/critical-edition-ontology#CriticalApparatusEntry-
1815 Content -->
1816
1817     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Crimi-
1818 calApparatusEntryContent">
1819         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1820 tology#ApparatusEntryContent"/>
1821         <rdfs:comment xml:lang="en">The content of a critical apparatus en-
1822 try.</rdfs:comment>
1823         <rdfs:label xml:lang="en">Critical Apparatus Entry Con-
1824 tent</rdfs:label>
1825     </owl:Class>
1826
1827
1828 <!--http://purl.org/critical-edition-ontology#CriticalApparatusEntryNote-->
1829
1830     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Crimi-
1831 calApparatusEntryNote">
1832         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1833 tology#CriticalApparatusEntryContent"/>
1834         <rdfs:comment xml:lang="en">Note that the editor inserts in an en-
1835 try of the critical apparatus to specify some philological aspects of a
1836 reading, of a group of readings, of the chosen or proposed reading or, in
1837 general, of the textual tradition at that point of the text.</rdfs:comment>
1838         <rdfs:label xml:lang="en">Critical Apparatus Entry Note</rdfs:la-
1839 bel>
1840     </owl:Class>
1841
1842
1843     <!-- http://purl.org/critical-edition-ontology#CriticalApparatusGroup -->
1844
1845     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Crimi-
1846 calApparatusGroup">
1847         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1848 tology#AccompanyingMaterial"/>
1849         <rdfs:comment xml:lang="en">Structure in which the critical appa-
1850 ratus can be divided, in order to distinguish the different types of infor-
1851 mation it contains and facilitate its reading. A critical apparatus group
1852 may contain a particular type of readings (for example substantive variants
1853 or orthographic variants) or it may group together the information about
1854 the variation in a given set of witnesses (for example a family of wit-
1855 nesses).</rdfs:comment>
1856         <rdfs:label xml:lang="en">Critical Apparatus Group</rdfs:label>
1857     </owl:Class>
1858
1859
1860     <!-- http://purl.org/critical-edition-ontology#CriticalEdition -->
1861
1862     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Crimi-
1863 calEdition">
1864         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1865 tology#ScholarlyEdition"/>
1866         <rdfs:comment xml:lang="en">A type of scholarly edition, created by

```

```

1867 applying the principles of textual criticism, with the aim of reconstruct-
1868 ing the original form or more original forms of the text of a
1869 work.</rdfs:comment>
1870     <rdfs:label xml:lang="en">Critical Edition</rdfs:label>
1871     </owl:Class>
1872
1873
1874     <!-- http://purl.org/critical-edition-ontology#CriticalIntervention -->
1875
1876     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Critic-
1877 calIntervention">
1878         <rdfs:subClassOf rdf:resource="http://purl.org/emmedi/hico#Inter-
1879 pretationAct"/>
1880         <rdfs:subClassOf>
1881             <owl:Restriction>
1882                 <owl:onProperty rdf:resource="http://purl.org/critical-edi-
1883 tion-ontology#isCarriedOutOn"/>
1884                 <owl:minQualifiedCardinality
1885 rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInte-
1886 ger">1</owl:minQualifiedCardinality>
1887                 <owl:onClass rdf:resource="http://purl.org/critical-edi-
1888 tion-ontology#Reading"/>
1889             </owl:Restriction>
1890         </rdfs:subClassOf>
1891         <rdfs:comment xml:lang="en">During the preparation of the critical
1892 text, it is the interpretative act through which the editor establishes or
1893 reconstructs the portion of text to be included in the critical text and
1894 the readings to be presented in the critical apparatus, starting from the
1895 readings present in the textual transmission.</rdfs:comment>
1896         <rdfs:label xml:lang="en">Critical Intervention</rdfs:label>
1897     </owl:Class>
1898
1899
1900     <!-- http://purl.org/critical-edition-ontology#CriticalIntervention-
1901 Criterion -->
1902
1903     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Critic-
1904 calInterventionCriterion">
1905         <rdfs:subClassOf rdf:resource="http://purl.org/emmedi/hico#Inter-
1906 pretationCriterion"/>
1907         <rdfs:comment xml:lang="en">The criterion or more broadly the rea-
1908 son why the editor has deemed a critical intervention necessary.</rdfs:com-
1909 ment>
1910         <rdfs:label xml:lang="en">Critical Intervention Criterion</rdfs:la-
1911 bel>
1912     </owl:Class>
1913
1914
1915     <!-- http://purl.org/critical-edition-ontology#CriticalInterventionType -->
1916
1917     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Critic-
1918 calInterventionType">
1919         <rdfs:subClassOf rdf:resource="http://purl.org/emmedi/hico#Inter-
1920 pretationType"/>
1921         <rdfs:comment xml:lang="en">The type of critical intervention made
1922 by the editor, for example: the correction of an evident error, the selec-
1923 tion of a variant (selectio), the conjecture (divinatio), etc.</rdfs:com-
1924 ment>
1925         <rdfs:label xml:lang="en">Critical Intervention Type</rdfs:label>
1926     </owl:Class>
1927

```

```

1928
1929     <!-- http://purl.org/critical-edition-ontology#CriticalText -->
1930
1931     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#CriticalText">
1932     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-ontology#EditedText"/>
1933     <rdfs:comment xml:lang="en">A type of edited text presented in a
1934     critical edition and edited for the purpose of reconstructing the original
1935     form or one of the original forms of the text of a work.</rdfs:comment>
1936     <rdfs:label xml:lang="en">Critical Text</rdfs:label>
1937     </owl:Class>
1938
1939
1940
1941     <!-- http://purl.org/critical-edition-ontology#CriticalTextPassage -->
1942
1943     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#CriticalTextPassage">
1944     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-ontology#EditedTextPassage"/>
1945     <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-crm/E90_Symbolic_Object"/>
1946     <rdfs:comment xml:lang="en">A portion of variable size of the critical text, made up of characters, words or sentences.</rdfs:comment>
1947     <rdfs:label xml:lang="en">Critical Text Passage</rdfs:label>
1948     </owl:Class>
1949
1950
1951
1952     <!-- http://purl.org/critical-edition-ontology#EditedText -->
1953
1954     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#EditedText">
1955     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-ontology#ScholarlyEditionComponent"/>
1956     <rdfs:comment xml:lang="en">Text of a work prepared by an editor or a group of editors within a scholarly edition.</rdfs:comment>
1957     <rdfs:label xml:lang="en">Edited Text</rdfs:label>
1958     </owl:Class>
1959
1960
1961
1962     <!-- http://purl.org/critical-edition-ontology#EditedTextPassage -->
1963
1964     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#EditedTextPassage">
1965     <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-crm/E90_Symbolic_Object"/>
1966     <rdfs:comment xml:lang="en">A portion of variable size of the edited text, made up of characters, words or sentences.</rdfs:comment>
1967     <rdfs:label xml:lang="en">Edited Text Passage</rdfs:label>
1968     </owl:Class>
1969
1970
1971
1972     <!-- http://purl.org/critical-edition-ontology#Editor -->
1973
1974     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Editor">
1975     <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-crm/E39_Actor"/>
1976     <rdfs:comment xml:lang="en">Person that created a scholarly edition. The editor must possess philological skills and textual criticism skills. The identity of the editor must be known, in order to make their
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988

```

