SADE Dokumentation und Referenzinstanz
(SADE Reference documentation)

Version 30.05.2015

Arbeitspaket 4.4

Verantwortlicher Partner: SUB Göttingen

TextGrid
Virtuelle Forschungsumgebung in den Geisteswissenschaften
<table>
<thead>
<tr>
<th>Projekt:</th>
<th>TextGrid – Institutionalisierung einer Virtuellen Forschungsumgebung in den Geisteswissenschaften</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMBF Förderkennzeichen:</td>
<td>01UG1203A</td>
</tr>
<tr>
<td>Laufzeit:</td>
<td>Juni 2012 bis Mai 2015</td>
</tr>
<tr>
<td>Dokumentstatus:</td>
<td>final</td>
</tr>
<tr>
<td>Verfügbarkeit:</td>
<td>öffentlich</td>
</tr>
<tr>
<td>Autoren:</td>
<td>Göbel, Mathias; SUB Leone, Claudio; SUB Veentjer, Ubbo; SUB</td>
</tr>
</tbody>
</table>

Table of Contents

Entwicklungshistorie .............................................................................................................. 5
SADE und TextGrid .............................................................................................................. 5
Umgebung (Technical Requirements) ...................................................................................... 5
Allgemeine Komponenten ....................................................................................................... 6
eXist-db .................................................................................................................................. 6
  Template Engine .................................................................................................................. 6
  SADE ................................................................................................................................... 6
Digilib .................................................................................................................................... 7
TextGrid-specifische Komponenten ....................................................................................... 7
tg-client ................................................................................................................................. 7
tg-connect .............................................................................................................................. 8
tg-menu .................................................................................................................................. 8
digilib-Proxy ........................................................................................................................ 8
SADE Module ........................................................................................................................ 9
  Navigation ............................................................................................................................ 9
  Viewer .................................................................................................................................. 10
     Auszug aus config.xml .................................................................................................... 10
     SemToNotes-Integration ................................................................................................. 11
Faceted Search ....................................................................................................................... 11
     Auszug aus config.xml .................................................................................................... 13
Installation ............................................................................................................................ 13
Server .................................................................................................................................... 13
  Download ............................................................................................................................. 13
  Starten ................................................................................................................................. 14
TextGrid Laboratory Plugin (Client) ...................................................................................... 14
Konfiguration des Plugins .................................................................................................... 15
TextGrid SADE Publish ........................................................................................................ 15
Publizieren von Objekten und Aggregationen aus dem TextGrid Lab ........................................ 15
Konfiguration der SADE-Instanz ........................................................................................ 17
  Config.xml .......................................................................................................................... 17
  Menüführung ....................................................................................................................... 17
     Menü oben ....................................................................................................................... 18
     Datenstruktur (Sidebar) ................................................................................................... 18
Zukünftige Anbindungs möglichkeit ..................................................................................... 18
DARIAH eXist Hosting ........................................................................................................... 18
Die Testinstanz ....................................................................................................................... 18
Development ......................................................................................................................... 19
SADE and TextGrid ................................................................................................................ 19
Environment (Technical Requirements) ................................................................................ 19
General Components .......................................................................................................... 19
eXist-db .................................................................................................................................. 19
  Template Engine ................................................................................................................ 20
  SADE .................................................................................................................................. 20
Digilib .................................................................................................................................... 20
TextGrid specific components .............................................................................................. 21
tg-client ................................................................................................................................. 21
tg-connect ............................................................................................................................. 21
tg-menu .................................................................................................................................. 22
digilib-Proxy ........................................................................................................................ 22

