

“DIGITAL NATIVE” CRITICAL EDITIONS AND HOMEMADE SCHOOL TEXT ANALYSIS: THE *HYPER PROJECT*

**Proff. Vincenzo Muggittu,
Domenico Cufalo**

University of Pisa

Abstract. Nowadays, although we are living in the “Digital Era”, there still do exist many serious issues which prevent scholars, teachers and students from enjoying an actual “Digital Critical Hypertext” – fully equipped with variants, paratexts, multiple translations, linguistic analysis, notes and links, but easy to develop and/or read and which is able to overcome the issues of traditional critical editing methodologies (e.g., Lachmann’s and Bédier’s methods). This paper deals with these issues and presents *HYPER*, an innovative application meant for both universities and schools, designed to set up and/or read a Digital Critical Hypertext in a fully interactive, customizable and user-friendly way.

Keywords: textual scholarship, philology, textual criticism, critical edition, hypertext, TEI, JSON, TEJSON, digital humanities, educational.

In this paper, we are proud to present *HYPER*, an innovative platform for digital editions conceived for both schools and universities.

There are already lots of similar projects¹ around the web, such as *Scholarly Digital Editions*, *The Homer Multitext*, Mastrorade’s *Euripides Scholia* or Münster University’s *New Testament Transcripts* (this last is our personal choice), but all these are heavily specialised projects, mainly targeted at scholars, and therefore could be unintuitive for students, novices or casual users. In our opinion, they all lack user-friendliness, due to at least three actual limitations:

- of TEI coding language;²
- of editing;
- of interfaces.

All these projects are mainly “server-oriented” and therefore they often have a high infrastructure cost. Moreover, even the opportunity to contribute to them often requires advanced computer skills.

HYPER, on the other hand, is designed to deal with these issues while offering a very portable infrastructure, which can enter into an old floppy disc, although equipped with a powerful and elegant graphical user interface (GUI) accessible to anyone, both for online and offline use; all you need is a web browser, regardless of the operating system.

¹ See the comprehensive survey of projects in *Digital Critical Editions of Texts in Greek and Latin* (http://wiki.digitalclassicist.org/Digital_Critical_Editions_of_Texts_in_Greek_and_Latin).

² Actually TEI, an XML language specifically designed to digitalize documents, is becoming a *de facto* standard: cf. its advantages over other languages in Stührenberg 2012.

Let us look at some issues and solutions in detail.

The limitations

Limitations of TEI coding language

TEI is very good for document digitalization, but is “document-oriented”, **not** “critical edition-oriented”: at its best, TEI allows digitalizations of paper critical editions, but itself is not designed for “digital native” critical editions. And this leads to some troubles.

Here is a couple of examples:

1. The TEI <gloss> element is very useful when one wants to “provide a gloss or definition for some other word or phrase”,³ but it cannot be used to mark an ancient “glossa” inside a manuscript;
2. The TEI <addition> element allows to describe *marginalia* of a single manuscript, but does not allow to insert the full text of all the *marginalia* from whatever manuscript into a critical edition, as sources of textual variants and analysis objects. The *Homer Multitext project* devised a clever workaround “using TEI *div* elements; each scholion is contained by a div with @type attribute = ‘scholion’”,⁴ but the TEI <div> element is designed for generic typographical text divisions,⁵ not for source paratext⁶ identifi-

cation (these problems are well-known: cf. e. g. Porter & Withers 2005, Monella 2008 and a recent overview by Salvagni 2012).

To sum up, TEI currently lacks specific tools to track and manage a clear distinction between notes made by the editor and every type of paratexts-marginalia already included in several sources.

Another instance of TEI limitations stays in the management of the critical apparatus: TEI allows a clear distinction between textual variants supplied by witnesses (@wit attribute) and emendations/corrections made by scholars and/or editors (@resp attribute), but the @resp attribute mixes up “real” corrections and readings proposed by transcribers. To sum up, @resp mixes up corrections not existing in any source and readings of an uncertain but an existing source.

Lastly, the availability of three different methods of apparatus-text linking⁷ could seem advantageous, but we fear this could lead to confusion and cause many difficulties in subsequent apparatus visualization (our personal choice – see below – has been the “parallel segmentation method”, because it’s inline [i.e., the variants are managed as part of a critical text, instead of being managed as a separated text]; only this method secures the position of the variants inside a critical text and gives many advantages in subsequent apparatus visualization).

Limitations of editing

History teaches a lot of things. Among them, history teaches that **simplicity, in-**

³ See <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/CO.html#COHQ>.

⁴ See <http://folio.furman.edu/hmt-doc/standards/housestyle.html>.

⁵ See <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/DS.html#DSDIV>.

⁶ We use the term *paratext* as meant by its inventor Gérard Genette.

⁷ “Location-referenced method”, “double-end-point-attached method”, and “parallel segmentation method”. See details in <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPLK>.

tuitiveness and **accessibility** are the key elements of a successful diffusion of texts and/or ideas. Unfortunately, nowadays critical editions, either printed or digital, have none of these criteria.

Actually, the digital critical editing situation is even worse. The work of critical editors has never been a simple “write and print” process in itself: critical editors have to collect, describe, analyze, compare a lot of data sources and make many difficult choices. But whoever wants to make a modern critical edition, relying on digital tools, must take, roughly, the following steps:

1. **Source coding:** transcribe all the data into digital language code (for instance, HTML, XMLs like TEI, TeXs...);
2. **Data processing:** order, link, compare, manipulate the acquired data in order to make the best critical edition;
3. **Desktop publishing:** set up the pagination, the layout, etc. – on the whole, set up the readability of the finally published edition (cf. Adobe’s software InDesign).

That is a lot to do, as you can see: the help from suited software is dramatically needed. Actually, the ultimate TEI editor, which supports all of the abovementioned steps, is the powerful `<oXygen/>`⁸. But `<oXygen/>` is a complex and advanced tool for expert XML coders. In order to use it, a very comprehensive knowledge of XML with hard training on its interface is mandatory. It is self-evident that such a software cannot win the challenge of “simplicity, intuitiveness and accessibility”, al-

though (or, rather, in that) this is targeted and well-suited for experts.

In fact, we think that a real improvement in critical editing would be to create a software that adheres to the following requirements:

- Does not require an advanced knowledge of coding languages;
- Allows comprehensive yet intuitive data processing;
- Supplies us with a comprehensive yet intuitive GUI (Graphical User Interface), not only in order to work with it, but also in order to publish and read the final product with it.

Limitations of interfaces

On the topic of interfaces, although TEI is “only” an XML language (therefore, its source code requires external tools for its optimal visualization and profitable reading), the *TEI consortium* has provided its tools for visualization.⁹ But these tools themselves require expert managing and allow only conversions to other file formats. Luckily, the above-mentioned online projects provide their own interfaces, but none of these interfaces manages to be comprehensive, yet easy to read; most of them are “flat” interfaces, without navigation panels nor customizable views, unable to provide a clear overview of the critical text.

We think that their main flaw is a design heavily influenced by paper text layouts. It is a “psychological” limitation: to think of digital “pages” (the word “pages” itself is misleading!) as paper pages, readable only in a “fixed” order, strictly “constrained” by the structure of a paper source. This

⁸ <http://www.oxygenxml.com/>.