```

1989 work citable.</rdfs:comment>
1990     <rdfs:label xml:lang="en">Editor</rdfs:label>
1991     </owl:Class>
1992
1993
1994     <!-- http://purl.org/critical-edition-ontology#EditorsGroup -->
1995
1996     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Edi-
1997 torsGroup">
1998         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
1999 tology#Editor"/>
2000         <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-
2001 crm/E74_Group"/>
2002         <rdfs:comment xml:lang="en">A group of editors that collaboratively
2003 create a scholarly edition.</rdfs:comment>
2004         <rdfs:label xml:lang="en">Editors Group</rdfs:label>
2005     </owl:Class>
2006
2007
2008     <!-- http://purl.org/critical-edition-ontology#ExternalCriticalInter-
2009 ventionCriterion -->
2010
2011     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Exter-
2012 nalCriticalInterventionCriterion">
2013         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2014 tology#CriticalInterventionCriterion"/>
2015         <rdfs:comment xml:lang="en">A criterion for carrying out the criti-
2016 cal intervention based on the genealogy of the textual transmis-
2017 sion.</rdfs:comment>
2018         <rdfs:label xml:lang="en">External Critical Intervention Crite-
2019 rion</rdfs:label>
2020     </owl:Class>
2021
2022
2023     <!-- http://purl.org/critical-edition-ontology#F1_Work -->
2024
2025     <owl:Class rdf:about="http://purl.org/critical-edition-ontol-
2026 ogy#F1_Work">
2027         <rdfs:comment xml:lang="en">This class comprises distinct intellec-
2028 tual ideas conveyed in artistic and intellectual creations, such a poems,
2029 stories or musical compositions.</rdfs:comment>
2030         <rdfs:label xml:lang="en">F1 Work</rdfs:label>
2031     </owl:Class>
2032
2033
2034     <!-- http://purl.org/critical-edition-ontology#F2_Expression -->
2035
2036     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#F2_Ex-
2037 pression">
2038         <rdfs:comment xml:lang="en">This class comprises the intellectual
2039 or artistic realisations of Works in the form of identifiable immaterial
2040 objects, such as texts, poems, jokes, musical or choreographic notations,
2041 movement pattern, sound pattern, images, multimedia objects, or any combi-
2042 nation of such forms. The substance of F2 Expression is signs.</rdfs:com-
2043 ment>
2044         <rdfs:label xml:lang="en">F2 Expression</rdfs:label>
2045     </owl:Class>
2046
2047
2048     <!-- http://purl.org/critical-edition-ontology#F3_Manifestation -->
2049

```

```

2050     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#F3_Man-
2051 ifestation">
2052     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2053 tology#F2_Expression"/>
2054     <rdfs:comment xml:lang="en">This class comprises products rendering
2055 one or more Expressions. A Manifestation is defined by both the overall
2056 content and the form of its presentation. The substance of F3 Manifestation
2057 is not only signs, but also the manner in which they are presented to be
2058 consumed by users, including the kind of media adopted.</rdfs:comment>
2059     <rdfs:label xml:lang="en">F3 Manifestation</rdfs:label>
2060 </owl:Class>
2061
2062
2063 <!-- http://purl.org/critical-edition-ontology#Gap -->
2064
2065 <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Gap">
2066 <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2067 tology#Reading"/>
2068 <rdfs:comment xml:lang="en">Missing passage within a witness. It
2069 may be due to a voluntary or involuntary omission on the part of the copy-
2070 ist, a mechanical gap, an erasure or the state of conservation of the wit-
2071 ness.</rdfs:comment>
2072 <rdfs:label xml:lang="en">Gap</rdfs:label>
2073 </owl:Class>
2074
2075
2076 <!-- http://purl.org/critical-edition-ontology#InternalCriticalInter-
2077 ventionCriterion -->
2078
2079 <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Inter-
2080 nalCriticalInterventionCriterion">
2081 <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2082 tology#CriticalInterventionCriterion"/>
2083 <rdfs:comment xml:lang="en">A criterion for carrying out the criti-
2084 cal intervention based on the intrinsic plausibility of the vari-
2085 ants.</rdfs:comment>
2086 <rdfs:label xml:lang="en">Internal Critical Intervention Crite-
2087 rion</rdfs:label>
2088 </owl:Class>
2089
2090
2091 <!-- http://purl.org/critical-edition-ontology#Introduction -->
2092
2093 <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Intro-
2094 duction">
2095 <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2096 tology#AccompanyingMaterial"/>
2097 <rdfs:comment xml:lang="en">Text of variable size which introduces
2098 the edited text within a scholarly edition, presenting the work itself and
2099 the work carried out by the editor. The introduction may be organized in
2100 different parts and may contain lists, pictures, and other materi-
2101 als.</rdfs:comment>
2102 <rdfs:label xml:lang="en">Introduction</rdfs:label>
2103 </owl:Class>
2104
2105
2106 <!-- http://purl.org/critical-edition-ontology#IntroductionPart -->
2107
2108 <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Intro-
2109 ductionPart">

```

