

Andrej Pančur*

Sustainability of Digital Editions: Static Websites of the History of Slovenia – Sistory Portal

IZVLEČEK

TRAJNOST DIGITALNH IZDAJ: STATIČNE SPLETNE STRANI PORTALA ZGODOVINA SLOVENIJE – SISTORY

Prispevek izhaja iz stališča, da je pri digitalnih izdajah potrebno poskrbeti za čim bolj celovito digitalno trajnost tako podatkov kot predstavij, funkcionalnosti in programske kode. To je velik izziv predvsem za manjše digitalno humanistične projekte z omejenim financiranjem, ki ne omogoča dolgoročnega vzdrževanja tehnično zahtevnih digitalnih izdaj. Kot alternativno rešitev so v prispevku predstavljene rešitve, ki jih v zadnjih letih ponuja hiter razvoj statičnih spletnih strani. Digitalne izdaje, ki temeljijo na TEI, so s pomočjo osnovnih XML (XSLT) in spletnih tehnologij (HTML, CSS, JavaScript) kot statične spletne strani uspešno vključene v repozitorij portala Sistory. Vse statične spletne strani imajo tudi možnost dinamičnega prikazovanja vsebine.

Ključne besede: digitalne izdaje, digitalno kuratorstvo, TEI, XSLT, statične spletne strani

ABSTRACT

The contribution is based on the position that, with regard to digital editions, the highest possible degree of digital sustainability of data, presentations, functionalities, and programme code should be ensured. This represents a significant challenge, especially in case of smaller digital humanities projects with limited financing, which does not allow for the long-term maintenance of technically-demanding digital editions. The alternative solutions facilitated by the swift development of static websites in the recent years are presented in the

* Institute of Contemporary History, Kongresni trg 1, SI-1000 Ljubljana, andrej.pancur@inz.si

contribution. Digital editions based on the TEI have been successfully included in the Sistory portal repository as static websites, employing basic XML (XSLT) and web technologies (HTML, CSS, JavaScript). All the static websites also have the possibility of displaying dynamic content.

Keywords: digital editions, digital curation, TEI, XSLT, static website

Introduction

In digital humanities, the awareness of the importance of digital sustainability and permanent preservation of digital sources has been present for a long time (Schaffner and Erway 2014, 7). The research data of an individual project usually outlives the project in the context of which it has been collected, organised, and published. Therefore it is very important to ensure a high-quality and sustainable storage of digital data even after the project itself has been concluded.

In the recent years, the technical aspects of research data management and long-term archiving (metadata, archive formats, preservation media, and documentation) have been the subject of intensive discussions. Only lately, however, have we begun to realise that the preservation of data in accordance with the specific requirements of various scientific disciplines is almost more important for the high-quality management and reuse of this data (Moeller et al. 2018). While in the natural and social sciences the data from measurements and questionnaires is typically used, in the humanities the use of cultural objects like manuscripts, texts, pictures, and recordings is predominant. Moreover, researchers in humanities will usually additionally process, visualise, tag, link, and interpret digital cultural objects (DHd-AG Datenzentren 2017, 7).

Such data processing is particularly important in case of digital editions, which are a crucial part of digital humanities (Andorfer et al. 2016). Naturally, digital scholarly editions mostly consist of the research in the context of which different transcriptions, indications, analyses, explanations, etc., are produced. Such research data in particular should therefore be available to the research community in the long term and under open access conditions (Robinson 2016). In the case of digital editions, the encoded text is the most crucial long-term result of the project. The display of information is vital as well, as it represents the outlook of the project group on this information in the context of a certain application. However, it is not that every such outlook is unique in any way or even the only one possible. Instead, this information can be displayed in a variety of ways (Turska et al. 2016). With each new interpretation, the number of other potential user interfaces even increases. Each such presentation is thus a new research result that deserves long-term storage as well.

Therefore, research results in humanities consist not only of research data, but also of the presentation environment and the applications that enable data interpretation,

searching, filtering, browsing, and linking (DHD-AG Datenzentren 2017, 7). If we only stored research data, the initial presentation would be lost forever, even though the presentation represents an integral part of any digital edition (Fechner 2018). At the same time, we should not forget that the programming code used for the creation of digital editions is an integral part of the scientific argumentation as well, just like the digital editions (Andrews and Zundert 2016).

Sustainable storage of digital editions therefore represents a particularly significant challenge. Moreover, digital editions can be very different from each other in terms of their contents, appearance, and functionality. They mostly result from specific research projects with relatively limited financial and human resources at their disposal. As the project group members come from the field of humanities, they often lack the suitable technical expertise, which is why they mostly need to rely on external contractors when it comes to technical development. Furthermore, digital editions depend on the very swift development of online technologies and standards (Andorfer et al. 2016).

As the number of digital editions increases rapidly, the challenges involved in the sustainable storage of digital editions will only become greater in the future (Fechner 2018). In case of smaller digital humanities projects with limited financing, which does not allow for the long-term maintenance of technically-demanding digital editions, this represents a significant challenge and will continue to do so. In the continuation, I will present alternative solutions offered by the rapid development of static websites. In the recent years, static websites have become one of the main online development trends. It appears that this trend will also persist in the future (Williams 2019). In the present contribution, I will present the experience gained by generating static websites for the digital editions in the context of the activities of the Research Infrastructure of Slovenian Historiography, which, among other tasks, also manages the History of Slovenia – Sistory web portal.¹ In this regard I will restrict my article solely to the static websites generated from XML files, encoded in accordance with the Text Encoding Initiative Guidelines (TEI) (TEI Consortium 2019). In digital humanities, the TEI Guidelines are the *de facto* standard for text encoding, used by many different humanities projects and studies (Romary et al. 2017, 5).

In the chapter *Modern Static Websites*, I will first present the main advantages and disadvantages of this type of websites. In our case, we have decided to upgrade the basic XSLT Stylesheets of the TEI Consortium. In the *Sistory TEI Profile* chapter, I will present generic upgrade of the TEI Stylesheets. In the chapter *Configuring and Upgrading the Sistory TEI Profile* I will outline the project-specific options for upgrading this profile. In both these chapters, I will also discuss the various options of adding dynamic contents to static websites. In the chapter *Publishing Digital Editions* I will outline how these static websites can be made available to the public, in particular by their inclusion in the Sistory portal's digital repository. In the *Conclusion*, I will also mention a few more general findings.

1 "Research Infrastructure of Slovenian Historiography," *History of Slovenia – Sistory*, accessed April 15, 2019, <http://www.sistory.si/publikacije/?menuBottom=2>.