Excursus: Functional Accounts ......................................................................................... 22
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SADE Module</td>
<td>22</td>
</tr>
<tr>
<td>Navigation</td>
<td>22</td>
</tr>
<tr>
<td>Viewer</td>
<td>23</td>
</tr>
<tr>
<td>SemToNotes-Integration</td>
<td>24</td>
</tr>
<tr>
<td>Faceted Search</td>
<td>25</td>
</tr>
<tr>
<td>Excerpt from config.xml</td>
<td>25</td>
</tr>
<tr>
<td>Installation</td>
<td>26</td>
</tr>
<tr>
<td>Server</td>
<td>26</td>
</tr>
<tr>
<td>Download</td>
<td>26</td>
</tr>
<tr>
<td>Start</td>
<td>26</td>
</tr>
<tr>
<td>TextGrid Laboratory Plugin (Client)</td>
<td>27</td>
</tr>
<tr>
<td>Configuration of the plug-in</td>
<td>27</td>
</tr>
<tr>
<td>TextGrid SADE Publish</td>
<td>27</td>
</tr>
<tr>
<td>Publishing objects and aggregations from the TextGrid Lab.</td>
<td>28</td>
</tr>
<tr>
<td>Configuration of the SADE instance</td>
<td>29</td>
</tr>
<tr>
<td>Config.xml</td>
<td>30</td>
</tr>
<tr>
<td>menu navigation</td>
<td>30</td>
</tr>
<tr>
<td>menu at the top</td>
<td>30</td>
</tr>
<tr>
<td>Data structure (Sidebar)</td>
<td>30</td>
</tr>
<tr>
<td>Further connecting features</td>
<td>30</td>
</tr>
<tr>
<td>DARIAH eXist Hosting</td>
<td>30</td>
</tr>
<tr>
<td>The Reference Instance</td>
<td>31</td>
</tr>
</tbody>
</table>
Entwicklungsgeschichte

SADE steht als Akronym für "Skalierbare Architektur für digitale Editionen" und ist ein Programmpaket, welches viele Komponenten umfasst, die zum Entwickeln und Darstellen einer digitalen Edition besonders geeignet sind. Der Fokus liegt dabei auf der Präsentation von XML-Dokumenten, die den Richtlinien der Text Encoding Initiative (TEI) folgen. Ursprünglich an der Berlin-Brandenburgische Akademie der Wissenschaften (BBAW) im Rahmen der Digitalisierungsinitiative "The Electronic Life Of The Academy" (TELOTA) entwickelt, wird diese Software seit 2012 von einer größeren Entwicklergemeinde weiterentwickelt. An dieser beteiligen sich unter anderem die BBAW, die Österreichische Akademie der Wissenschaften (ÖAW; im Rahmen des Projektes CLARIN: Common Language Resources and Technology Infrastructure), das Max-Planck-Institut für Wissenschaftsgeschichte (MPI-WG), das Cologne Center for eHumanities (CCeH) und die Niedersächsische Staats- und Universitätsbibliothek Göttingen (SUB; im Rahmen von TextGrid).

Ferner bezeichnet sich eine Programmkomponente innerhalb des Paketes als SADE. Diese Komponente steuert den gesamten Output. Es handelt sich dabei um eine Anwendung, die in XQuery geschrieben ist und in der XML-Datenbank aktiv ist. Demnach ist die XML-Datenbank "eXist-db" (http://exist-db.org/) auch wichtigster Bestanteil des Programmpaketes.

SADE und TextGrid


Umgebung (Technical Requirements)

Das gesamte Programmpaket funktioniert als Webapplikation, die ihren eigene Servlet-Container (Jetty) mitbringt. Damit reduzieren sich die technischen Anforderungen auf eine Java-Laufzeitumgebung (JRE).
Allgemeine Komponenten

eXist-db


Template Engine


SADE


eine Zeitleiste oder eine Landkarten-basierte Visualisierung, können als XAR-Module der Community zur Verfügung gestellt werden. Im TextGrid-Kontext gibt es derzeit zwei Projekte, die frühere Versionen von SADE in modifizierter Form einsetzen: Blumenbach-Online, die digitale Edition der Notizbücher Theodor Fontanes und die “Bibliothek der Neologie”.

**Digilib**


**TextGrid-spezifische Komponenten**


**tg-client**

Die meisten Komponenten von TextGrid lassen sich per REST/SOAP-Schnittstelle oder Webinterface ansteuern. Das XQuery-Skript stellt all diese Funktionen in der Datenbank zur Verfügung. So kann man aus allen anderen Modulen oder ggf. auch von der Weboberfläche aus auf Authentifizierungsdienste und mehr zugreifen. Es bietet derzeit Anknüpfungen an:
• die SPARQL-Schnittstelle von tg-search
• die von tg-crud bereitgestellten Metadaten-Objekte
• die von tg-crud bereitgestellten Daten-Objekte
• den Authentifizierungsdienst tg-auth (inkl. einer zwischengespeicherten Session-ID, um unnötige Last bei tg-auth zu vermeiden)

und es bietet zudem eine Funktion, um die URI-Prefixes “textgrid:” aus den Objektreferenzen zu entfernen. Damit stehen die Kernservices von TextGrid innerhalb der Datenbank zur Verfügung und können in andere Module eingebunden werden.