```

2110         <rdfs:comment xml:lang="en">A part of the introduction.</rdfs:com-
2111 ment>
2112         <rdfs:label xml:lang="en">Introduction Part</rdfs:label>
2113     </owl:Class>
2114
2115
2116     <!-- http://purl.org/critical-edition-ontology#Note -->
2117
2118     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Note">
2119         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2120 tology#ApparatusEntry"/>
2121         <rdfs:comment xml:lang="en">An annotation to the critical text,
2122 which is part of the commentary.</rdfs:comment>
2123         <rdfs:label xml:lang="en">Note</rdfs:label>
2124     </owl:Class>
2125
2126
2127     <!-- http://purl.org/critical-edition-ontology#NoteContent -->
2128
2129     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Note-
2130 Content">
2131         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2132 tology#ApparatusEntryContent"/>
2133         <rdfs:comment xml:lang="en">Content of a note.</rdfs:comment>
2134         <rdfs:label xml:lang="en">Note Content</rdfs:label>
2135     </owl:Class>
2136
2137
2138     <!-- http://purl.org/critical-edition-ontology#PassagePositionSelector
2139 -->
2140
2141     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Pas-
2142 sagePositionSelector">
2143         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2144 tology#PassageSelector"/>
2145         <rdfs:comment xml:lang="en">Type of selector of a passage of the
2146 critical text which indicates where the passage is positioned, usually
2147 through the number of the structural element in which the passage is con-
2148 tained (for example the verse, the line, the paragraph, etc.) or through
2149 the cross reference to an alphanumeric exponent or other graphic
2150 sign.</rdfs:comment>
2151         <rdfs:label xml:lang="en">Passage Position Selector</rdfs:label>
2152     </owl:Class>
2153
2154
2155     <!-- http://purl.org/critical-edition-ontology#PassageQuoteSelector -->
2156
2157     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Pas-
2158 sageQuoteSelector">
2159         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2160 tology#PassageSelector"/>
2161         <rdfs:comment xml:lang="en">Type of selector of a passage of the
2162 critical text which indicates the passage by copying it in its entirety or
2163 in abbreviated form.</rdfs:comment>
2164         <rdfs:label xml:lang="en">Passage Quote Selector</rdfs:label>
2165     </owl:Class>
2166
2167
2168     <!-- http://purl.org/critical-edition-ontology#PassageSelector -->
2169

```



```

2170     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Passag-
2171 eSelector">
2172         <owl:equivalentClass rdf:resource="http://www.w3.org/ns/oa#Selec-
2173 tor"/>
2174         <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-
2175 crm/E41_Appellation"/>
2176         <rdfs:comment xml:lang="en">Sequence of characters that allows to
2177 identify exactly a passage within a text. There are several types of se-
2178 lector. Often it is a reference to the location of the passage, for example
2179 the line or verse number, or a copy of the passage concerned.</rdfs:com-
2180 ment>
2181         <rdfs:label xml:lang="en">Passage Selector</rdfs:label>
2182     </owl:Class>
2183
2184     <!-- http://purl.org/critical-edition-ontology#Reader -->
2185
2186     <owl:Class rdf:about="http://purl.org/critical-edition-ontol-
2187 ogy#Reader">
2188         <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-
2189 crm/E21_Person"/>
2190         <rdfs:comment xml:lang="en">Person that reads the scholarly edi-
2191 tion.
2192         Note: the term "reads" was chosen to sum up the different
2193 ways in which a critical edition can be used.
2194         </rdfs:comment>
2195         <rdfs:label xml:lang="en">Reader</rdfs:label>
2196     </owl:Class>
2197
2198
2199     <!-- http://purl.org/critical-edition-ontology#Reading -->
2200
2201     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Read-
2202 ing">
2203         <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-
2204 crm/E90_Symbolic_Object"/>
2205         <rdfs:subClassOf>
2206             <owl:Restriction>
2207                 <owl:onProperty rdf:resource="http://purl.org/critical-edi-
2208 tion-ontology#isReadingOf"/>
2209                 <owl:qualifiedCardinality
2210 rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInte-
2211 ger">1</owl:qualifiedCardinality>
2212                 <owl:onClass rdf:resource="http://purl.org/critical-edi-
2213 tion-ontology#Witness"/>
2214             </owl:Restriction>
2215         </rdfs:subClassOf>
2216         <rdfs:comment xml:lang="en">A sequence of characters, words, or
2217 phrases that can be read in a preserved or lost witness.</rdfs:comment>
2218         <rdfs:label xml:lang="en">Reading</rdfs:label>
2219     </owl:Class>
2220
2221
2222     <!-- http://purl.org/critical-edition-ontology#ReadingGroup -->
2223
2224     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Read-
2225 ingGroup">
2226         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2227 tology#CriticalApparatusEntryContent"/>
2228         <rdfs:comment xml:lang="en">A set of readings within an entry of
2229

```

```

2230 the critical apparatus, grouped to highlight some peculiarities of the tex-
2231 tual tradition or to facilitate consultation. A group of readings is graph-
2232 ically distinct from the other contents of the critical apparatus
2233 entry.</rdfs:comment>
2234     <rdfs:label xml:lang="en">Reading Group</rdfs:label>
2235     </owl:Class>
2236
2237
2238     <!-- http://purl.org/critical-edition-ontology#ReadingInApparatus -->
2239
2240     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Read-
2241 ingInApparatus">
2242         <owl:equivalentClass rdf:resource="https://w3id.org/cao#Reading"/>
2243         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2244 tology#CriticalApparatusEntryContent"/>
2245         <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-
2246 crm/E90_Symbolic_Object"/>
2247         <rdfs:comment xml:lang="en">Reading presented in a critical appa-
2248 ratus entry. A reading in the critical apparatus may correspond to two or
2249 more readings of different witnesses. The reading in the critical apparatus
2250 can be presented in a diplomatic (according to a particular witness) or in-
2251 terpretative form.</rdfs:comment>
2252         <rdfs:label xml:lang="en">Reading In Apparatus</rdfs:label>
2253     </owl:Class>
2254
2255
2256     <!-- http://purl.org/critical-edition-ontology#ReadingInApparatusCause
2257 -->
2258
2259     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Read-
2260 ingInApparatusCause">
2261         <owl:equivalentClass rdf:resource="https://w3id.org/cao#Reading-
2262 Cause"/>
2263         <rdfs:comment xml:lang="en">The type of error made by the copyist
2264 during the act of copying or the factor that may have led the copyist to
2265 produce an erroneous or innovative reading.</rdfs:comment>
2266         <rdfs:label xml:lang="en">Reading In Apparatus Cause</rdfs:label>
2267     </owl:Class>
2268
2269
2270     <!-- http://purl.org/critical-edition-ontology#ReadingInApparatusType -
2271 -->
2272
2273     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Read-
2274 ingInApparatusType">
2275         <owl:equivalentClass rdf:resource="https://w3id.org/cao#Read-
2276 ingType"/>
2277         <rdfs:comment xml:lang="en">Category that the editor assigns to a
2278 lesson to classify it and make their evaluation explicit.</rdfs:comment>
2279         <rdfs:label xml:lang="en">Reading In Apparatus Type</rdfs:label>
2280     </owl:Class>
2281
2282
2283     <!-- http://purl.org/critical-edition-ontology#ScholarlyEdition -->
2284
2285     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Schol-
2286 arlyEdition">
2287         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2288 tology#F2_Expression"/>
2289         <rdfs:comment xml:lang="en">The content of a publication that makes
2290 the text of a literary work or work of historical-cultural interest, mostly

```