Modern Static Websites

All websites used to be static at first, which is why all of the digital editions in the field of digital humanities were initially created as static HTML websites. This was also true in case of the Slovenian scholarly digital editions (Ogrin and Erjavec 2009),² which have introduced the paradigm of digital editions in Slovenia (Ogrin 2005). The creators of these digital editions soon encountered certain shortcomings of static websites. In particular, they missed the option of carrying out structured text searches, adaptable URL query string parameters, and dynamic web content association. In the case of newer digital editions, they therefore opted for the Fedora Commons platform (Erjavec et al. 2011).

By that point, the internet had been, for a long time already, dominated by dynamic websites that had successfully replaced the outdated static websites, where the contents could only be altered by the developers directly editing the HTML code. By means of content management systems (e.g. the very popular WordPress, Drupal, and Joomla), dynamic websites have finally made it possible for technically unskilled users to start publishing on the internet.

The contents of dynamic websites are stored in databases. The server does not construct the contents until the user demands that a website be displayed, adapted to the demands of the user. A suitable programming language is used to communicate with the server. The biggest problem of such dynamic websites is that its technical solutions are often more complicated than the actual needs of their users.

Modern static websites, however, have been created as an answer to the problems exhibited by dynamic websites. Unlike the latter, static websites do not employ databases and server-side programming languages, but are simply a collection of HTML, CSS, and JavaScript files. Static websites therefore enjoy numerous advantages in comparison with dynamic websites (Rinaldi 2015):

- efficiency: as static websites do not require any databases or server-side processing, they are not in danger of becoming slow;
- hosting: because static websites do not rely on a server-side programming language, their hosting is simple and cheap. There are even free options, for example the GitHub Pages service;
- security: static websites do not require any databases or server-side programming languages that hackers could breach. Therefore such sites are safe until the files they consist of are stored securely;
- maintenance: as static websites do not rely on any databases, server-side programming languages, or content management systems, their maintenance is extremely simple;
- versioning: since static websites consist exclusively of text files, all of their versions can be quite simply stored in version control systems like Git.

2 *Scholarly Digital Editions of Slovenian Literature, eZISS*, accessed April 15, 2019, <http://nl.ijs.si/e-zrc/index-en.html>.

These reasons are particularly important to ensure the sustainability of digital editions. The use of standard formats like TIFF and JPEG for digital photographs, HTML and XML for texts, and so on, ensures that the digital editions created will remain readable and useful for a long time to come (Rosselli Del Turco 2016). Consequently, this paradigm started to be emphasised in other similar projects in the field of digital humanities as well (Viglianti 2017; Daengeli and Zumsteg 2017; Diaz 2018).

These reasons, however, are less convincing in case we expect digital editions to contain user-generated contents as well. Therefore, static websites are not appropriate for all digital editions in the field of digital humanities, as such solutions will often fail to satisfy the needs of the creators and users. On the other hand, countless digital projects do not call for very complex content and its display. In such cases the existing solutions provided by static websites can be more than satisfactory, especially because modern static websites do not completely lack the option of adding dynamic contents. In reality, static websites have only experienced their renaissance with the appearance of various services and programming solutions that allowed such websites to include dynamic contents.

Modern static websites are no longer coded manually, but are instead generated by employing static website generators. Nowadays, the selection of such generators is extremely broad. One of the most popular is Jekyll,³ which is also used in the creation of GitHub pages. Thus its use has also spread to humanities (Visconti 2016). Static website generators assume that the users will write the contents using text formatting syntax like Markdown markup language, which is very popular among developers.⁴ These formats can then be converted to HTML sites with a website generator and then published online. However, the Markdown syntax is very deficient and only allows for basic content publishing. As such, it is inappropriate for the tagging of complex humanities texts. Consequently, humanities texts are most often encoded with Extensible Markup Language (XML). Furthermore, XSLT (Extensible Stylesheet Language for Transformation) is used as a tool for XML conversion. Together, these are the key technologies employed by digital humanities (Flanders et al. 2016). As the use of XSLT transformations is often very similar to static site generator conversions, we can describe XSLT as a “modern, efficient static site generator” as well (Kraetke and Imsieke 2016).

Sistory TEI Profile

For many years, the TEI Consortium has been regularly maintaining and updating the XSL Stylesheets, which can be used to generate, on the basis of TEI documents, not only (X)HTML websites, but also many other formats, including LaTeX, XSL-FO, EPUB, DOCX, and ODT. These XSL stylesheets are freely available from

³ *Jekyll • Simple, blog-aware, static sites*, accessed April 15, 2019, <https://jekyllrb.com/>.

⁴ *Daring Fireball: Markdown*, accessed April 15, 2019, <https://daringfireball.net/projects/markdown/>.

the GitHub repository and regularly updated in accordance with the new versions of the TEI Guidelines.⁵ Not only is the relevant written documentation very good, but the programming code comments are exemplary as well. XSLT stylesheets are also used, among other things, to generate the static website for each version of the TEI Guidelines.⁶

Most importantly, by means of custom profiles, the XSLT stylesheets of the TEI Consortium allow for very flexible adaptations to different project requirements. In fact, the XSL Stylesheets for TEI have been written with the intention of being as adaptable as possible. Numerous parameters exist that can be configured according to preferences. The stylesheets contains many variables and templates, which can be adapted to specific requirements. The authors of the code even thought of empty (hook) templates, to which custom contents and XSLT programming code may be added. I have made use of all these options when writing the Sistory profile for the XSLT stylesheets of the TEI Consortium. (Pančur 2019a)

Initially, I based the creation of these profiles on the needs of the Research Infrastructure of Slovenian Historiography for flexible and prompt publication of our technical documentation online. In the context of the Research Infrastructure, my colleagues and I are managing the History of Slovenia – Sistory portal, which also contains a repository and digital library. Therefore we have decided to include these digital editions into the existing infrastructure as intensively as possible. Until 2016, the static websites of these digital editions had been stored on an additional www2 server of the Sistory portal,⁷ while the digital library itself had only stored the metadata about the digital editions and links to these static sites. After the upgrade of the Sistory portal in 2016, we could start storing the HTML and all other files related to these digital editions directly in the repository and the digital library.

Due to the desire to maximize the integration of digital editions into the Sistory portal, I also tried to bring the external appearance of digital editions as close as possible to the user interface of the portal. As an example, Figure 1 shows a snapshot of the home page of the portal between the years 2012 and 2016, and in Figure 2, the user interface of the digital edition of 2014.

5 *TEI XSL Stylesheets*, accessed April 15, 2019, <https://github.com/TEIC/Stylesheets>.