⁹ <https://github.com/TEIC/Stylesheets>.

approach is wise when we must digitalize ONE source, without appending or linking many data to it; but if we need to make a FULL critical text based on collation/comparison and switching of multiple sources, and/or a school text equipped with lots of information and links to other sources, the “paper approach” to text visualization turns out to be not only limited, but also counter-productive.

Fact is, a critical text published on paper is subject to natural limitations compelling the editors to omit informations for the sake of readability, especially by “eliminating” the sources believed to be *deteriores* (i. e. less important). This affects both “positive” and “negative” critical apparatuses, as well as it affects very different methodic approaches like Bédier’s (which privileges **one** supposed best source) or Lachmann’s (which selects the best witnesses of every text tradition branch, **discarding every supposed secondary copy**).¹⁰

Therefore, while trying to achieve an impossible compromise between comprehensiveness and readability, the editors encounter either of the two outcomes:

- A lot of information about text tradition is lost, just like it happens with digital “lossy” compressions such as MP3 or JPG;
- The apparatus is made to grow into a confusing proliferation of symbols and abbreviations, losing clear reference to their sources and meaning.

¹⁰ We will not discuss here the well-known *vexata quaestio* about the pros and cons of every critical editing method – it will be sufficient to recommend the reading of fundamental works like Pasquali 1962 and Contini 1986, as well as the reading of recent overviews like Meneghetti 1996 (indeed, this essay goes beyond the Italian critical editing, and takes into consideration the main European approaches).

It is no accident that in the specific field of Classical Humanities, many scholars still show a **bad attitude** (rightly pointed out by Monella 2012) **to underestimate and to “sacrifice” the natural plurality of text sources** (“natural”, because a text is often multiplied and adjusted as much as possible from the very beginning in order to reach and fit many different hearers and/or readers¹¹), considering the “textual variance” as an *hindrance*, which keeps them out from a supposed “unique original source”, or as a mere *tool* at its best, by which they try to rebuild this “unique original source”: we think this is the consequence of being almost unconsciously constrained by the “paper layout”, which heavily affects the data organization of their editorial work.

The history of Plato’s editions provides good examples of the abovementioned attitudes: on the one hand, we have Stallbaum’s edition¹², which had the undoubted merit of providing a complete collation of many manuscripts, even in spite of readability; on the other hand, we can observe the very strong, indeed excessive tendency of Martin Schanz to eliminate lots of manuscripts, judged as *descripti*. Schanz’s axe fell on many manuscripts, but he himself, years later, or, more often, other scholars recognized them as primary witnesses: we think, for example, to *Marcianus Graecus Appendix Classis IV 1* (Plato’s cod. T, middle of X cent.), of which Schanz in 1874 claimed that it was the most important among a group of twelve mss, but that, in 1876, he himself thought to be the best of

¹¹ The ancient Greeks were much more sensitive to this issue: cf. Plato’s observations about “fitting discourses” in *Phaedrus* 275 e.

¹² Stallbaum 1827-60.

his «schlechte Handschriften-Klasse», to the point that, in 1877, he dedicated to this codex an entire monograph;¹³ or to *Vindobonensis Suppl. Gr. 7* (cod. W, end of XI cent.), which was recognized as a primary source only in 1892 by Král and systematically collated only in 1959 by Dodds.¹⁴

The importance of having at our disposal comprehensive information is shown by the following two examples, describing the effects of missing or neglected data:

1. Codex *Vaticanus Palatinus gr. 173* (Plato's P, middle of X cent.) was firstly recognized as a primary source by Dodds (1959, 40), but only after the accurate analysis of Menchelli (1991) this manuscript has been considered the oldest representative of the so-called third family and is consequently collated by modern editors. The lack of a full knowledge of this manuscript has led to a curious accident. In 1976, Pintaudi re-printed a strange text containing a set of platonic scholia and of etymologies drawn from *Cratylus*, transmitted by two manuscripts of the Palaeologan Age. The scholar thought that this text was a sort of lexicon, but thanks to a lucky chance, Cufalo (2015) was able to recognize that this “lexicon” is nothing else but a copy, yet the first recognized *apographon* of P: in fact, during the summer of 2003, he collated the whole Vatican manuscript and copied large sections of it and especially its pages with the *excerpta* from *Cratylus* which have allowed the identification.

¹³ See Schanz 1874, 84; Schanz 1876, 669-670; Schanz 1877.

¹⁴ See Král 1892; Dodds 1959.

2. In Plato's *Phaedo* 81 c 11, both the Oxford editors (not only the classic Burnet's edition, but also the more recent one by Strachan) did not mention that only the *Venetus Marcianus graecus App. Class. IV, I* has the accepted reading *αἰδοῦς*: perhaps they overlooked the widespread variant *ἀειδοῦς* as a negligible scribe blunder. But this variant should be handled with care: it implies a typical *Phaedo*'s pun using the words *ἀϊδής* (“invisible”), *ἀειδής* (“formless”) and *Αἰδης* (“Hades / the nether world”), and it is strongly supported by indirect tradition (Stobaeus's *Anthology*, and a IX-X cent. arabic translation quoted by al-Bīrūnī¹⁵ having the undebatable *مما لا صورة له*, “of the formless”).¹⁶

The above examples have made clear, we hope, how the omission of information can slow down the development of our knowledge: some data could not seem useful **today**, but **tomorrow** they could lead to new discoveries!

To sum up, we should be very careful before “eliminating” a variant or a source, even if it is confirmed that a source is a copy of another, for many reasons:

1. the instance of a single source copied from another single source, without intervention of a third source or more (i. e., the so-called *contaminatio*), is rather unfrequent;
2. the assumption that the original text should be one ultimate manuscript is easily undermined by the occurrence of drafts, multiple redactions and texts with so-called “adia-

¹⁵ *India*, 32, 14 ss. ed. Sachau.

¹⁶ Muggittu 2007, 30.

phore” variants (f. i. a poem whose wording was slightly adapted by its author to a different audience);

3. **Every variant has its own value and history** that goes even beyond the histories of its material sources.

In such a perspective, focused both on the history of the “migrations” of single variants or variant groups from a source to another (sometimes “landing” on the margins, sometimes merging with the source’s main text) and on the history of the sources even the so-called *contaminatio* between sources would not be necessarily a dreadful disgrace blurring the derivation of a source from another.¹⁷ Rather, the contamination should be frequently acknowledged as a lucky rescue of very ancient and valuable variants, which found a safe harbour inside the “contaminated” late source. In such a perspective, to collect carefully and thoroughly information about every variant and its context with the help of devices and data banks not constrained by the “paper layout” is obviously mandatory, as well as possible at long last.

An overview of the “HYPER solution”

Background

Of course, *HYPHER*’s concept is influenced by our experiences with excellent codes and software.

In the beginning there were the *Thesaurus Linguae Graecae* (*TLG*)¹⁸ and *Perseus Digital Library*:¹⁹ the former (1971-)

is a very comprehensive digital data bank of all the ancient Greek literature; the latter (1985-) is a younger data bank, less comprehensive than *TLG*, but free, open-source, link-oriented (for example, it links an original text with many translations) and open to more fields (like Arabic literature, Germanic, etc.).

Unfortunately, these are the main limitations of *TLG* and *Perseus*:

1. neither *TLG* nor *Perseus* are built upon digital **critical** texts;
2. *TLG* does not allow any editing, whereas *Perseus*, although its code (*Perseus’ Java Hopper*) and texts are open-source, requires advanced coding skills for editing and customizing.

