

1 Using the Publish Tool SADE

3 General Components

3.1 eXist-db

eXist is an XML database, which allows to store XML documents, to retrieve and further process these by using XML technologies such as XQuery and XSLT. In case of larger amounts of data, the search can be accelerated by applying an index. eXist 2.2 provides a newly designed range index, which allows a faster search in individual XML fields or attributes. Especially the faceted search benefits from this development. eXist not only provides the data base and the support of XML technologies, but also some relevant useful applications, which can be integrated appropriately. These are usually available via an own public repository. The here provided sample data set includes the works of William Shakespeare and exemplifies its use with TEI documents.

3.1.1 Template Engine

The template engine of SADE is a fork of an earlier version of the eXist one. Changes concerning the template engine of newer eXist versions have been partially back ported into the SADE template engine. It makes it possible to write SADE modules that work independently of CSS and website design and with various representations. There is e.g. a bootstrap template for the faceted search. Templates with other CSS frameworks are also possible. The module's functionality remains the same in all representations.

3.1.2 SADE

As a software within the eXist-db, SADE uses this template engine and XAR modules. XAR modules are packages that combine XML resources as XQuery scripts, XSLT style sheets and XML allowing to install them in eXist as bundled apps or as extensions of functionality. The format used in eXist XAR is a modified version of the XPath package one. These packages can be distributed as an individual upload or via package repositories.

Using the eXist template engine ensures the independence of such XAR modules from the layout of the represented website. Modules, templates and paths to the XML data can be defined in a configuration file in order to create own portals with the necessary components and individual designs. That way, data, data processing and visualization are also kept separately within the database.

Components like the view of TEI transformations, the search or the navigation - usually needed in digital editions - belong to the core part of the project and can further be developed. Own developments, such as a timeline or a map-based visualization, can be offered to the community as XAR modules.

At present, there are two projects within TextGrid that use earlier versions of SADE in a modified form: "Blumenbach-online", the "digital edition of the notebooks of Theodor Fontane" and the "Bibliothek der Neologie".

3.2 Digilib

Digilib is a server-based software for image editing. It allows, among other things, to retrieve images in different scales and formats or even individual image sections from the server. Digilib offers a service that carries out operations on the pictures, provides a REST-API as well as a web-based client. The latter one allows user interaction with images offered by the service and image processing functions.

Digilib has been available in TextGrid since the second funding phase. Under one of the Mellon Foundation funded project to integrate IIIF in TextGrid, the integration had significantly been improved in 2013. This also led to an optimization of speed. In future, the integrated Digilib service can be used instead of a SADE-integrated version for images hosted on TextGrid. One advantage of this solution is that digital reproductions can be directly stored in TextGrid in the format in which they leave the scanning process - the TIFF format. Furthermore, it is no longer necessary to provide the images as JPEGs as it used to be when publishing with the SADE-integrated version.

Images do not need to be copied to the SADE installation, when using the TextGridLab for publication. When documents that contain images are output, objects are retrieved as JPEG in a suitable resolution (100px for thumbnails, 1500px for the view of scanned pages) based on the TextGrid URI of Digilib. Thus, in the future, speed optimisation can be carried out centrally via the TextGrid service. Individual SADE servers are relieved of the image conversion. SADE simplifies the use of the DARIAH-eXist-Hosting because of its independence of the integrated Digilib in so far as only one eXist database has to be available for a working installation.

3.3 TextGrid specific components

The components listed below function as modules within the eXist database as well as they do without the main component SADE. The extensions of SADE will be described in a subsequent section. All source codes are freely available at <http://github.com/ubbo/>.

3.3.1 tg-client

Most components of TextGrid can be controlled via REST/SOAP interface or web interface. The XQuery script provides all these functions within the database. Thus, authentication and other services are accessible via all other modules or if necessary also via the web interface. It currently offers connecting features to:

- the SPARQL-interface of TG-search
- the metadata objects provided by TG-crud
- the data objects provided by TG-crud
- the authentication service TG-auth (including a cached session ID to avoid unnecessary load on TG-auth)

Another feature allows to remove the URI prefix "textgrid" from the object references. In this way, the core services of TextGrid are provided within the database and can be integrated into other modules.