```

2291 of the past, otherwise inaccessible, accessible to the public. The edition
2292 is called "scholarly" as it is created by a scholar or a group of scholars,
2293 applying shared methodologies in their own philological field.</rdfs:com-
2294 ment>
2295     <rdfs:label xml:lang="en">Scholarly Edition</rdfs:label>
2296     </owl:Class>
2297
2298
2299 <!--http://purl.org/critical-edition-ontology#ScholarlyEditionComponent -->
2300
2301     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Schol-
2302 arlyEditionComponent">
2303         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2304 tology#F2_Expression"/>
2305         <rdfs:comment xml:lang="en">A component of a scholarly edition.
2306 Usually the component of a scholarly edition is a text, but it can also be
2307 an image, a multimedia object or another kind of object. Among the main
2308 components of a scholarly edition are the edited text and the materials ac-
2309 companying the edited text.</rdfs:comment>
2310         <rdfs:label xml:lang="en">Scholarly Edition Component</rdfs:label>
2311     </owl:Class>
2312
2313
2314 <!--http://purl.org/critical-edition-ontology#ScholarlyEditionPublication--
2315 >
2316
2317     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Schol-
2318 arlyEditionPublication">
2319         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2320 tology#F3_Manifestation"/>
2321         <rdfs:comment xml:lang="en">The publication of a scholarly edition,
2322 which comprises both the content (the scholarly edition itself), its
2323 presentation and the material support.</rdfs:comment>
2324         <rdfs:label xml:lang="en">Scholarly Edition Publication</rdfs:la-
2325 bel>
2326     </owl:Class>
2327
2328
2329 <!-- https://www.w3.org/ns/oa#Selector -->
2330
2331     <owl:Class rdf:about="https://www.w3.org/ns/oa#Selector">
2332         <rdfs:comment xml:lang="en">A resource which describes the segment
2333 of interest in a representation of a Source resource, indicated with
2334 oa:hasSelector from the Specific Resource. This class is not used directly
2335 in the Annotation model, only its subclasses.</rdfs:comment>
2336         <rdfs:label xml:lang="en">Selector</rdfs:label>
2337     </owl:Class>
2338
2339
2340 <!-- http://purl.org/critical-edition-ontology#Siglum -->
2341
2342     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Sig-
2343 lum">
2344         <rdfs:comment xml:lang="en">Alphanumeric string that identifies a
2345 witness within a critical edition.</rdfs:comment>
2346         <rdfs:label xml:lang="en">Siglum</rdfs:label>
2347     </owl:Class>
2348
2349
2350 <!-- http://purl.org/critical-edition-ontology#SiglumReference -->
2351

```

```

2352     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Siglum-
2353 Reference">
2354         <rdfs:comment xml:lang="en">Reference to a witness siglum, used
2355 within the critical apparatus (and in other apparatuses), next to readings
2356 and optionally to reconstructed passage references.</rdfs:comment>
2357         <rdfs:label xml:lang="en">Siglum Reference</rdfs:label>
2358     </owl:Class>
2359
2360
2361     <!-- http://purl.org/critical-edition-ontology#StemmaCodicum -->
2362
2363     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Stemma-
2364 Codicum">
2365         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2366 tology#AccompanyingMaterial"/>
2367         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2368 tology#IntroductionPart"/>
2369         <rdfs:subClassOf rdf:resource="http://www.cidoc-crm.org/cidoc-
2370 crm/E36_Visual_Item"/>
2371         <rdfs:comment xml:lang="en">Graph that represents the genealogical
2372 tree of the textual transmission of a work.</rdfs:comment>
2373         <rdfs:label xml:lang="en">Stemma Codicum</rdfs:label>
2374     </owl:Class>
2375
2376
2377     <!-- http://purl.org/critical-edition-ontology#SurfaceWitness -->
2378
2379     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Sur-
2380 faceWitness">
2381         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2382 tology#Witness"/>
2383         <rdfs:comment xml:lang="en">Witness used as a reference to estab-
2384 lish the linguistic style of the critical text.</rdfs:comment>
2385         <rdfs:label xml:lang="en">Surface Witness</rdfs:label>
2386     </owl:Class>
2387
2388
2389     <!-- http://www.w3.org/ns/oa#TextualBody -->
2390
2391     <owl:Class rdf:about="http://www.w3.org/ns/oa#TextualBody"/>
2392
2393
2394     <!-- http://purl.org/critical-edition-ontology#TextualTradition -->
2395
2396     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Textu-
2397 alTradition">
2398         <rdfs:subClassOf>
2399             <owl:Restriction>
2400                 <owl:onProperty rdf:resource="http://purl.org/critical-edi-
2401 tion-ontology#hasPart"/>
2402                 <owl:minQualifiedCardinality
2403 rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInte-
2404 ger">1</owl:minQualifiedCardinality>
2405                 <owl:onClass rdf:resource="http://purl.org/critical-ed-
2406 tion-ontology#Witness"/>
2407             </owl:Restriction>
2408         </rdfs:subClassOf>
2409         <rdfs:comment xml:lang="en">Set of witnesses that preserve a
2410 work.</rdfs:comment>
2411         <rdfs:label xml:lang="en">Textual Tradition</rdfs:label>
2412     </owl:Class>

```

```

2413
2414
2415     <!-- http://purl.org/critical-edition-ontology#UnclearPassage -->
2416
2417     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Un-
2418 clearPassage">
2419         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2420 tology#Reading"/>
2421         <rdfs:comment xml:lang="en">Witness passage of variable size, which
2422 the editor cannot read and report with certainty due to how it appears
2423 within its material support.</rdfs:comment>
2424         <rdfs:label xml:lang="en">Unclear Passage</rdfs:label>
2425     </owl:Class>
2426
2427
2428     <!-- http://purl.org/critical-edition-ontology#Witness -->
2429
2430     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Wit-
2431 ness">
2432         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2433 tology#F2_Expression"/>
2434         <rdfs:subClassOf>
2435             <owl:Restriction>
2436                 <owl:onProperty rdf:resource="http://purl.org/critical-edi-
2437 tion-ontology#isPartOf"/>
2438                 <owl:qualifiedCardinality
2439 rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInte-
2440 ger">1</owl:qualifiedCardinality>
2441                 <owl:onClass rdf:resource="http://purl.org/critical-edi-
2442 tion-ontology#TextualTradition"/>
2443             </owl:Restriction>
2444         </rdfs:subClassOf>
2445         <rdfs:subClassOf>
2446             <owl:Restriction>
2447                 <owl:onProperty rdf:resource="http://purl.org/critical-edi-
2448 tion-ontology#isPreservedIn"/>
2449                 <owl:qualifiedCardinality
2450 rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInte-
2451 ger">1</owl:qualifiedCardinality>
2452                 <owl:onClass rdf:resource="http://purl.org/critical-edi-
2453 tion-ontology#WitnessCarrier"/>
2454             </owl:Restriction>
2455         </rdfs:subClassOf>
2456         <rdfs:comment xml:lang="en">Copy of the text of a work.</rdfs:com-
2457 ment>
2458         <rdfs:label xml:lang="en">Witness</rdfs:label>
2459     </owl:Class>
2460
2461
2462     <!-- http://purl.org/critical-edition-ontology#WitnessCarrier -->
2463
2464     <owl:Class rdf:about="http://purl.org/critical-edition-ontology#Wit-
2465 nessCarrier">
2466         <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-on-
2467 tology#F3_Manifestation"/>
2468         <rdfs:subClassOf>
2469             <owl:Restriction>
2470                 <owl:onProperty rdf:resource="http://purl.org/critical-edi-
2471 tion-ontology#preserves"/>
2472                 <owl:minQualifiedCardinality

```