Nevertheless, the simple fact that *TLG* was initially designed to be a CD-ROM that could be used in local computers over the years boosted the development of many “*TLG* readers”, local software tools able to read and show the *TLG* contents. In our opinion Peter Heslin’s *Diogenes*²⁰ stands out among them, because it satisfies the following criteria:

- is open-source and free;
- can be easily installed into many Operating Systems (Linux, Windows, Mac);
- has a built-in morphological parser able to recognize the speech parts with good accuracy;
- links three dictionaries (Greek, Latin, English), taken from *Perseus*, directly to every *TLG* CD-ROM’s word;

¹⁷ See the famous formulation by Maas 1950, 31: “Gegen die Kontamination ist noch kein Kraut gewachsen”.

¹⁸ Cf. history details in <http://www.tlg.uci.edu/about/history.php>.

¹⁹ <http://www.perseus.tufts.edu/hopper/>.

²⁰ <https://community.dur.ac.uk/p.j.heslin/Software/Diogenes/>.

- has a very simple and intuitive interface, almost “point and click” (for example, if you click on a word the dictionary will be automatically opened beside the main text).

Although *Diogenes* cannot overcome the limitations of *TLG* and *Perseus*, it remains a prime example of a good “digital native” **reader**.

However, in order to fill the gap caused by the absence of digital critical texts, an **EDITOR** is required.

The first worthy of mention is Stefan Hagel’s *Classical Text Editor (CTE)*,²¹ an impressive critical text processor with a very advanced variants management; *CTE* can also export to TEI and HTML.

Still, *CTE* isn’t free and not very user-friendly, it is primarily designed for paper critical editions and above all is a processor, not a desktop publisher: it does not face directly the issues of digital edition desktop publishing and visualization. Anyway, *CTE* remains a prime example of good critical text **processor**.

More oriented toward digital publishing is the *Maurolico Project*,²² to which we have personally contributed.

The *Maurolico Project* has its own tools to generate entire HTML interfaces starting from text files coded in a TeX language, *Mauro-TeX*, specifically designed for variants management: *de facto*, an interesting alternative to TEI. But this project requires an update to the latest HTML standards; besides, there are still no visual editors designed to code with *Mauro-TeX*, nor simple tools to compile it.

Last but not least, we would mention the Italian high school-oriented site <http://www.poesialatina.it>, maintained by Giuseppe Frappa. It is an intuitive, a **really** interactive and also immersive site: although this site is not devised for scholarly critical editing and its interface has the same “paper flatness” as aforesaid projects, it gives a “lightweight” and amazing view of translated classical texts together with a brilliant toolkit for teaching and student training.

We believe this Classical Humanities site is by far more “Digital Native” than many others, and we believe this site should be taken into consideration as paragon by whoever works in Digital Humanities field being careful to students and not-specialized end users. In our opinion, *HYPER* makes the next step along the same challenging path.

HYPER as an editor

HYPER is both an editor and a reader.

As an editor, *HYPER* provides an effective GUI (Graphical User Interface), mostly “point and click”, for easy and fast editing, also accessible to casual users. The user does not need to write code in order to append data to the text: *HYPER*’s interface uses a simple system of panels and menus activated by mouse clicks on every word of the text.

Whenever *HYPER* is set in “editor view”, after loading a text on which the user will work,

- by left-clicking a word, an “analysis panel” is opened (see Figure No. 1), into which the user can insert linguistic, translation and links data;

²¹ <http://cte.oeaw.ac.at/>.

²² <http://maurolico.free.fr/intro.htm>.

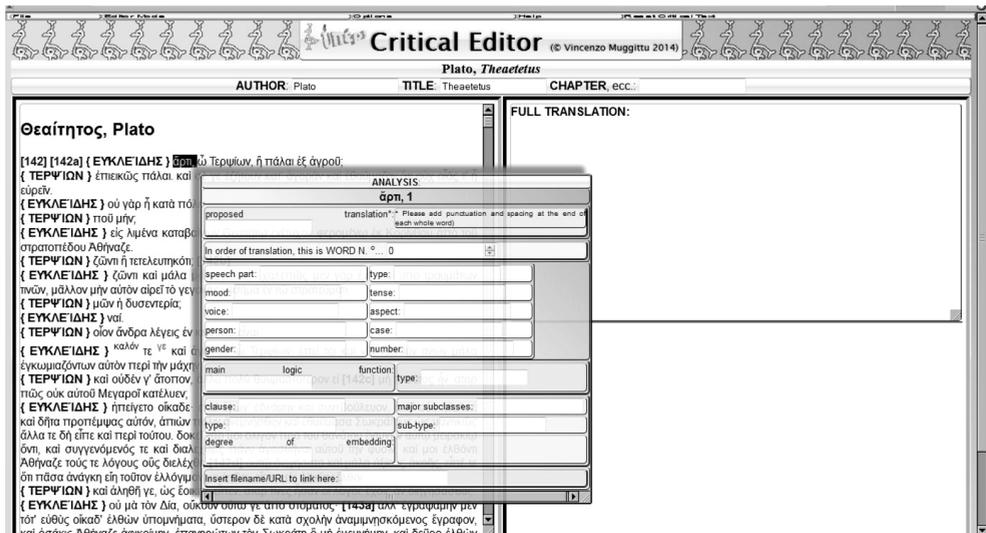


Figure No. 1: the analysis panel

- by right-clicking a word, a contextual menu is opened (see Figure No. 2), which gives an easy access

to several advanced operations, like paratext editing, source list editing and, above all, variants editing.

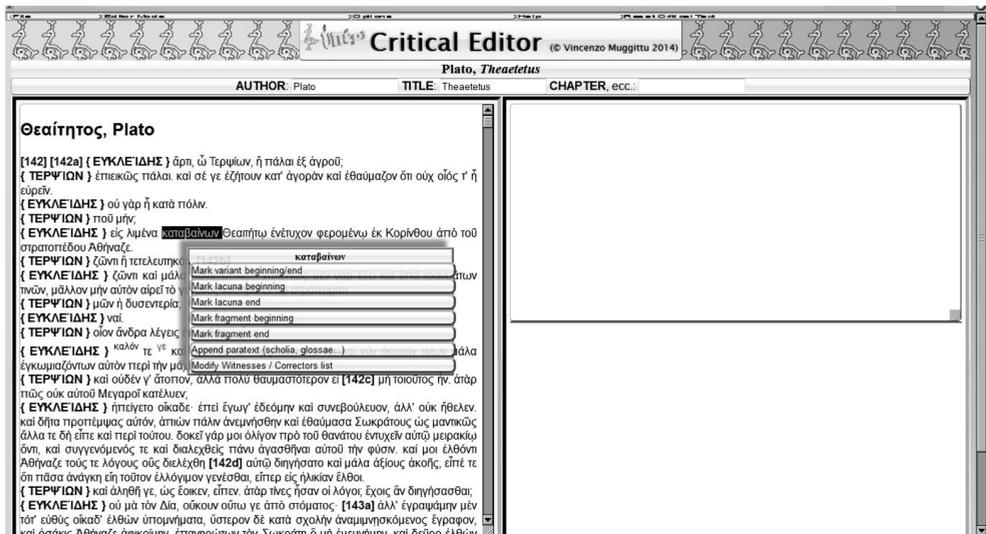


Figure No. 2: the contextual menu

The concept is very simple. For example, whenever the user wants to add a variant, he will mark an area of text as a “vari-

able area” with the help of the contextual menu (see Figure No. 3a).