6 "P5: Guidelines for Electronic Text Encoding and Interchange," *TEI: Text Encoding Initiative*, accessed April 15, 2019, <https://www.tei-c.org/release/doc/tei-p5-doc/en/html/index.html>.

7 *www2.Sistory.si*, accessed April 15, 2019, <http://www2.sistory.si/>.

Figure 1: Home page of the History of Slovenia – Sistory portal of 2016

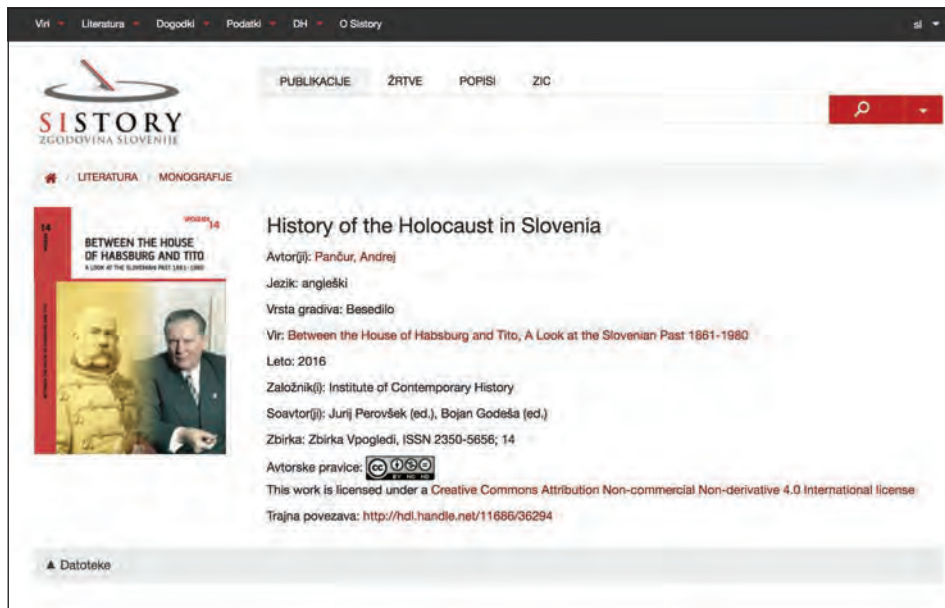
Source: Spletni arhiv Narodne in univerzitetne knjižnice, accessed April 10, 2018, <http://nukrobi2.nuk.uni-lj.si:8080/wayback/20160225143401/http://www.sistory.si/>.

Figure 2: The 2014 digital edition user interface

Source: (Gašparič 2014), accessed April 10, 2018, http://www2.sistory.si/publikacije/monografije/Gaspacic_Parliamentaria1/ch01.html.

Even though the colour scheme is identical and the layout of the logo, the search bar, main top navigation menu, and the contents are very closely modelled after the Sistory portal, the user interfaces are nevertheless not the same. At the time, the user interface of the portal was still based on the old HTML 4 technology, but I had already started to use responsive website design and HTML 5 for the digital editions. In this regard, I decided to use the responsive front-end framework ZURB Foundation.⁸ I keep my adaptations as well as CSS and JS additions in the GitHub repository. (Pančur 2019b) As the use of this framework turned out to be extremely useful, we also included it in the new Sistory portal in 2016. Subsequently I also adapted the appearance of the digital editions to the new portal design (compare Figures 3 and 4).

Figure 3: Top navigation menu, search bar, and metadata page of the Sistory portal



Source: (Pančur 2016).

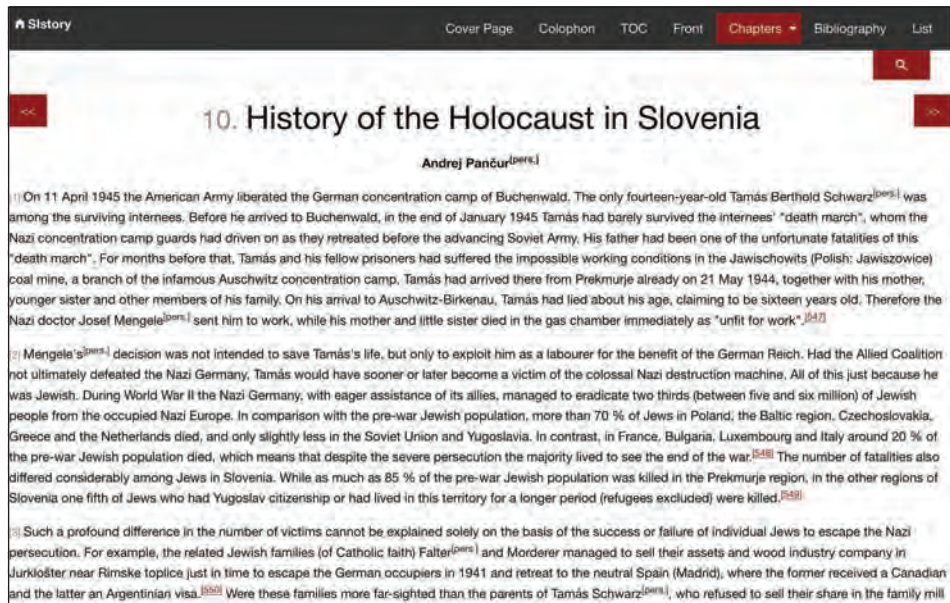
Apart from the originally envisioned technical documentation, we soon also started to publish other sorts of publications – in particular monographs, collections of scientific texts, and magazines – online in the HTML format. Therefore I reconfigured the Sistory TEI profile with the aim of facilitating the publication of these sorts of digital editions. The profile allows for the transformations of:

- individual TEI documents;

⁸ Foundation: The most advanced responsive front-end framework in the world, accessed April 15, 2019, <https://foundation.zurb.com/>.

- several TEI documents from a shared TEI corpus. In this case, each TEI document needs to be converted separately. The TEI corpus itself and its `<teiHeader>` need to be converted separately, as in this manner a common cover, colophon, and tables of contents are generated.

Figure 4: The 2016 digital edition user interface



Source: (Pančur 2016), accessed April 15, 2019, <http://www.sistory.si/cdn/publikacije/36001-37000/36294/ch10.html>.

The digital edition's main navigation menu is located at the very top of the web page, as horizontal navigation with a drop-down menu. The structure of this navigation reflects the structure, sections, and divisions of the individual TEI documents. In the continuation I will briefly outline the possible content sections of the navigation as well as the TEI document. In practice, no TEI document contains every single one of these sections. Instead, the authors of TEI documents can use and arrange them completely in accordance with their needs.