```

2473   rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:minQualifiedCardinality>
2474
2475       <owl:onClass rdf:resource="http://purl.org/critical-edition-ontology#Witness"/>
2476   </owl:Restriction>
2477   </rdfs:subClassOf>
2478   <rdfs:comment xml:lang="en">The physical support of a witness, usually a manuscript or a printed book.</rdfs:comment>
2479   <rdfs:label xml:lang="en">Witness Carrier</rdfs:label>
2480   </owl:Class>
2481
2482
2483
2484
2485   <!-- http://purl.org/critical-edition-ontology#WitnessDescription -->
2486
2487   <owl:Class rdf:about="http://purl.org/critical-edition-ontology#WitnessDescription">
2488     <rdfs:comment xml:lang="en">Text that can be provided within the witnesses list, to describe a witness in more or less depth and dealing with different aspects: the state of conservation, the material aspect, the palaeographic analysis, etc.</rdfs:comment>
2489     <rdfs:label xml:lang="en">Witness Description</rdfs:label>
2490   </owl:Class>
2491
2492
2493
2494
2495   <!-- http://purl.org/critical-edition-ontology#WitnessesList -->
2496
2497   <owl:Class rdf:about="http://purl.org/critical-edition-ontology#WitnessesList">
2498     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-ontology#AccompanyingMaterial"/>
2499     <rdfs:subClassOf rdf:resource="http://purl.org/critical-edition-ontology#IntroductionPart"/>
2500     <rdfs:comment xml:lang="en">List of the witnesses that preserve the text of a work. The list is presented within a critical edition, usually inside the introduction.</rdfs:comment>
2501     <rdfs:label xml:lang="en">Witnesses List</rdfs:label>
2502   </owl:Class>
2503
2504
2505
2506
2507   <!-- http://purl.org/emmedi/hico#InterpretationAct -->
2508
2509   <owl:Class rdf:about="http://purl.org/emmedi/hico#InterpretationAct"/>
2510
2511
2512
2513
2514
2515
2516
2517   <!-- http://purl.org/emmedi/hico#InterpretationCriterion -->
2518
2519   <owl:Class rdf:about="http://purl.org/emmedi/hico#InterpretationCriterion"/>
2520
2521
2522
2523
2524   <!-- http://purl.org/emmedi/hico#InterpretationType -->
2525
2526   <owl:Class rdf:about="http://purl.org/emmedi/hico#InterpretationType"/>
2527
2528
2529
2530
2531   <!-- http://www.cidoc-crm.org/cidoc-crm/E21_Person -->
2532
2533   <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E21_Person"/>
2534
2535
2536
2537   <!-- http://www.cidoc-crm.org/cidoc-crm/E90_Symbolic_Object -->

```

```

2534
2535     <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E90_Sym-
2536 bolic_Object"/>
2537
2538
2539     <!-- http://www.cidoc-crm.org/cidoc-crm/E36_Visual_Item -->
2540
2541     <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E36_Vis-
2542 ual_Item"/>
2543
2544
2545     <!-- http://www.cidoc-crm.org/cidoc-crm/E39_Actor -->
2546
2547     <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E39_Actor">
2548         <owl:equivalentClass rdf:re-
2549 source="http://www.w3.org/ns/prov#Agent"/>
2550     </owl:Class>
2551
2552
2553     <!-- http://www.cidoc-crm.org/cidoc-crm/E41_Appellation -->
2554
2555     <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E41_Appella-
2556 tion"/>
2557
2558
2559     <!-- http://www.cidoc-crm.org/cidoc-crm/E74_Group4 -->
2560
2561     <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E74_Group">
2562
2563     </owl:Class>
2564
2565
2566     <!-- http://www.cidoc-crm.org/cidoc-crm/E90_Symbolic_Object -->
2567
2568     <owl:Class rdf:about="http://www.cidoc-crm.org/cidoc-crm/E90_Sym-
2569 bolic_Object"/>
2570
2571
2572     <!-- http://www.w3.org/ns/prov#Agent -->
2573
2574     <owl:Class rdf:about="http://www.w3.org/ns/prov#Agent"/>
2575
2576
2577     <!-- https://w3id.org/cao#BaseReading -->
2578
2579     <owl:Class rdf:about="https://w3id.org/cao#BaseReading"/>
2580
2581
2582     <!-- https://w3id.org/cao#Reading -->
2583
2584     <owl:Class rdf:about="https://w3id.org/cao#Reading"/>
2585
2586
2587     <!-- https://w3id.org/cao#ReadingCause -->
2588
2589     <owl:Class rdf:about="https://w3id.org/cao#ReadingCause"/>
2590
2591
2592     <!-- https://w3id.org/cao#ReadingType -->
2593
2594     <owl:Class rdf:about="https://w3id.org/cao#ReadingType"/>

```