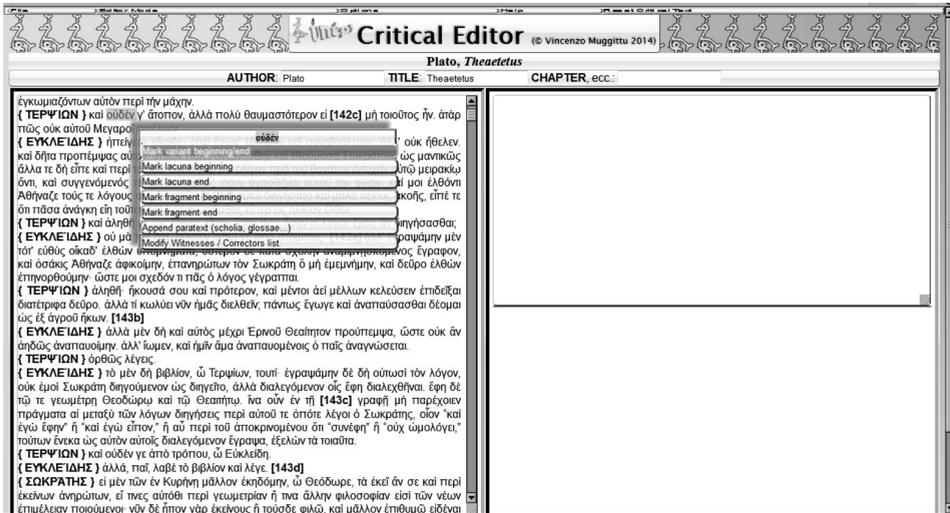


Figure No. 3a: marking a “variable area” with the contextual menu

Then, he will append the variants to this “variable area” with the help of a specific “critical apparatus panel”, which will

be opened by a simple double-click on the “variable area” (see Figure No. 3b).

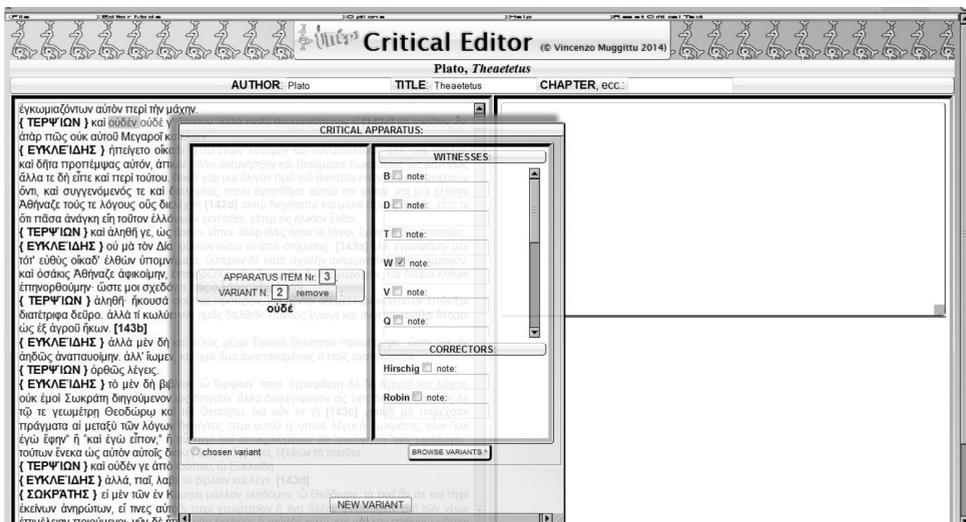


Figure No. 3b: adding variants with the critical apparatus panel

This panel automatically loads the source list; the user will only have to add the variant(s), select its source(s) and add optional comments.

As you can see, with *HYPER*, these advanced operations of critical editing **do not require advanced coding skills** (in fact, they require no coding skill at all); the user operates directly on the areas of text he needs, by simply clicking on them and activating the required panels.

When the user has gone through all

the steps, *HYPER* will easily save all the work in its own code, arranged for optimal reading.

HYPER as a reader

As a reader, *HYPER* provides an **effective visualization of critical texts**.

Its starting design is intentionally minimal: it has an almost hidden menu on top, titles, one main page division, apparently similar to the “text+facing page” layout (see below, Figure No. 4).

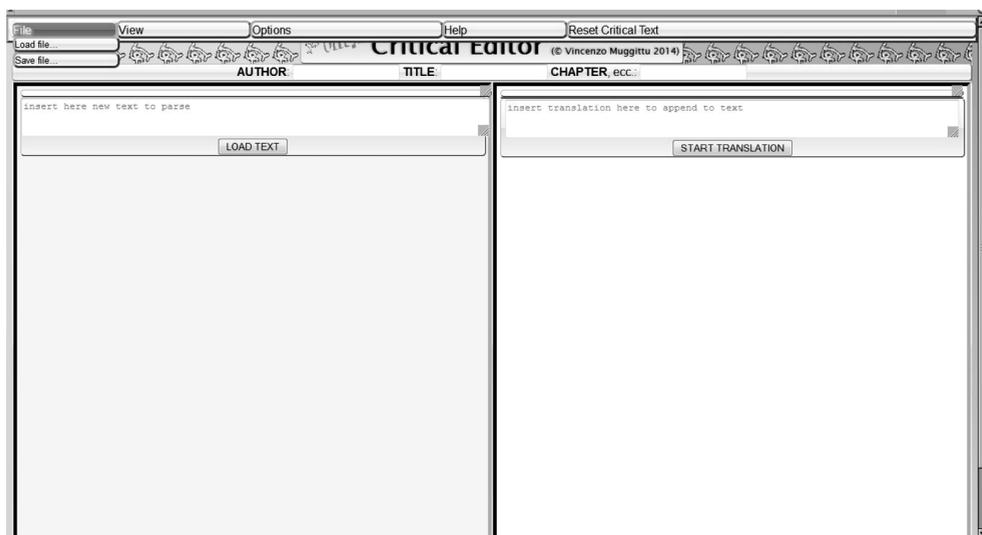


Figure No. 4: *HYPER's* starting layout

If the user loads a text, the FULL text will show up on the left section: it can be scrolled downside until its last world; also, the user can resize the text section as he likes (see below, Figure No. 5).

Whenever *HYPER* is set in “reader view”, on the right section, accordingly to his menu preferences, the user **can view multiple translations** (see below, Figure No. 6).