The central part of the content is always contained within the `<body>` element. The main content must be contained within a single or several `<div>` elements with the obligatory attribute `@xml:id`. Each `<div>` element represents its own division of the content or chapter. Therefore the navigation bar's single drop-down menu displays all of the `<div>` divisions contained within the `<body>` element. A variety of contents, encoded in the relevant TEI document within the `<front>` and `<back>` elements, may also be accessible before and after this part of the drop-down menu. Figure 5 thus illustrates all of these main content sections.

Figure 5: The main content sections of a TEI document

```

<text>
  <front>
    <titlePage>
      <docTitle>
        <titlePart>Title of digital edition</titlePart>
      </docTitle>
      <docAuthor>Author of digital edition</docAuthor>
      <docEdition>Document edition</docEdition>
      <docImprint>
        <pubPlace>Publication place</pubPlace>
        <docDate>Publication date</docDate>
      </docImprint>
      <graphic url="url_of_cover_page_image.jpg"/>
    </titlePage>
    <div type="preface" xml:id="prf-01">
      <!-- Introductory chapter -->
    </div>
  </front>
  <body>
    <div type="chapter" xml:id="ch01">
      <!-- Chapter with main content -->
    </div>
  </body>
  <back>
    <div type="bibliogr" xml:id="bibl01">
      <!-- Bibliography -->
    </div>
    <div type="appendix" xml:id="app01">
      <!-- Appendix -->
    </div>
    <div type="summary" xml:id="sum01">
      <!-- Summary -->
    </div>
  </back>
</text>

```

Only `<titlePage>` is obligatory, because it is converted to the default start page (index.html) and, as such, accessible through the navigation bar – at the very top, as the Title Page. The `<front>` element may contain one or several `<div>` elements, which represent the introductory chapters section in the navigation. The `<back>` element includes three possible content sections (bibliographies, annexes, summaries), which is why they must always be assigned the appropriate `@type` attribute. Each of these sections can consist of one or more chapters. In most cases, the conversion of the content of these divisions is based on the standard XSLT stylesheets of the TEI Consortium, which I have only partly adapted to the needs of our own digital editions. I have written the transformations for the generated `<divGen>` divisions from scratch.

All of them have been included in the Sistory TEI profile. These generated divisions can be included in the `<front>` (Figure 6) or the `<back>` element (Figure 7), and each of the `<divGen>` elements must include a `<head>` with an arbitrary division title. These titles are then included in the digital edition's navigation.

Figure 6: The list of all possible generated `<divGen>` divisions, contained in the `<front>` element

```
<front>
  <divGen type="cip" xml:id="cip">
    <head>Colophon</head>
  </divGen>
  <divGen type="teiHeader" xml:id="teiHeader">
    <head>TEI header</head>
  </divGen>
  <divGen type="toc" xml:id="id-toc">
    <head>Table of contents</head>
  </divGen>
  <divGen type="toc" xml:id="id-images">
    <head>List of images</head>
  </divGen>
  <divGen type="toc" xml:id="id-charts">
    <head>List of charts</head>
  </divGen>
  <divGen type="toc" xml:id="id-tables">
    <head>List of tables</head>
  </divGen>
  <divGen type="toc" xml:id="id-titleAuthor">
    <head>Table of contents where the name
      of the author is also displayed</head>
  </divGen>
  <divGen type="toc" xml:id="id-titleType">
    <head>Simplified table of contents</head>
  </divGen>
  <divGen type="search" xml:id="search">
    <head>Search</head>
  </divGen>
</front>
```

Unlike the aforementioned `<div>` elements, where the use of `@xml:id` identifiers is merely recommended (the HTML files that contain these divisions are named after these identifiers), in case of generated divisions they are obligatory and also have a semantic meaning that is of key importance for their conversion. The `@type` attribute defines the main category, which is particularly highlighted in the horizontal navigation. The `@xml:id` attribute more precisely defines the subcategory, shown in the navigation drop-down menu. The most extensive category is the Table of Contents (TOC) group, which, apart from the various tables of the contents of chapters and

subchapters, also contains a list of tables, figures, and charts. In reality, the list of charts is merely a separate group of list of figures (`<figure>`), which includes figures with the `@type` attribute and `chart` value.

The `<back>` element involves only a single category of generated divisions that includes various lists of persons, places, and organisations. The generated divisions include all of the persons mentioned in the TEI document, encoded with the `<persName>` element, all places encoded with `<placeName>`, or all organisations encoded with `<orgName>`. All of the named entities, encoded in this manner, must also be assigned the `@ref` attribute, in order to refer to the appropriate canonical element in the list of entities (`<listPerson>` for persons, `<listOrg>` for organisations, and `<listPlace>` for places) in the TEI header (`<teiHeader>`). The `<placeName>` element's `@ref` attribute may also contain a reference to the GeoNames⁹ or DBpedia¹⁰ URI, where the Sistory profile can process the geographical coordinates and display them in the list of places.

Figure 7: The list of all possible elements for automatically generated text division `<divGen>`, contained in the `<front>` element

```
<back>
  <divGen type="index" xml:id="id-persons">
    <head>List of persons</head>
  </divGen>
  <divGen type="index" xml:id="id-places">
    <head>List of places</head>
  </divGen>
  <divGen type="index" xml:id="id-organizations">
    <head>List of organizations</head>
  </divGen>
</back>
```

As it is also possible to use the Sistory profile to convert the TEI documents from the TEI corpus, the `<divGen>` elements from the various TEI documents cannot possess the same `@xml:id` identifiers. Therefore the subcategories of the generated divisions are specified in such a manner that the subcategory's identifier is stated after the final hyphen of this identifier's value (see Figures 6 and 7, where the `id` before the hyphen in `@xml:id` attribute defines the arbitrary identifier, while the subcategory is stated after the hyphen).

The Sistory profile also allows for the display of dynamic contents. The Tipue Search engine is included as a basic functionality.¹¹ It can be included with a generated division (`<divGen>`) of the `search` type in the `<front>` element. Tipue Search is an open source jQuery plugin, which can be relatively easily integrated even in static sites. In

9 GeoNames, accessed April 15, 2019, <http://www.geonames.org/>.

10 DBpedia, accessed April 15, 2019, <http://wiki.dbpedia.org/>.

11 Tipue Search, accessed April 15, <http://www.tipue.com/search/>.

the graphical user interface, the search bar is located immediately below the bottom navigation, while the element `<divGen>` generates a `search.html` web page that includes a dynamic display of search results. The content of the TEI document is indexed, as a JavaScript object (JSON), in the file `tipusearch_content.js`, which needs to be located in the same folder as the `search.html` file. Content indexation takes place at the level of paragraphs (`<p>`), lists (`<list>`), tables (`<table>`), figures (`<figure>`), and all other possible TEI elements, which are direct child elements of the text division `<div>`. Therefore, all of these elements must include a `@xml:id` attribute for unique identifier. Lists are the only exception: when they do not possess the `@xml:id` attribute, whereas their child elements do, then the latter are indexed.