```

2595
2596
2597 <!--////////////////////////////////////
2598 //
2599 // Individuals
2600 //
2601 //////////////////////////////////////-->
2602
2603
2604
2605 <!-- http://purl.org/critical-edition-ontology#addition -->
2606
2607 <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2608 ogy#addition">
2609 <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2610 ogy#ReadingInApparatusType"/>
2611 <rdfs:comment xml:lang="en">An addition is a text string that the
2612 copyist inserts into the text but is not present in the antigraph, or a
2613 text string omitted by the copyist and added later elsewhere in the wit-
2614 ness.</rdfs:comment>
2615 <rdfs:label xml:lang="en">addition</rdfs:label>
2616 </owl:NamedIndividual>
2617
2618
2619 <!-- http://purl.org/critical-edition-ontology#characteristicReading -->
2620
2621 <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2622 ogy#characteristicReading">
2623 <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2624 ogy#ReadingInApparatusType"/>
2625 <rdfs:comment xml:lang="en">Non erroneous reading common to several
2626 witnesses, which can confirm their common origin if this is demonstrated by
2627 conjunctive errors.</rdfs:comment>
2628 <rdfs:label xml:lang="en">characteristic reading</rdfs:label>
2629 </owl:NamedIndividual>
2630
2631
2632 <!-- http://purl.org/critical-edition-ontology#conjecture -->
2633
2634 <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2635 ogy#conjecture">
2636 <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2637 ogy#ReadingInApparatusType"/>
2638 <rdfs:comment xml:lang="en">Conjectural reading proposed by an edi-
2639 tor external to the critical edition or by the copyist of a wit-
2640 ness.</rdfs:comment>
2641 <rdfs:label xml:lang="en">conjecture</rdfs:label>
2642 </owl:NamedIndividual>
2643
2644
2645 <!-- http://purl.org/critical-edition-ontology#conjunctiveError -->
2646
2647 <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2648 ogy#conjunctiveError">
2649 <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2650 ogy#ReadingInApparatusType"/>
2651 <rdfs:comment xml:lang="en">Error present in two or more witnesses,
2652 which allows to deduce that they are copies of each other or copies of the
2653 same lost witness.</rdfs:comment>
2654 <rdfs:label xml:lang="en">conjunctive error</rdfs:label>
2655 </owl:NamedIndividual>

```



```

2656
2657
2658     <!-- http://purl.org/critical-edition-ontology#deletion -->
2659
2660     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2661 ogy#deletion">
2662         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2663 ogy#ReadingInApparatusType"/>
2664         <rdfs:comment xml:lang="en">String of text that the copyist somehow
2665 designated to be removed.</rdfs:comment>
2666         <rdfs:label xml:lang="en">deletion</rdfs:label>
2667     </owl:NamedIndividual>
2668
2669
2670     <!-- http://purl.org/critical-edition-ontology#deliberateOmission -->
2671
2672     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2673 ogy#deliberateOmission">
2674         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2675 ogy#ReadingInApparatusType"/>
2676         <rdfs:comment xml:lang="en">Text string deliberately omitted by the
2677 copyist in the act of copying the text.</rdfs:comment>
2678         <rdfs:label xml:lang="en">deliberate omission</rdfs:label>
2679     </owl:NamedIndividual>
2680
2681
2682     <!-- http://purl.org/critical-edition-ontology#diffraction -->
2683
2684     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2685 ogy#diffraction">
2686         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2687 ogy#ReadingInApparatusType"/>
2688         <rdfs:comment xml:lang="en">The phenomenon whereby tradition pre-
2689 sents multiple alternative and distinct variants for a single textual pas-
2690 sage, resulting from the scribes' reaction to a difficulty in the original
2691 text.</rdfs:comment>
2692         <rdfs:label xml:lang="en">diffraction</rdfs:label>
2693     </owl:NamedIndividual>
2694
2695
2696     <!-- http://purl.org/critical-edition-ontology#diffractionInAbsentia -->
2697
2698     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2699 ogy#diffractionInAbsentia">
2700         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2701 ogy#ReadingInApparatusType"/>
2702         <rdfs:comment xml:lang="en">The same as diffraction, but with the
2703 peculiarity that none of the variants attested within the textual transmis-
2704 sion is considered to be the origin of the phenomenon by the edi-
2705 tor.</rdfs:comment>
2706         <rdfs:label xml:lang="en">diffraction in absentia</rdfs:label>
2707     </owl:NamedIndividual>
2708
2709
2710     <!-- http://purl.org/critical-edition-ontology#diffractionInPresentia -->
2711
2712     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2713 ogy#diffractionInPresentia">
2714         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2715 ogy#ReadingInApparatusType"/>
2716         <rdfs:comment xml:lang="en">The same as diffraction, but with the

```

```

2717 peculiarity that one of the variants attested within the textual transmis-
2718 sion is considered to be the origin of the phenomenon by the edi-
2719 tor.</rdfs:comment>
2720     <rdfs:label xml:lang="en">diffraction in presentia</rdfs:label>
2721     </owl:NamedIndividual>
2722
2723
2724     <!-- http://purl.org/critical-edition-ontology#easilyCorrigibleError -->
2725
2726     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2727 ogy#easilyCorrigibleError">
2728         <rdfs:type rdf:resource="http://purl.org/critical-edition-ontol-
2729 ogy#ReadingInApparatusType"/>
2730         <rdfs:comment xml:lang="en">Obviously erroneous reading and there-
2731 fore easily corrigible.</rdfs:comment>
2732         <rdfs:label xml:lang="en">easily corrigible error</rdfs:label>
2733     </owl:NamedIndividual>
2734
2735
2736     <!-- http://purl.org/critical-edition-ontology#emendatio -->
2737
2738     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2739 ogy#emendatio">
2740         <rdfs:type rdf:resource="http://purl.org/critical-edition-ontol-
2741 ogy#CriticalInterventionType"/>
2742         <rdfs:comment xml:lang="en">A critical intervention can be classi-
2743 fied as 'emendatio' when the editor corrects a passage in the text of the
2744 work.</rdfs:comment>
2745         <rdfs:label xml:lang="lat">emendatio</rdfs:label>
2746     </owl:NamedIndividual>
2747
2748
2749     <!-- http://purl.org/critical-edition-ontology#emendatioExFontibus -->
2750
2751     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2752 ogy#emendatioExFontibus">
2753         <rdfs:type rdf:resource="http://purl.org/critical-edition-ontol-
2754 ogy#CriticalInterventionType"/>
2755         <rdfs:comment xml:lang="en">A critical intervention can be classi-
2756 fied as 'emendatio ex fontibus' when the editor corrects a passage in the
2757 text, based on the sources they consider to have been used in writing the
2758 text of the work.</rdfs:comment>
2759         <rdfs:label xml:lang="lat">emendatio ex fontibus</rdfs:label>
2760     </owl:NamedIndividual>
2761
2762
2763     <!-- http://purl.org/critical-edition-ontology#emendatioOpeCodicum -->
2764
2765     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2766 ogy#emendatioOpeCodicum">
2767         <rdfs:type rdf:resource="http://purl.org/critical-edition-ontol-
2768 ogy#CriticalInterventionType"/>
2769         <rdfs:comment xml:lang="en">A critical intervention can be classi-
2770 fied as 'emendatio ope codicum' when the editor corrects a passage in the
2771 text, resorting to a good reading transmitted by one or more wit-
2772 nesses.</rdfs:comment>
2773         <rdfs:label xml:lang="lat">emendatio ope codicum</rdfs:label>
2774     </owl:NamedIndividual>
2775
2776
2777     <!-- http://purl.org/critical-edition-ontology#emendatioOpeIngenii -->

```