3.3.2 tg-connect

This script is used for document transmissions. It is responsible for retrieving and storing documents and uses the functions provided by the tg-client. It collects the information transmitted by the Publish tool SADE (TextGridLab Plugin) from the TextGrid Laboratory and processes them. Hence, the documents are transmitted from the TextGrid Repository to the SADE instance. Finally, the script is retrieved in order to set up the menu-navigation.

3.3.3 tg-menu

The hierarchical structure, which is represented in the TextGrid Laboratory within the Navigator, is set up with the module TG-menu. This structure will then be incorporated as a menu into the digital presentation interface. Here, it is still paid attention to keep the data separated from the layout. Based on this structure, an XML document is mainly generated and stored in the collection associated with the project. Finally, another processing step is necessary to be compliant with the respective template - the layout of the digital edition: An XSLT, which has to be stored in the project collection under the name of the selected template, is automatically retrieved. This allows using flexible layouts and designs.

3.3.4 digilib-Proxy

In the original version of the TextGrid SADE-publish-component published images were stored in the local directory of the Digilib SADE-installation. Therefore, authentication within TextGrid was only necessary when images were published. In the future, the TextGrid Digilib-service can be used for image representation. So whenever images are going to be displayed, an authentication will be necessary. This applies in the case where a rights management for the image query is needed, e.g. for image-documents especially released for the edition in question. Therefore a Digilib proxy was implemented in XQuery, a script that is able to use a valid TextGrid-account and that coordinates these queries. Requests from the portal to the Digilib are submitted to the proxy. This proxy uses a deposited TextGrid-user-account to log in on the TextGrid Repository, obtains a SessionID and the authorization to see the requested image. The authorization is passed on to the end user, while the corresponding session ID remains hidden.

Excursus: Functional Accounts

So-called functional accounts are already used by the two mentioned projects; their data must be specified in the configuration of SADE. These are TextGrid-accounts, which assume only an observer status within the projects in question. Currently each TextGrid-user-account is able to create TextGrid-projects and to publish data. As for all TextGrid-accounts a real user is stored: Liability and responsibility are clarified. But all users, who have expanded access rights within the portal, could potentially know a user account stored in the project-portal of SADE. An even more limited account type is therefore necessary. These are user accounts that explicitly only have read-access to projects they are associated with.

3.3.5 SADE Module

Three new modules for SADE were programmed: one for the navigation, a viewer and a faceted search. All three modules use the template engine.

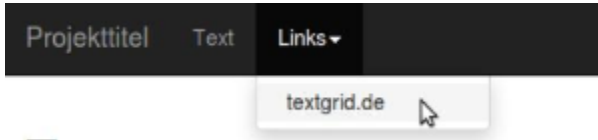
3.3.5.1 Navigation

The navigation module was developed in order to make the contents of the SADE-portal available within its own navigation menu. This module allows defining the menu structure of the portal in an XML file. Individual links are provided with a name and a reference; relative links remain within the portal. External links are also possible. A user manual has been edited for this module. The portal-navigation can be described in a simple XML structure; e.g. a XML configuration and the resulting layout of the menu (in the bootstrap template):

```

<navigation>
<item label="Text" link="index.html?id=text.md"/>
<submenu label="Links">
<item label="textgrid.de" link="http://textgrid.de "/>
</submenu>
</navigation>

```



Currently, no submenus can be nested at this point.

3.3.5.2 Viewer

The programmed viewer module checks the file extension and distinguishes between HTML-, Markdown- and XML-documents. Instructions are provided for these document types, each of those leading to an HTML output. XML documents are examined in order to find out if they are created with the Text-Image-Link Editor integrated in TextGrid or if a different structure is given.