Configuring and Upgrading the Sistory TEI Profile

Much like the main XSL Stylesheets of the TEI Consortium, the Sistory profile has been created to allow for its adaptation to the requirements of any individual project. To this end, it includes a few original parameters of the TEI Consortium's XSLT stylesheets which affect the default stylesheet output, to which I have added a few new Sistory parameters. All of these parameters can be set up anew for each conversion, but it is more appropriate that new project profiles be created for each individual project. The conversion usually proceeds in the following manner: the project's custom profile imports the Sistory profile, which, in turn, imports the TEI XSLT transformations, and adds overrides (see Figure 8).

Figure 8: Chained XSLT conversions with additional profiles



For example, during conversion, the default Sistory profile thus expects every chapter or the first `<div>` text division to become a separate HTML web page. In this case, navigation through forward and back buttons is added to the web pages automatically. Unlike the original TEI transformations, this navigation also includes the `<divGen>` generated divisions. However, by changing the `splitLevel` parameter (originally a parameter included in the TEI conversions), it is possible to specify that subchapters also become separate HTML web pages. The forward/back and up/down navigation between the web pages has now been appropriately adapted. The current Sistory profile only supports a depth of three text divisions.

The `documentationLanguage` parameter may currently be used to specify the Slovenian, English, or Serbian navigation (in the Latin or Cyrillic script). By adding new translations to the `myi18n.xml` document, it is possible to further expand this localisation. The localisation of the Tipue Search engine has been suitably taken care of as well.

The Sistory profile also allows for the parallel display of the texts' various language versions. In this case, all of the main `<div>` text divisions and `<divGen>` generated divisions must contain `@xml:lang` attributes with the appropriate language code as well as `@corresp` attributes pointing at all the other language versions of the text in question (see Figure 9). Simultaneously, the `languages-locale` parameter must be set to the value `true`, while the `languages-locale-primary` parameter must specify the language code of the starting `index.html` file.

Figure 9: Localization and language setting in TEI document

```
<body>
  <div type="chapter" xml:id="ch01" xml:lang="sl"
      corresp="#en-ch01">
    <!-- Slovenian chapter with main content -->
  </div>
  <div type="chapter" xml:id="en-ch01" xml:lang="en"
      corresp="#ch01">
    <!-- English chapter with main content -->
  </div>
</body>
```

The display of the TEI document's metadata from the `<teiHeader>` element is similarly adaptable. This transformation can be initially specified by including the generic division (`<divGen>`), whose `@type` attribute value should be set to `teiHeader` (see Figure 6). The entire content of the `<teiHeader>` element is converted to `<dl>` (definition list HTML element), where `<dt>` (description term element) defines the name of the TEI element as well as the names and attribute values (element [attribute = value | attribute = value]), while `<dd>` (definition description element) defines the text contents of the TEI element. Of course, the definitions are appropriately nested. With additional parameters, this transformation can be configured in such a way as to

display the descriptive names of elements and attributes in the English or Slovenian language instead of their names.

Apart from this simple Sistory profile configuration, any additional XSLT transformation that can be completely adapted to the needs of an individual digital edition can be included during the conversion of a project. Simultaneously, by using various JavaScript libraries and plugins as well as web applications, it is also possible to enable additional dynamic content display. For example, in the case of the Sistory portal's digital editions, I have successfully used DataTables¹² to display large quantities of tabled data, Highcharts¹³ for charts, Google Maps for maps, and ImageViewer¹⁴ and Viewer.js for images.¹⁵ These are merely examples: there are alternatives, and every year many new possibilities emerge.

Figure 10: The simultaneous display of facsimiles, diplomatic transcription, and critical transcription of the *Kapelski pasijon* passion play.

The screenshot shows a web browser displaying the digital edition of 'Kapelski pasijon'. The page title is 'Vzporedni' (Parallel). Below the title, there are navigation options: 'Vrata prikaza (strani)', 'Način vzporednega prikaza (facs-dipl-crit)', and 'Prelom vrstice: Da / Ne'. The main content area is divided into three columns:

- Facsimile:** A large image of a handwritten manuscript page.
- Diplomatic Transcription (No 31):** Text in Slovenian: '(Big telen vte niedor, der erste Jud) Ufhanite gori vi moi tovarihi vsi , inā preitrahite Se tega kapelauza teh ludi , an je sam tukei katier je nas ikuea lojo zoperniu verju dol use kupei , mi ozhmo niemo fe **fueper ftati** izvorna oblika → regularizirano: fueperftati • lamina: zoperstati • NoSketch ... inu **fnathemi** fhtriki lbestu luesati
- Critical Transcription (No 31):** Text in Slovenian: '[Lustrinski prevod: Teda] padejo veria zembu.] 999] — Prvi Jud [222] Ustanite gori, vi moji tovari vsi! [223] Knā prestrašite se tega zapelavca teh ludi. [224] An je sam tukej. [225] katier je nas skuez sojo coprnju viju dol vse kupei. [226] Mi očmo njemo se zuiper stati [227] inu z našemi štriki zbestu zvezati.

Below the critical transcription, there is another section for 'No 32' with text: '— Xlta: Kaga vi jhete. — Dar Judem, mi jihemo Jesusa Nazarenskiga. All vi mena jhete , taku pužite **late** poiti'

Source: Kapelski pasijon, GitHub pages, <https://dariah-si.github.io/Kapelski-pub/>

12 DataTables: Table plug-in for jQuery, accessed April 15, 2019, <https://datatables.net/>.

13 Highcharts, accessed April 15, 2019, <https://www.highcharts.com/products/highcharts/>.

14 ImageViewer, accessed April 15, 2019, <http://ignitersworld.com/lab/imageViewer.html>.

15 Viewer.js, JavaScript image viewer, accessed April 15, 2019, <https://fengyuanchen.github.io/viewerjs/>.

Simultaneously, in 2017, with the publication of Saxon-JS¹⁶, the possibilities of dynamically displaying the contents of XML documents in static web pages have even improved. Saxon-JS is an XSLT 3.0 run-time written in pure JavaScript. It could contribute to XSLT once again becoming a client-side technology that works in a browser (Lumley et al. 2017). For digital editions, I have thus started to successfully use an Saxon extension function *ixsl:query-params*, which parses the query parameters of the HTML page URI. In the case of the *Kapelski pasijon* (The Železna Kapla Passion Play) digital edition, I have thus created and used the following parameters to generate a dynamic parallel display of facsimiles as well as the diplomatic and critical transcription: *type*, *mode*, *page*, and *lb* (line break). These parameters have allowed me to construct a dynamic display of extremely complex contents (Figure 10), which can still be optionally upgraded in the future digital editions.