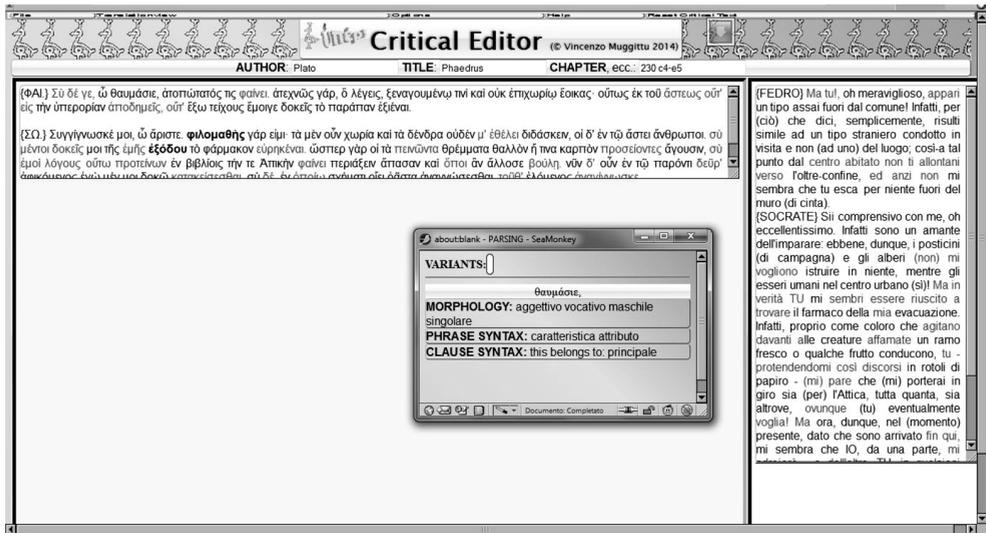


Figure No. 5: customizing *HYPER*'s sections

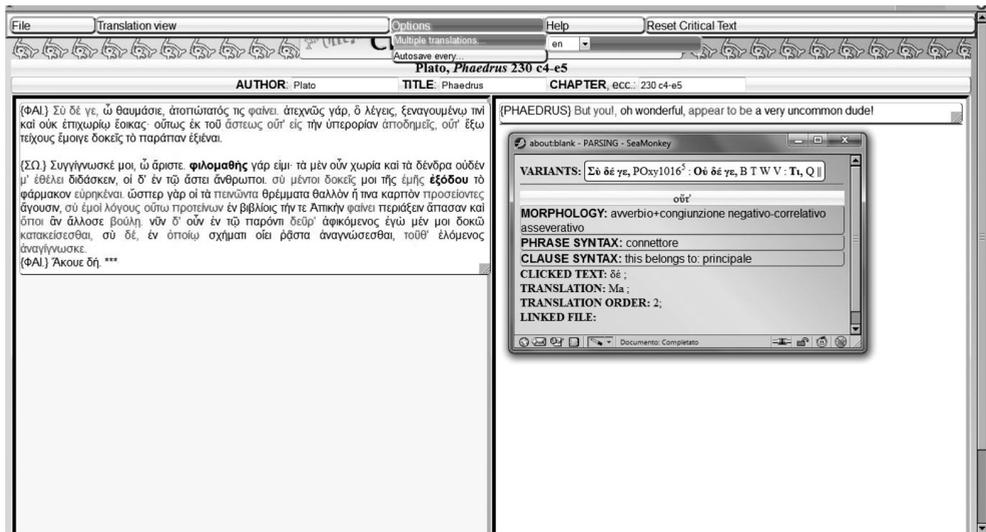


Figure No. 6: setting *HYPER* to show an english translation as well as an italian translation. Note the specific menu items above

The user can also **compare two different sources of the text**, loading one on the left and another on the right (see below, Figure No. 7).

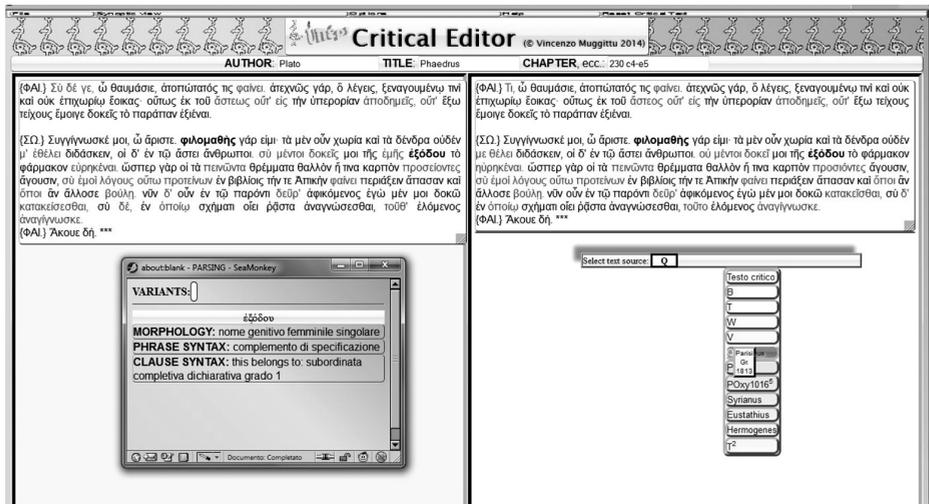


Figure No. 7: comparing a Plato's critical text [left] with the Q source [right]

By moving the mouse over a bold text, a small popper window is automatically opened: this window will show **local pages and/or external links with comments and/or other information about this text** (see below, Figure No. 8);

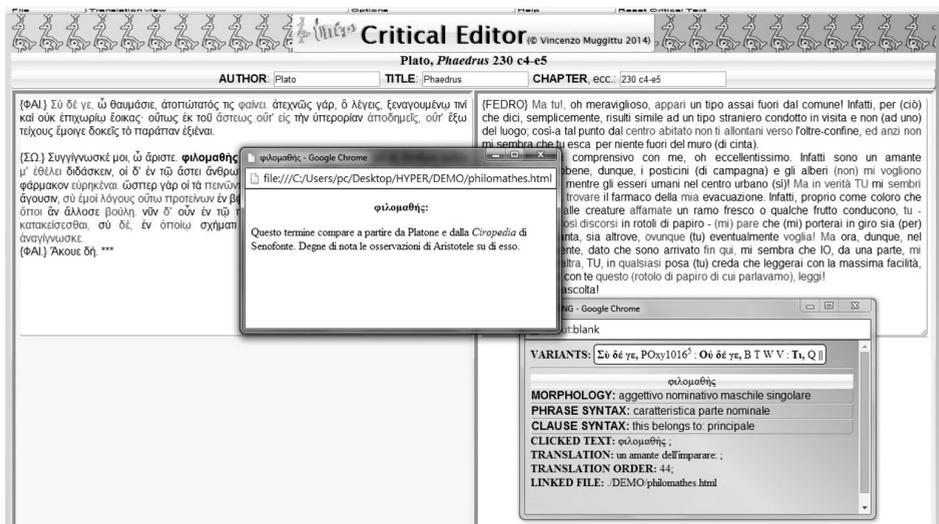


Figure No. 8: showing a simple comment from an HTML external file

By moving the mouse over any word in the text, a small popup window will **show linguistic information** (already seen in many figures).

By clicking a text with variants (red color), this window will also **show va-**

riant information; by moving the mouse over a critical symbol in this window, the full name of that witness will show up, **preventing any misunderstanding about the meaning of the symbol** (see below, Figure No. 9).