For the latter an appropriate XSLT style sheet according to the namespace is selected. This allows to display Markdown, TEI and XML documents with the same viewer. The viewer supports a page by page output of TEI documents, ignoring elements previously included in the document. Empty elements, which are potentially and semantically overlapping, are neglected. The Markdown representation was integrated in order to fast and easily create individual texts for the portal, which currently are not searchable. Support and tutorial documentations of the TextGrid-SADE – reference-installation are written in Markdown. TEI documents will be integrated into the website by using style sheets (provided by the TEI-C, <http://www.tei-c.org/Tools/Stylesheets/>). For layout purposes, individual style sheets instead of the built-in ones can be specified in the multiviewer section within the project-configuration-file (config.xml). One style sheet can be specified for each namespace of the document's root node.

```

<module key="multiviewer">
<param key="xslt">
<!-- Nutze das eigene Stylesheet aus dem xslt Ordner im
Projektverzeichnis (local=true) für den TEI namespace -->
<stylesheet
namespace="http://www.tei-c.org/ns/1.0"
local="true"
location="xslt/my-tei-stylesheet.xsl" /><!-- Eigenes stylesheet für den XML dokumente mit eigenem Namespace,
Übergabe eines Parameters an das XSLT -->
<stylesheet
namespace="http://my.project.org/specific-namespace/1.1"
local="true"
location="xslt/my-project.xslt.xsl">
<parameters>
<param name="imgParams" value="?dh=200"/>
</parameters>
</stylesheet>
</param>
</module>

```


SemToNotes-Integration

Software components from the project "SemToNotes" (<https://github.com/HKI/Koeln/SemToNotes> and <http://hkikoeln.github.io/SemToNotes/>) have been integrated in the SADE environment for the representation of synoptic views. SemToNotes is an annotation and visualization tool, which is suitable for the description and presentation of topological structures in image documents and their content related links.

FOOBAR DOKUMENTATION WEITERFÜHRENDE LINKS INFO search

BILD METADATEN TRANSKRIPTION LESEFASSUNG TXT-IMG-LINKS

Universitaet Goettingen, Cod. Ms. philos. 63



Metadaten

Conrad Kyesser: Bellifortis [BLATT 007r: Jupiter]

Digitale Edition des Manuskripts Bellifortis 24. Mai 2014
 Editor: Max Teich
 Herausgeber: Elmar Mittler, Oliver Schmid

Quellenbeschreibung:

- Ort: Goettingen
- Insitution: Universitaet Goettingen

Transkription/Übersetzung

Arcitenens spinas tenet ancas minutaque membra Atque remota locis et quod dat vesice

In pede piscis habet nervum nervique dolorem

Sum vexillum Jovis viridi depictus colore
 Dux ego stellarum dominus sum princeps earum
 In oriente pridem mihi domus in septentrione
 Sagittam cum piscibus altus mihi dedit in esse

Lesefassung

Abbildung

Beschreibung: Der Schütze
 Der Schütze hat dornige Kniee und verkürzte Glieder - und aus ihrer Lage versetzte, und was der Blase Schaden tut.

3.3.5.3 Faceted Search

A SADE module has been written for the search within published XML data, which performs the search and creates facets according to its configuration. These can be specified with XPath expressions within the configuration. On the web page, facets can be excluded from the search results by clicking on it. The result can be limited to one facet by clicking on it.

An XSLT is used in order to display a single hit. It extracts metadata from the found document and represents it within the result field. This allows to output own metadata fields from the TEI document. It can be configured in the config.xml of the project, within the module with the key="faceted-search". Excerpt from config.xml


```
<module key="faceted-search">
[... ]
<!-- Benutze ein eigenes XSLT für die Trefferanzeige -->
<param key="result-xslt">/db/sade-projects/my-project/xslt/search-
hit.xslt</param>
<param key="facets">
<!-- Vorgegebene Facetten, können geändert oder gelöscht werden -->
<facet key="persons" title="Person">
<xpath>tei:author</xpath>
</facet>
<facet key="keywords" title="Schlagwort">
<xpath>tei:term</xpath>
</facet>
<facet key="dates" title="Zeit">
<xpath>tei:date/@when</xpath>
</facet>
<facet key="places" title="Publikationsort">
<xpath>tei:pubPlace</xpath>
</facet>
<!-- Eigene Facette -->
<facet key="personstext" title="Person im Text">
<xpath>tei:persName</xpath>
<xpath>tei:rs</xpath>
</facet>
</param>
</module>
```