Publishing Digital Editions

The default Sistory profile transformation generates all the HTML, JS, and any other files in a single folder. As the digital editions generated in this manner consist solely of static web pages, they can also be used on personal computers. In this manner it is possible to effectively test the digital editions even before publishing them online, where we can swiftly and simply publish them on any accessible servers. Additionally, the GitHub repository web pages are a free option that can also ensure an efficient version control.

However, the main purpose of Sistory profiles is to include digital editions directly into the Sistory portal's repository and its digital library. Thus we can efficiently store all of the digital editions' files by adding persistent Handle System identifiers and checksums for all the relevant files, as well as flexibly organise digital editions as one or several digital objects with one or several intellectual entities. Each intellectual entity has its own Handle identifier and metadata. It can include several files or none at all. The files belonging to an individual intellectual entity are located in the same folder. The path to this folder also includes the suffix of the Handle persistent identifier, which is, in the case of the Sistory portal, always a numerical value (e.g., for the suffix 555, the relative path would be */cdn/publikacije/1-1000/555/file*). Therefore, the Sistory XSLT profile must know the values of these identifiers in advance. Thus we can precisely determine, even in advance, whether the entire contents of a digital edition should be contained in a single intellectual entity of the Sistory portal, or whether various digital edition files should be included in various intellectual entities. These identifiers can be recorded among the rest of the metadata in *<teiHeader>*, within the *<publicationStmnt>* element, as a value of one or more *<idno>* elements. This element

16 "Saxon-JS," *Saxonica*, accessed April 15, 2019, <http://www.saxonica.com/saxon-js/index.xml>.

requires that the value of the *@type* attribute be specified as *sistory* or *si4*, while the *@corresp* attribute should point at all the appropriate *<div>* and *<divGen>* divisions whose content will be included in the intellectual entity with this identifier.

The Sistory XSLT profile is open source and available in the GitHub repository. (Pančur 2019a) Another GitHub repository also contains all of the digital editions currently kept on the Sistory portal. The project upgrades of the Sistory XSLT profile for each of these editions are available as well. (Pančur 2018) I regularly expand and maintain the Sistory profile in accordance with the changes of the TEI XSLT stylesheets.

Conclusion

There are several advantages to a digital editions infrastructure organised in this manner:

- using format that is most common in digital humanities: TEI XML (Neuefeind 2019, 221);
- using a single XML technology (XSLT) for various sorts of digital editions, which enjoys a wide support in the TEI community;
- the possibility of simply including JavaScript libraries and plugins;
- flexibly adding dynamic contents with Saxon-JS;
- in comparison with other technologies (dynamic sites), static sites ensure a relative sustainability and simple maintenance of digital editions;
- using Git version control to store the various versions of digital editions together with the software used to generate static websites;
- open access to the complete digital editions code in the GitHub and GitLab software development platforms;
- the possibility of sharing digital editions on the GitHub Pages and GitLab Pages, and, last but not least, the possibility of including them in the History of Slovenia – Sistory portal.

Acknowledgements

The work presented in this paper was supported by the Slovenian historiography research infrastructure (I0- 0013), and the Slovenian ESFRI infrastructures DARIAH-SI which are financially supported by the Slovenian Research Agency.

Sources and Literature

Datasets and Academic Software:

- Pančur, Andrej. 2018. *Electronic publishing on Sistory*. Distributed by GitHub. <https://github.com/Sistory/publications>.
- Pančur, Andrej. 2019a. *Sistory TEI Stylesheets*. Distributed by GitHub. <https://github.com/Sistory/Stylesheets>.
- Pančur, Andrej. 2019b. *Sistory: additional CSS and JS*. Distributed by GitHub. <https://github.com/Sistory/themes>.

Literature:

- Andorfer, Peter, Matej Đurčo, Thomas Stäcker, Christian Thomas, Vera Hildenbrandt, Hubert Stigler, Sibylle Söring, and Lukas Rosenthaler. 2016. "Nachhaltigkeit technischer Lösungen für digitale Editionen: Eine kritische Evaluation bestehender Frameworks und Workflows von und für Praktiker_innen." In *DHd 2016: Modellierung – Vernetzung – Visualisierung: Die Digital Humanities als fächerübergreifendes Forschungsparadigma: Konferenzabstracts*, 36–39. Universität Leipzig. <http://www.dhd2016.de/>.
- Andrews, Tara, and Joris van Zundert. 2016. "What Are You Trying to Say? The Interface as an Integral Element of Argument." In *Digital Scholarly Editions as Interfaces, International Symposium at the University of Graz, Austria*, 31–32. Graz: Centre for Information Modelling – Austrian Centre for Digital Humanities. https://static.uni-graz.at/fileadmin/gewi-zentren/Informationsmodellierung/PDF/dse-interfaces_BoA21092016.pdf.
- Daengeli, Peter, and Simon Zumsteg. 2017. "Hermann Burgers Lokalbericht: Hybrid-Edition mit digitalem Schwerpunkt." In *DHd 2017: Digitale Nachhaltigkeit: Konferenzabstracts*, 151–55. Universität Bern. <http://www.dhd2017.ch/>.
- DHd-AG Datenzentren. 2017. *Geisteswissenschaftliche Datenzentren im deutschsprachigen Raum: Grundsatzpapier zur Sicherung der langfristigen Verfügbarkeit von Forschungsdaten*. Hamburg. DOI: 10.5281/zenodo.1134760.
- Erjavec, Tomaž, Jan Jona Javoršek, Matija Ogrin, and Petra Vide Ogrin. 2011. "Od biografskega leksikona do znanstvenokritične izdaje: vprašanje trajnosti elektronskih besedil." *Knjižnica* 55, No. 1: 103–14. <https://knjiznica.zbds-zveza.si/knjiznica/article/view/6004>.
- Diaz, Chris. 2018. "Using Static Site Generators for Scholarly Publications and Open Educational Resources." *Code4Lib Journal*, No. 44. <https://journal.code4lib.org/articles/1386>.
- Fechner, Martin. 2018. "Eine nachhaltige Präsentationsschicht für digitale Editionen." In *DHd 2018: Kritik der digitalen Vernunft: Konferenzabstracts*, edited by Georg Vogeler, 203–7. Universität zu Köln. <http://dhd2018.uni-koeln.de/>.
- Flanders, Julia, Syd Bauman, and Sarah Connell. 2016. "XSLT: Transforming our XML data." In *Doing Digital Humanities: Practice, Training, Research*, edited by C. Crompton, R. J. Lane and R. Siemens, 255–72. Oxon and New York: Routledge.
- Gašparič, Jure. 2014. *Slovenski parlament: Politično zgodovinski pregled od začetka prvega do konca šestega mandata (1992–2014)*. Ljubljana: Inštitut za novejšo zgodovino. <http://hdl.handle.net/11686/26950>.
- Kraetke, Martin, and Gerrit Imsieke. 2016. "XSLT as a Modern, Powerful Static Website Generator: Publishing Hogrefe's Clinical Handbook of Psychotropic Drugs as a Web App." In *Proceedings of XML in, Web Out: International Symposium on sub rosa XML*, Balisage Series on Markup Technologies, vol. 18. <https://doi.org/10.4242/BalisageVol18.Kraetke02>.