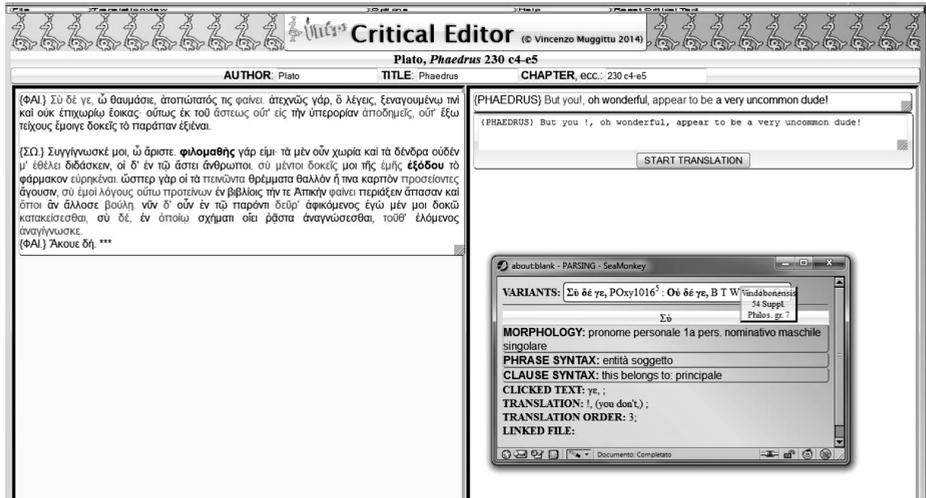


Figure No. 9: showing the full source name

Moreover, by clicking a witness symbol, all variants are automatically switched,

so that the user can see **the full text of that single witness** (see below, Figure No. 10).

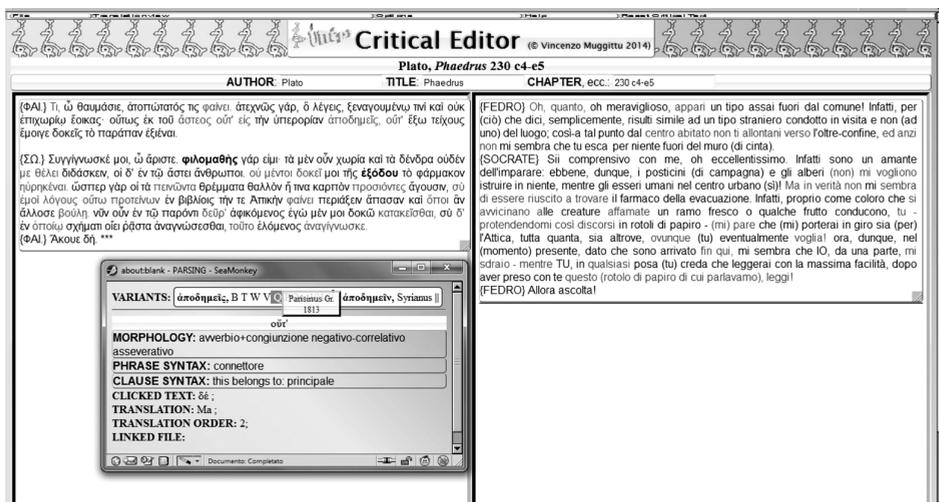


Figure No. 10: showing only the full text of a single source

Also, the translations are automatically adjusted, showing to the user the unique features of that witness in respect

of its content and meanings (see below, Figures Nos.11a-b).

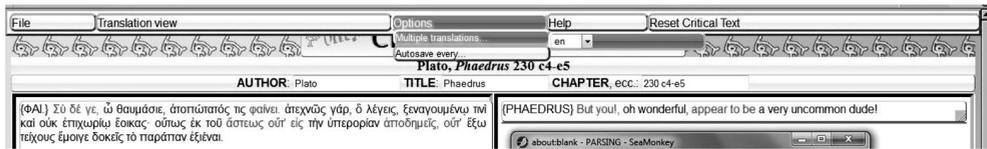


Figure No. 11a: (before source switching) Plato's critical text paragraph

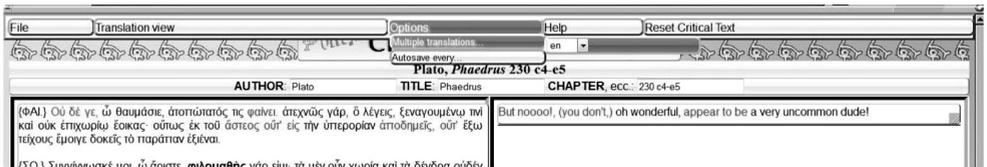


Figure No. 11b: (after source switching) Plato's source B paragraph: the translation has been automatically adjusted

The text includes also every source's paratext. The paratexts are hidden by default to prevent an excessive cluttering of the text; but, for example, whenever the user moves the mouse over some aste-

risks, popups with the full paratext and its sources show up, together with its translation below the main translation area (see below, Figure No. 12).

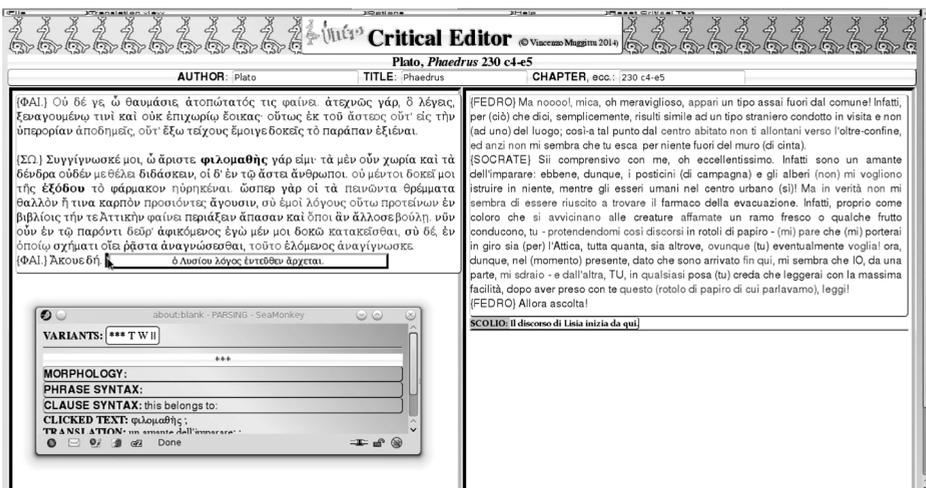


Figure No. 12: showing a paratext. Note its sources (blue popup window) and its translation (lower-right side)

In short, you have not only a full and easy-to-read overview of a critical text with every information linked to it but you have also **a clear and comprehensive overview of every single source** you'd like to see, together with **its own translations and links**.

To sum up, *HYPER* can help to “**easily rescue all the sources**”, restoring and showing their full roles in both text constitution and tradition history.

Moreover, even without using the advanced features of critical editing, *HYPER* is also designed to help high school students to train themselves in text translation, analysis and explanation, giving them a better text approach (“click-this-word-and-go!”) on tools like PCs, tablets and IWBs.

Some technical details

Actually, *HYPER*'s installation size does not exceed 300 kB. This could seem incredible, but it is the pure truth. How is this possible?

The fact is that *HYPER* relies totally on browser engines: the browsers generate *HYPER*'s interface. Also, the interaction between *HYPER* and browser is intentional because a browser is nothing more than an hypertext²³ reader; therefore, the browser supplies *HYPER* with its own tools for hypertext reading and manipulation. *HYPER* do nothing but tell the browser how it should visualize the critical text, as well as instruct it how the critical text is linked to other data.

²³ “Hypertext”, i.e., “digital text”, roughly (we use this word with its original meaning, as coined by the sociologist Ted Nelson, yet not excluding its semiotic meaning). *HYPER*'s name is born from this word and also from the Greek word ὑπέρ, i.e. “over”.