```

2778
2779     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2780 ogy#emendatioOpeIngenii">
2781         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2782 ogy#CriticalInterventionType"/>
2783         <rdfs:comment xml:lang="en">A critical intervention can be classi-
2784 fied as 'emendatio ope ingenii' when the editor corrects a place in the
2785 text, proposing a conjectural reading, not attested by any wit-
2786 ness.</rdfs:comment>
2787         <rdfs:label xml:lang="lat">emendatio ope ingenii</rdfs:label>
2788     </owl:NamedIndividual>
2789
2790
2791     <!-- http://purl.org/critical-edition-ontology#error -->
2792
2793     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2794 ogy#error">
2795         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2796 ogy#ReadingInApparatusType"/>
2797         <rdfs:comment xml:lang="en">Reading that is considered not to be in
2798 the original form of the text as it is incongruous with respect to the
2799 grammatical, semantic, metric, narrative coherence of the text.</rdfs:com-
2800 ment>
2801         <rdfs:label xml:lang="en">error</rdfs:label>
2802     </owl:NamedIndividual>
2803
2804
2805     <!-- http://purl.org/critical-edition-ontology#indifferentVariant -->
2806
2807     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2808 ogy#indifferentVariant">
2809         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2810 ogy#ReadingInApparatusType"/>
2811         <rdfs:comment xml:lang="en">Variant for which according to the edi-
2812 tor it is not possible to express a preference with respect to the others
2813 equally attributable to the original form of the text.</rdfs:comment>
2814         <rdfs:label xml:lang="en">indifferent variant</rdfs:label>
2815     </owl:NamedIndividual>
2816
2817
2818     <!-- http://purl.org/critical-edition-ontology#innovation -->
2819
2820     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2821 ogy#innovation">
2822         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2823 ogy#ReadingInApparatusType"/>
2824         <rdfs:comment xml:lang="en">Reading that alters the text of the an-
2825 tigraph and therefore cannot be attributed to the original form of the
2826 text.</rdfs:comment>
2827         <rdfs:label xml:lang="en">innovation</rdfs:label>
2828     </owl:NamedIndividual>
2829
2830
2831     <!--http://purl.org/critical-edition-ontology#interScribendumCorrection -->
2832
2833     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2834 ogy#interScribendumCorrection">
2835         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2836 ogy#ReadingInApparatusType"/>
2837         <rdfs:comment xml:lang="en">Error present in two or more witnesses,
2838 which allows to deduce that they are copies of each other or copies of the

```

```

2839 same lost witness.</rdfs:comment>
2840     <rdfs:label xml:lang="en">inter scribendum correction</rdfs:label>
2841     </owl:NamedIndividual>
2842
2843
2844     <!-- http://purl.org/critical-edition-ontology#lectioDifficilior -->
2845
2846     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2847 ogy#lectioDifficilior">
2848         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2849 ogy#ReadingInApparatusType"/>
2850         <rdfs:comment xml:lang="en">More complex or rare or refined reading
2851 compared to the alternative variants, and therefore probably origi-
2852 nal.</rdfs:comment>
2853         <rdfs:label xml:lang="lat">lectio difficilior</rdfs:label>
2854     </owl:NamedIndividual>
2855
2856
2857     <!-- http://purl.org/critical-edition-ontology#lectioFacilior -->
2858
2859     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2860 ogy#lectioFacilior">
2861         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2862 ogy#ReadingInApparatusType"/>
2863         <rdfs:comment xml:lang="en">Less complex or rare or refined reading
2864 compared to the alternative variants, and therefore probably not origi-
2865 nal.</rdfs:comment>
2866         <rdfs:label xml:lang="lat">lectio facilior</rdfs:label>
2867     </owl:NamedIndividual>
2868
2869
2870     <!-- http://purl.org/critical-edition-ontology#lectioSingularis -->
2871
2872     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2873 ogy#lectioSingularis">
2874         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2875 ogy#ReadingInApparatusType"/>
2876         <rdfs:comment xml:lang="en">Reading transmitted by a single witness
2877 or a single sub-family of witnesses who are isolated in the stemma with re-
2878 spect to the tradition as a whole.</rdfs:comment>
2879         <rdfs:label xml:lang="lat">lectio singularis</rdfs:label>
2880     </owl:NamedIndividual>
2881
2882
2883     <!-- http://purl.org/critical-edition-ontology#mechanicLacuna -->
2884
2885     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2886 ogy#mechanicLacuna">
2887         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2888 ogy#ReadingInApparatusType"/>
2889         <rdfs:comment xml:lang="en">Omission due to the physical loss of
2890 part of the witness.</rdfs:comment>
2891         <rdfs:label xml:lang="en">mechanic lacuna</rdfs:label>
2892     </owl:NamedIndividual>
2893
2894
2895     <!-- http://purl.org/critical-edition-ontology#monogeneticError -->
2896
2897     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2898 ogy#monogeneticError">

```

```

2899     <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2900 ogy#ReadingInApparatusType"/>
2901     <rdfs:comment xml:lang="en">An error which is unlikely to have been
2902 committed by several copyists independently of each other.</rdfs:comment>
2903     <rdfs:label xml:lang="en">monogenetic error</rdfs:label>
2904     </owl:NamedIndividual>
2905
2906
2907     <!-- http://purl.org/critical-edition-ontology#orthographicVariant -->
2908
2909     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2910 ogy#orthographicVariant">
2911         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2912 ogy#ReadingInApparatusType"/>
2913         <rdfs:comment xml:lang="en">Reading that differs on a formal level
2914 (graphic and/or phonetic) from the reading considered original by the pub-
2915 lisher.</rdfs:comment>
2916         <rdfs:label xml:lang="en">orthographic variant</rdfs:label>
2917     </owl:NamedIndividual>
2918
2919
2920     <!-- http://purl.org/critical-edition-ontology#polygeneticError -->
2921
2922     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2923 ogy#polygeneticError">
2924         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2925 ogy#ReadingInApparatusType"/>
2926         <rdfs:comment xml:lang="en">An error that is likely to have been
2927 committed by several copyists independently of each other.</rdfs:comment>
2928         <rdfs:label xml:lang="en">polygenetic error</rdfs:label>
2929     </owl:NamedIndividual>
2930
2931
2932     <!-- http://purl.org/critical-edition-ontology#selectio -->
2933
2934     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2935 ogy#selectio">
2936         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2937 ogy#CriticalInterventionType"/>
2938         <rdfs:comment xml:lang="en">A critical intervention can be classi-
2939 fied as 'selectio' when the editor constitutes the passage of the critical
2940 text by choosing among the variants.</rdfs:comment>
2941         <rdfs:label xml:lang="lat">selectio</rdfs:label>
2942     </owl:NamedIndividual>
2943
2944
2945     <!-- http://purl.org/critical-edition-ontology#separativeError -->
2946
2947     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2948 ogy#separativeError">
2949         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2950 ogy#ReadingInApparatusType"/>
2951         <rdfs:comment xml:lang="en">Error present in a witness A and absent
2952 in a witness B, which allows to deduce that B does not derive from
2953 A.</rdfs:comment>
2954         <rdfs:label xml:lang="en">separative error</rdfs:label>
2955     </owl:NamedIndividual>
2956
2957
2958     <!-- http://purl.org/critical-edition-ontology#significantError -->
2959

```