- Lumley, John, Debbie Lockett, and Michael Kay. 2017. "Compiling XSLT3, in the Browser, in Itself." In *Proceedings of Balisage: The Markup Conference 2017*, Balisage Series on Markup Technologies, vol. 19. <https://doi.org/10.4242/BalisageVol19.Lumley01>.
- Moeller, Katrin, Matej Đurčo, Barbara Ebert, Marina Lemaire, Lukas Rosenthaler, Patrick Sahle, Ulrike Wuttke, and Jörg Wettlaufer. 2018. Die "Summe geisteswissenschaftlicher Methoden? Fachspezifisches Datenmanagement als Voraussetzung zukunftsorientierten Forschens." In *DHd 2018: Kritik der digitalen Vernunft: Konferenzabstracts*, edited by Georg Vogeler, 89–93. Universität zu Köln. <http://dhd2018.uni-koeln.de/>.
- Neuefeind, Claes, Philip Schildkamp, and Brigitte Mathiak. 2019. "Technologienutzung im Kontext Digitaler Edition – eine Landschaftsvermessung." In *DHd 2019: Digital Humanities: multimedial & multimodal. Konferenzabstracts*, 219–22. Universität Mainz, Universität Frankfurt. <https://dhd2019.org/>.
- Ogrin, Matija, and Tomaž Erjavec. 2009. "Ekdotika in tehnologija: Elektronske znanstvenokritične izdaje slovenskega slovstva." *Jezik in slovstvo* 54, No. 6 (2009): 57–72. <http://www.dlib.si/?URN=URN:NBN:SI:doc-BOC8BANS>.
- Ogrin, Matija, ed. 2005. *Znanstvene razprave in elektronski mediji: razprave*. Ljubljana: Založba ZRC, ZRC SAZU. <http://nl.ijs.si/e-zrc/bib/eziss-knjiga.pdf>.
- Pančur, Andrej. 2016. "History of the Holocaust in Slovenia." In *Between the House of Habsburg and Tito: A Look at the Slovenian Past*, edited by Jurij Perovšek and Bojan Godeša. Ljubljana: Inštitut za novejšo zgodovino. <http://hdl.handle.net/11686/36294>.
- Rinaldi, Brian. 2015. *Static Site Generators: Modern Tools for Static Website Development*. Sebastopol, CA: O'Reilly Media.
- Robinson, Peter. 2016. "Why Interfaces Do Not and Should Not Matter for Scholarly Digital Editions." In *Digital Scholarly Editions as Interfaces, International Symposium at the University of Graz, Austria*, 29–30. Centre for Information Modelling – Austrian Centre for Digital Humanities.
- https://static.uni-graz.at/fileadmin/gewizentren/Informationsmodellierung/PDF/dse-interfaces_BoA21092016.pdf.
- Rosselli Del Turco, Roberto. 2016. "The Battle We Forgot to Fight: Should We Make a Case for Digital Editions?" In *Digital Scholarly Editing: Theories and Practices*, edited by Matthew James Driscoll and Elena Pierazzo, 19–238. Cambridge: Open Book Publishers. <http://dx.doi.org/10.11647/OBP.0095>.
- Romary, Laurent, Piotr Banski, Jack Bowers, Emiliano Degl'innocenti, Matej Đurčo, Roberta Giacomini, Klaus Illmayer, Adeline Joffres, Fahad Khan, Mohamed Khemakhem, et al. 2017. *Report on Standardization (draft)*. [Technical report] 4.2 Inria. <https://hal.inria.fr/hal-01560563>.
- Schaffner, Jennifer and Ricky Erway. 2014. *Does Every Research Library Need a Digital Humanities Center?* Dublin, Ohio: OCLC Research. <https://www.oclc.org/content/dam/research/publications/library/2014/oclcresearch-digital-humanities-center-2014.pdf>.
- TEI Consortium, ed. 2019. *TEI P5: Guidelines for Electronic Text Encoding and Interchange 3.5.0*. TEI Consortium. <http://www.tei-c.org/Guidelines/P5/>.
- Turska, Magdalena, James Cummings, and Sebastian Rahtz. 2016. "Challenging the Myth of Presentation in Digital Editions." *Journal of the Text Encoding Initiative*, No. 9. DOI: 10.4000/jtei.1453.
- Vigiante, Raffaele. 2017. "Your Own Shelley-Godwin Archive: An off-line strategy for an on-line publication (poster)." In *TEI 2017 Victoria*. https://hcmc.uvic.ca/tei2017/abstracts/t_126_vigianteshelleygodwin.html.
- Visconti, Amanda. 2016. "Building a Static Website with Jekyll and GitHub Pages." *The Programming Historian*, 5. <https://programminghistorian.org/lessons/building-static-sites-with-jekyll-github-pages>.
- Williams, Martin. 2019. *Web Development Trends 2019* (blog). March 14, 2019. Accessed April 12, 2019. <https://www.keycdn.com/blog/web-development-trends-2019>.

Andrej Pančur

SUSTAINABILITY OF DIGITAL EDITIONS: STATIC WEBSITES OF THE HISTORY OF SLOVENIA – SISTORY PORTAL

SUMMARY

The contribution is based on the position that, with regard to digital editions, the highest possible degree of digital sustainability of data, presentations, functionalities, and programme code should be ensured. This represents a significant challenge, especially in case of smaller digital humanities projects with limited financing, which does not allow for the long-term maintenance of technically-demanding digital editions. The alternative solutions facilitated by the swift development of static web pages in the recent years are presented in the contribution.