It could not be otherwise because true “digital native” critical text must have the same main hypertext properties:

- to be “dynamic” and “event-driven” – i. e., every hypertext part must properly and immediately react to “events”, such as a mouse click, finger touch, drag, scroll, swipe, etc.;
- these events must properly trigger every needed editing function (in “editor mode”) and/or links to information stored in other hypertexts and/or scripts (in “reader mode”);
- this information must be clearly visualized, without being invasive;
- also, its visualization must be customizable in order to fit every user's need.

We saw above that P. Heslin's *TLG* reader *Diogenes* already met many of these requirements; in fact, technically speaking, *Diogenes* itself is a series of scripts coded in PERL²⁴ language interacting with the browser. But *HYPER* is even more lightweight than *Diogenes* because it does not need neither the additional installation (in Windows) of a PERL interpreter nor the additional interaction with it.

Moreover, it is important to stress *HYPER*'s concept of data storing and organization: although *HYPER* loads smoothly a differently designed code like XML-TEI, it does not process all the loaded data as parts of the text: we mean, *HYPER*'s output doesn't mix up documents and information about them.

For example, *HYPER*'s output sets up every source/document's paratext and variant as parts of the text, applying a

²⁴ <http://www.perl.org/>.

“parallel segmentation method” just like TEI. We already mentioned some advantages of this method above. Here, we can add that this method has also the advantages of:

- **wisely acknowledging the variants** as vital and historical extensions of the whole text, as well as peculiarities of one source/document, and, above all,
- **simplifying the search queries:** whenever you search for multiple words or phrases, the variant will be correctly and easily returned together with the text which “surrounds” it.

But unlike the most common TEI’s output (usually, HTML code processed by XSLT stylesheets), *HYPER*’s output doesn’t set up links, source symbols or editor notes as parts of the critical text: *HYPER* manages all the information *about* the critical text as unprinted “variables”, “objects” and “properties” which prevent from “inflating” and “contaminating” the output text with data related but not belonging to it; moreover, such informations affect the visualization immediately and effectively, managing to be clearly and easily accessible whenever the user “activates” the text words to which they refer.

Let us explain in detail by using the following string of TEI code as an example:

You

```
<app>
  <lem wit = "#A">are</lem>
  <rdg wit = "#B">were</rdg>
</app>
```

good

Its markup tags (between angle brackets) mean that the text “You are good” has

“are” (editor’s choice) in its source/document labeled A, while it has the variant “were” in its source/document labeled B. That’s fine, but

- such a code implies the mixing up of Text and metatextual data (like the editorial symbols, and even editor’s own preferences), although these last ones are bracketed;
- such a code, by itself, does not allow the user to see only one selected source/document, or any other data type whenever/wherever he needs it.

The most common HTML output of such a code worsens the situation, because the informations about variants are usually recoded into a viral-like textual proliferation of `` tags and `@class` attributes:

You

```
<span class="app">
  <span class="lem">
    <span class="wit_A">are
  </span></span>
  <span class="rdg">
    <span class="wit_B">were
  </span></span>
</span>
```

good

Instead, *HYPER* manages every Source/Document’s symbol not as text, but as “variables” and “properties” ‘hidden’ into every variant; by doing so, whenever an event (like a mouse click) triggers a variant, this event will directly set many *HYPER*’s internal functions to load the properties of the variant as their parameters, affecting all the interface – it has already been demonstrated (Figure No. 10) how, after clicking the “Q” symbol, the text automatically becomes the “Q”-source’s text.

More technically speaking, *HYPER*'s default data management is JSON²⁵ based (we named this default as T[ext] E[ncoding]JSON),²⁶ as (see tables below):

XML-TEI	JSON
Requires a syntactical schema declaration in head of every file	Does not need schema pre-definitions at all
Its syntax is based on a huge set of elements and strictly defined attributes	Its syntax is based “only” on a hierarchy of nested objects, clearly distinct and fully customizable, marked simply by [], {}, “”, ., ; ,
Its syntax standard is very strict, and it allows to append certain information types only where its schema allows it	Its syntax is very flexible yet neat, and it allows to append every needed bit of information almost everywhere without any risk of confusion
Requires very complex stylesheets (XSLT) in order to read its output, e.g., as a “raw” HTML page	Is pre-set as a series of nested objects, variables and attributes which <i>HYPER</i> loads on the browser DIRECTLY as a real, fully fledged HTML dynamic hypertext

Here’s a sample code comparison:

XML-TEI	TEJSON ²⁷
<pre><?xml version="1.0" encoding="UTF-8"?> <?xml-model href="http://www.tei- c.org/release/xml/tei/custom/schema/relaxng/tei_all .rng" type="application/xml" schematypens="http://relaxng.org/ns/structure/1.0"? > <?xml-model href="http://www.tei- c.org/release/xml/tei/custom/schema/relaxng/tei_all .rng" type="application/xml" schematypens="http://purl.oclc.org/dsdl/schematron ?> <TEI xmlns="http://www.tei-c.org/ns/1.0"> <teiHeader> <fileDesc> <titleStmt> <title>Test</title> <author>John Doe</author> </titleStmt> <publicationStmt> <p><!-- supply publication information --></p> </publicationStmt> <sourceDesc> <listWit> <witness xml:id="A">Aianus 39</witness> <witness xml:id="B">Bianus 185</witness> <witness xml:id="C">Cianus 1</witness> <witness xml:id="D">Dianus 54</witness> </listWit> </sourceDesc> </fileDesc> </teiHeader> <text> <body></pre>	<pre>{ "author": "John Doe", "title": "Test", "witness": [{"source": "Aianus 39", "sigle": "A"}, {"source": "Bianus 185", "sigle": "B"}, {"source": "Cianus 1", "sigle": "C"}, {"source": "Dianus 54", "sigle": "D"}], "text": [{ "innerHTML": "you", "gramGrp": [["pronoun", "personal", "singular"]], "app": { "lem": [{ "innerHTML": "are", "gramGrp": [["verb", "present", "indicative", "active"]], "wit": ["A", "B", "D"] }], "rdg": [{ "innerHTML": "were", "gramGrp":</pre>

²⁵ It is not by chance that nowadays JSON’s (<http://www.json.org/index.html>) data-interchange format is increasingly appreciated as an alternative to XMLs.

²⁶ Following Muggittu 2015.

²⁷ Please note that this TEJSON code, while being equivalent to the facing TEI code, also includes linguistic information (“gramGrp”) which TEI would allow only in dictionaries at present.

<pre> <p>You <app> <lem wit = "#A">are</lem> <rdg wit = "#B">were</rdg> </app> good </p> </body> </text> </TEI> </pre>	<pre> ["verb", "simple past", "indicative", "active"] }, {"wit": ["C"]}]]], { "innerHTML": "good", "gramGrp": ["adjective", "qualifier", "singular"] }] } </pre>
--	--

As you can see, this TEJSON code is much more flexible, hypertext-oriented (see its output in all figures above), readable, shorter and faster than the TEI code.

To sum up, while TEI code manages data as markup, *HYPER* (although it can also smoothly import the TEI code) sets up a code which manages data as JSON scripts, in order to obtain the most performing and fittest output on browser's windows.