```

2960     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2961 ogy#significantError">
2962         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2963 ogy#ReadingInApparatusType"/>
2964         <rdfs:comment xml:lang="en">Conjunctive or separative er-
2965 ror.</rdfs:comment>
2966         <rdfs:label xml:lang="en">significant error</rdfs:label>
2967     </owl:NamedIndividual>
2968
2969
2970     <!-- http://purl.org/critical-edition-ontology#substantiveVariant -->
2971
2972     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2973 ogy#substantiveVariant">
2974         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2975 ogy#ReadingInApparatusType"/>
2976         <rdfs:comment xml:lang="en">Reading that differs on a semantic
2977 level from the reading considered original by the editor.</rdfs:comment>
2978         <rdfs:label xml:lang="en">substantive variant</rdfs:label>
2979     </owl:NamedIndividual>
2980
2981
2982     <!-- http://purl.org/critical-edition-ontology#variant -->
2983
2984     <owl:NamedIndividual rdf:about="http://purl.org/critical-edition-ontol-
2985 ogy#variant">
2986         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2987 ogy#ReadingInApparatusType"/>
2988         <rdfs:comment xml:lang="en">Reading of one witness different from
2989 that of another, without one of them necessarily being errone-
2990 ous.</rdfs:comment>
2991         <rdfs:label xml:lang="en">variant</rdfs:label>
2992     </owl:NamedIndividual>
2993
2994
2995     <!-- https://w3id.org/cao#addition -->
2996
2997     <owl:NamedIndividual rdf:about="https://w3id.org/cao#addition">
2998         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
2999 ogy#ReadingInApparatusType"/>
3000         <owl:sameAs rdf:resource="http://purl.org/critical-edition-ontol-
3001 ogy#addition"/>
3002     </owl:NamedIndividual>
3003
3004
3005     <!-- https://w3id.org/cao#conjecture -->
3006
3007     <owl:NamedIndividual rdf:about="https://w3id.org/cao#conjecture">
3008         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3009 ogy#ReadingInApparatusType"/>
3010         <owl:sameAs rdf:resource="http://purl.org/critical-edition-ontol-
3011 ogy#conjecture"/>
3012     </owl:NamedIndividual>
3013
3014
3015     <!-- https://w3id.org/cao#correction -->
3016
3017     <owl:NamedIndividual rdf:about="https://w3id.org/cao#correction">
3018         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3019 ogy#ReadingInApparatusType"/>
3020     </owl:NamedIndividual>

```

```

3021
3022
3023     <!-- https://w3id.org/cao#deletion -->
3024
3025     <owl:NamedIndividual rdf:about="https://w3id.org/cao#deletion">
3026         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3027 ogy#ReadingInApparatusType"/>
3028         <owl:sameAs rdf:resource="http://purl.org/critical-edition-ontol-
3029 ogy#deletion"/>
3030     </owl:NamedIndividual>
3031
3032
3033     <!-- https://w3id.org/cao#dittography -->
3034
3035     <owl:NamedIndividual rdf:about="https://w3id.org/cao#dittography">
3036         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3037 ogy#ReadingInApparatusCause"/>
3038     </owl:NamedIndividual>
3039
3040
3041     <!-- https://w3id.org/cao#haplography -->
3042
3043     <owl:NamedIndividual rdf:about="https://w3id.org/cao#haplography">
3044         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3045 ogy#ReadingInApparatusCause"/>
3046     </owl:NamedIndividual>
3047
3048
3049     <!-- https://w3id.org/cao#homeoarchy -->
3050
3051     <owl:NamedIndividual rdf:about="https://w3id.org/cao#homeoarchy">
3052         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3053 ogy#ReadingInApparatusCause"/>
3054     </owl:NamedIndividual>
3055
3056
3057     <!-- https://w3id.org/cao#homeoteleuton -->
3058
3059     <owl:NamedIndividual rdf:about="https://w3id.org/cao#homeoteleuton">
3060         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3061 ogy#ReadingInApparatusCause"/>
3062     </owl:NamedIndividual>
3063
3064
3065     <!-- https://w3id.org/cao#incorporation -->
3066
3067     <owl:NamedIndividual rdf:about="https://w3id.org/cao#incorporation">
3068         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3069 ogy#ReadingInApparatusCause"/>
3070     </owl:NamedIndividual>
3071
3072
3073     <!-- https://w3id.org/cao#omission -->
3074
3075     <owl:NamedIndividual rdf:about="https://w3id.org/cao#omission">
3076         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3077 ogy#ReadingInApparatusType"/>
3078     </owl:NamedIndividual>
3079
3080
3081     <!-- https://w3id.org/cao#polyptoton -->

```

```
3082
3083     <owl:NamedIndividual rdf:about="https://w3id.org/cao#polyptoton">
3084         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3085 ogy#ReadingInApparatusCause"/>
3086     </owl:NamedIndividual>
3087
3088
3089     <!-- https://w3id.org/cao#saut-du-meme-au-meme -->
3090
3091     <owl:NamedIndividual rdf:about="https://w3id.org/cao#saut-du-meme-au-
3092 meme">
3093         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3094 ogy#ReadingInApparatusCause"/>
3095     </owl:NamedIndividual>
3096
3097
3098     <!-- https://w3id.org/cao#transposition -->
3099
3100     <owl:NamedIndividual rdf:about="https://w3id.org/cao#transposition">
3101         <rdf:type rdf:resource="http://purl.org/critical-edition-ontol-
3102 ogy#ReadingInApparatusType"/>
3103     </owl:NamedIndividual>
3104 </rdf:RDF>
```