Static websites enjoy numerous advantages in comparison with dynamic websites: efficiency, hosting, security, maintenance, and versioning. These reasons are particularly important to ensure the sustainability of digital editions. These reasons, however, are less convincing in case we expect digital editions to contain user-generated contents as well. Therefore, static websites are not appropriate for all digital editions in the field of digital humanities. On the other hand, countless digital projects do not call for very complex content and its display. In such cases the existing solutions provided by static websites can be more than satisfactory, especially because modern static websites do not completely lack the option of adding dynamic contents. Modern static websites are generated by employing static website generators. Humanities texts are most often encoded with Extensible Markup Language (XML). Extensible Stylesheet Language for Transformation (XSLT) is used as a tool for XML conversion: also in static websites. Digital editions based on the TEI have been successfully included in the Sistory portal repository as static web pages, employing basic XML (XSLT) and web technologies (HTML, CSS, JavaScript). All the static web pages also have the possibility of displaying dynamic content.

In the case of Sistory portal, we have decided to upgrade the basic XSLT Stylesheets of the TEI Consortium. In the *Sistory TEI Profile* chapter, I will present generic upgrade of the TEI Stylesheets. In the chapter *Configuring and Upgrading the Sistory TEI Profile* I will outline the project-specific options for upgrading this profile. In both these chapters, I will also discuss the various options of adding dynamic contents to static websites. In the chapter *Publishing Digital Editions* I will outline how these static websites can be made available to the public, in particular by their inclusion in the Sistory portal's digital repository. In the *Conclusion*, I will also mention a few more general findings.

There are several advantages to a digital editions infrastructure organised in this manner: using format that is most common in digital humanities (TEI XML); using

a single XML technology (XSLT) for various sorts of digital editions, which enjoys a wide support in the TEI community; the possibility of simply including JavaScript libraries and plugins; flexibly adding dynamic contents with Saxon-JS; in comparison with other technologies (dynamic sites), static sites ensure a relative sustainability and simple maintenance of digital editions; using Git version control to store the various versions of digital editions together with the software used to generate static websites; open access to the complete digital editions code in the GitHub and GitLab software development platforms; the possibility of sharing digital editions on the GitHub Pages and GitLab Pages, and, last but not least, the possibility of including them in the History of Slovenia – Sistory portal.

Andrej Pančur

TRAJNOST DIGITALNH IZDAJ: STATIČNE SPLETNE STRANI PORTALA ZGODOVINA SLOVENIJE – SISTORY

POVZETEK

Prispevek izhaja iz stališča, da je pri digitalnih izdajah potrebno poskrbeti za čim bolj celovito digitalno trajnost tako podatkov kot predstavij, funkcionalnosti in programske kode. To je velik izziv predvsem za manjše digitalno humanistične projekte z omejenim financiranjem, ki ne omogoča dolgoročnega vzdrževanja tehnično zahtevnih digitalnih izdaj. Kot alternativno rešitev so v prispevku predstavljene rešitve, ki jih v zadnjih letih ponuja hiter razvoj statičnih spletnih strani.

Statične spletne strani imajo v primerjavi s dinamičnimi številne prednosti: zmogljivost, gostovanje, varnost, vzdrževanje in kontrola verzij. Ti razlogi so zlasti pomembni zaradi trajnosti digitalnih izdaj. Vendar so ti razlogi manj prepričljivi, če glede digitalnih izdaj pričakujemo, da bodo vsebovale tudi uporabniško generirano vsebino. Zato statične spletne strani niso primerne za vse digitalne izdaje s področja digitalne humanistike. Po drugi strani pa je zelo veliko digitalnih projektov, kjer vsebina in njen prikaz nista tako zelo zahtevni. V teh primerih bi bile obstoječe rešitve, ki jih prinašajo statične spletne strani, več kot zadovoljive, predvsem zaradi tega, ker moderne statične strani niso povsem brez možnosti dodajanja dinamičnih vsebin. Moderne statične spletne strani generiramo s pomočjo generatorjev statičnih spletnih strani. Besedila v humanistiki večinoma kodiramo z XML označevalnim jezikom. XSLT pa uporabljamo kot orodje za pretvorbo XML: tudi v statične spletne strani. Digitalne izdaje, ki temeljijo na TEI, so s pomočjo osnovnih XML (XSLT) in spletnih tehnologij (HTML, CSS, JavaScript) kot statične spletne strani uspešno vključene v repozitorij portala Sistory. Vse statične spletne strani imajo tudi možnost dinamičnega prikazovanja vsebine.

V primeru portala SIstory smo se odločili za nadgradnjo osnovnih pretvorb XSLT konzorcija TEI. V poglavju SIstory TEI profil bom predstavil svojo generično nadgradnjo pretvorb XSLT konzorcija TEI. V poglavju Konfiguracija in nadgradnja SIstory profila bom nato predstavil projektne specifične možnosti nadgradnje tega profila. V obeh teh poglavjih bom predstavil še različne možnosti dodajanja dinamične vsebine statičnim spletnim stranem. V poglavju Publiciranje digitalnih izdaj bom omenil, kako te statične spletne strani damo na razpolago javnosti, predvsem z vključitvijo v digitalni repozitorij portala SIstory. V Sklepu naposled dodam še nekaj pomembnejših splošnih ugotovitev.

Tako vzpostavljena infrastruktura za digitalne izdaje ima več prednosti: uporaba podatkov, ki so v digitalni humanistiki najbolj razširjeni (TEI-XML); uporaba enotne XML tehnologije (XSLT) za različne vrste digitalnih izdaj, ki ima široko podoro v TEI skupnosti; možnost enostavnega vključevanja JavaScript knjižnic in vtičnikov; fleksibilno dodajanje dinamične vsebine s Saxon-JS; statične spletne strani zagotavljajo v primerjavi z ostalimi tehnologijami (dinamične spletne strani) relativno trajnost digitalnih izdaj ter relativno enostavno vzdrževanje; uporaba Git kontrole verzij za shranjevanje različnih izdaj digitalnih izdaj, skupaj s programsko opremo, ki smo jo uporabili pri generiranju statičnih spletnih strani; odprti dostop do celotne kode digitalnih izdaj v platformah za razvoj programske opreme GitHub in GitLab; možnost gostovanja digitalnih izdaj v GitHub Pages in GitLab Pages in nenazadnje možnost vključitve v portal Zgodovina Slovenije – SIstory.