Conclusions (for now)

So, what does *HYPER* offer in order to overcome many longstanding problems of critical editing and text analysis, specially for our Digital Age?

HYPER offers:

- **Portability and Compatibility** – you can install *HYPER* within every operating system, using it on the most world-wide used browsers;
- **Lightness** – *HYPER* requires a very little installation space;
- **Fitness** – you can modify *HYPER*'s interface in order to suit your needs;
- **Simplicity** – *HYPER* eliminates the immediate need to have advanced coding skills;

- **User-friendliness** – *HYPER* helps to easily append, organize and read the information about every element of the text;
- **A real critical HYPertext** – *HYPER* allows you to interact with every text's element in many ways (clicking, touching, etc.);
- **Power** – *HYPER* manages a virtually unlimited amount of variants, paratexts, etc., yet without affecting the readability of the text;
- **Comprehensiveness** – thanks to *HYPER*, you, the editor, will no longer need to discard the exceeding information.

Of course the *HYPER* project is very young. The screenshots above (perhaps you already noticed they are taken from tests with many browsers and operating systems) are from a "beta version", and our work and testing is in progress; *HYPER* is bound to get even more improvement and features beside those already illustrated. But we think this is sufficient for now; with this project we hope to give our humble help and suggestions to other students, teachers and scholars, in order to go on the march into the future of critical editing and text analysis.

REFERENCES

- Contini, Gianfranco. 1986. *Breviario di ecdotica*. Milano: R. Ricciardi.
- Cufalo, Domenico. 2015. Il Lexicon quod Theaeteti vocatur e il codice Palatino greco 173 di Platone. *Lemmata*. Maria Tziatzi, Margarethe Billerbeck, Franco Montanari & Kyriakos Tsantsanoglou, eds. Berlin – Boston: De Gruyter, 452-472.
- Dodds, Eric Robertson. 1959. *Plato. Gorgias*. Oxford: Clarendon Press.
- Král, Josef. 1892. Über den Platocodex der Wiener Hofbibliothek suppl. phil. gr. 7. *Wiener Studien* 14, 161-208.
- Maas, Paul. 1950. *Textkritik*, 2nd ed., Leipzig: Teubner.
- Menchelli, Mariella. 1991. Il Vaticano Palatino gr. 173 (P) di Platone e il Parigino gr. 1665 di Diodoro. *Bollettino dei Classici* ser. III fasc. III, 93-117.
- Meneghetti, Maria Luisa. 1996. La Critica Testuale in Italia. *Quaderns d'Italià* 1. (<http://www.raco.cat/index.php/QuadernsItalia/article/view/119306/157416>)
- Monella, Paolo. 2008. Towards a Digital Model to Edit the Different Paratextuality Levels within a Textual Tradition. *Digital Medievalist* 4. (<http://www.digitalmedievalist.org/journal/4/monella/>)
- Monella, Paolo. 2012. Why Are There No Digital Scholarly Editions of 'Classical' Texts?. In: *Constitutio Textus: La "ricostruzione" Del Testo Critico – Constitutio Textus: Establishing the Critical Text*, Verona. (http://www.unipa.it/paolo.monella/lincei/files/why/why_paper.pdf)
- Pasquali, Giorgio. 1962. *Storia della tradizione e critica del testo*. Firenze: Le Monnier.
- Muggittu, Vincenzo. 2007. *PHAEDO ARABUS: testimoni arabi del Fedone*, Pisa: PhD thesis, unpublished.
- Muggittu, Vincenzo. 2015. HYPERcritical texts. In: *SEMINARIO DI CULTURA DIGITALE (Facoltà d'Informatica dell'Università di Pisa)*, 28-10-2015. (<http://www.labcd.unipi.it/seminari/vincenzo-muggittu-hypercritical-texts/>)
- Plato. 1827-1860. *Platonis opera omnia*. Edited by Gottfried Stallbaum. Leipzig: O. Holtze.
- Plato. 1900. *Platonis Opera*. Edited by John Burnet. Oxford: Clarendon Press.
- Plato. 1995. *Platonis opera*. Edited by E. A. Duke, W. S. M. Nicoll, W. F. Hicken, D. B. Robinson, and J. C. G. Strachan. Oxford: Clarendon Press.
- Porter, Dot & Ben Withers. 2005. Using the P5 Chapter on Manuscript Description for Detailed Physical Description: A Potential Approach. (<http://www.tei-c.org/Membership/Meetings/2005/schedule.xml>)
- Bīrūnī, Muḥammad ibn Aḥmad. 1925. *Kitāb Abī al-Rayḥān Muḥammad ibn Aḥmad al-Bīrūnī fī taḥqīq mā lil-Hind min maqūlah maqbūlah fī al-'aql aw mardhūlah = al-Bīrūnī's India, an account of the religion, philosophy, literature, chronology, astronomy, customs, laws and astrology of India about A.D. 1030*. Edited by Eduard Sachau. Leipzig: Otto Harrassowitz.
- Salvagni, Chiara. 2012. Digital Critical Editions of Homer. (<http://www.digitalclassicist.org/wip/wip2012.html>)
- Schanz, Martin. 1874. *Studien zur Geschichte des platonischen Textes*. Würzburg: Druck und Verlag der Stahel'schen Buch- und Kunsthandlung.
- Schanz, Martin. 1876. Untersuchungen über die platonischen Handschriften. *Philologus* 35, 643-670.
- Schanz, Martin. 1877. *Über den Platocodex der Markusbibliothek in Venedig Append. Class. 4 Nr. 1, den Archetypus de zweiten Handschriftenfamilie*. Leipzig: Bernhard Tauchnitz
- Stührenberg, Maik. 2012. The TEI and Current Standards for Structuring Linguistic Data – An Overview. *Journal of the Text Encoding Initiative* 3. (<http://jtei.revues.org/523>)

„SKAITMENINĖS KILMĖS“ KRITINIAI LEIDIMAI IR DĖSTYTOJO RuošIAMA STUDIJUOJAMŲ TEKSTŲ ANALIZĖ: PROJEKTAS „HYPER“

V. Muggittu, D. Cufalo

S a n t r a u k a

Nors dabartiniai laikai dažnai yra vadinami skaitmenine era, vis dar egzistuoja daugybė objektyvių veiksmų, kurie trukdo mokslininkams, dėstytojams

ir studentams naudotis visomis skaitmeninio kritinio hiperteksto galimybėmis (turime galvoje visus teksto variantus, paratekstus, visus egzistuojančius

vertimus, lingvistinę teksto analizę, išnašų ir nuorodų terpimą), kurios neabejotinai pranoksta tradicinę kritinio teksto leidimo metodologiją. Šiame straipsnyje pristatoma nauja, tebetobulinama inovatyvi,

interaktyvi, draugiška vartotojui skaitmeninės leidybos platforma „Hyper“, kuri, straipsnio autorių nuomone, galėtų jau dabar būti veiksmingai naudojama auštosiose mokyklose ir universitetuose.

Gauta 2016 07 27

Priimta publikuoti 2016 10 18

Autorių adresai

Dr. Vincenzo Muggittu

Via Sassari, 40

56123 Pisa (PI), Italy

El. paštas: vincenzo.muggittu@tin.it

Dr. Domenico Cufalo

Via Vittorio Veneto, 18

56127 Pisa (PI), Italy

El. paštas: cufalo@gmail.com