**tg-connect**
Dieses Skript wird bei der Dokumentenübertragung genutzt. Es ist für das holen und ablegen der Dokumente verantwortlich und bedient sich intensiv der von tg-client bereitgestellten Funktionen. Es greift dabei die aus dem TextGrid Laboratory vom [Publish-Tool SADE (TextGridLab Plugin)](link) übertragenen Informationen ab und verarbeitet diese, überträgt also die Dokumente aus dem TextGrid Repository in diese SADE-Instanz. Abschließend wird das Skript zum Aufbau der Menüführung abgerufen.

**tg-menu**

**digilib-Proxy**
Nutzeraccount am TextGrid Repository an, bezieht eine SessionID und damit die Berechtigung, das abzurufende Bild sehen zu dürfen. Diese Berechtigung wird an den Endnutzer weitergegeben, die zugehörige Session-ID bleibt aber versteckt.

Exkurs: Funktionsaccounts

SADE Module
Es wurden drei neue Module für SADE programmiert: eines für die Navigation, ein Viewer und eine facettierte Suche. Alle drei Module nutzen die Template Engine.

Navigation

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

Illustration 1: Mit dem Navigationsmodul erstelltes Menü
Derzeit können an dieser Stelle keine Untermenüs verschachtelt werden.
Viewer


In der Projekt-Konfigurationsdatei können eigene Stylesheets zur Darstellung anstelle der integrierten Stylesheets angegeben werden. Dafür ist in der config.xml die Sektion Multiviewer zuständig. Hier kann pro namespace des Dokument root Knotens ein Stylesheet angegeben werden.

Auszug aus config.xml

```xml
<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"/>
  </param>

  <!-- 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>
</module>
```
SemToNotes-Integration

Faceted Search
Für die Suche in publizierten XML-Daten wurde ein SADE-Modul geschrieben, welches die Suche durchführt und entsprechend seiner Konfiguration Facetten bildet. In der Konfiguration lassen sich für die einzelnen zu bildenden Facetten XPath-Ausdrücke angeben. Facetten lassen sich in der Webseite per Klick auf aus dem Such-Ergebnis ausschließen. Mit Klick auf die Facette lässt sich das Ergebnis auf diese Facette einschränken.

Illustration 2: Ergebnis der mit Hilfe des Text-Bild-Link-Editors erstellten Daten
Für die Darstellung eines einzelnen Ergebnistreffers wird ein XSLT verwendet, das Metadaten aus dem gefundenen Dokument extrahiert und beim Suchtreffer darstellt. Das bietet eine flexible Möglichkeit, eigene Metadatenfelder aus dem TEI Dokument bei der Suchtrefferanzeige auszugeben. Konfiguriert wird dieses in der config.xml des Projektes, im module mit key="faceted-search".
Auszug aus config.xml

```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>
<!-- 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>
```

Installation

Server

Download
Dieser Report bezieht sich auf die im TextGrid Wiki bereitgestellte SADE-Version latest-stable.tar.gz (SADE-Server)
Starten


<table>
<thead>
<tr>
<th>Dateiname</th>
<th>Funktion</th>
</tr>
</thead>
<tbody>
<tr>
<td>backup</td>
<td>erstellt ein Backup der Datenbank</td>
</tr>
<tr>
<td>client</td>
<td>Start Script for eXist interactive client</td>
</tr>
<tr>
<td>run</td>
<td>mit Startoptionen für Linked Java Object Builder und YourKit Java Profiler mit Launcher und Tray-Icon</td>
</tr>
<tr>
<td>server</td>
<td>Start Script for Jetty + eXist</td>
</tr>
<tr>
<td>shutdown</td>
<td>Stop Jetty + eXist</td>
</tr>
<tr>
<td>startup</td>
<td>Start Script for Jetty + eXist mit Option &quot;Java debugging via JDWP on port 4000 in Server mode&quot; (auskommentiert) ohne Launcher und Tray-Icon</td>
</tr>
</tbody>
</table>


TextGrid Laboratory Plugin (Client)

Konfiguration des Plugins

Um das Plugin nutzen zu können, muss ein Endpunkt in Form einer URL angegeben werden, die zu einem gestarteten SADE-Server führt. Im Fall einer lokalen Installation auf dem eigenen PC lautet diese URL: http://localhost:8080/exist/apps/textgrid-connect/publish/textgrid/
Als „Authorized User“ gibt man „admin“ an und das Passwortfeld kann in der Standardkonfiguration leer bleiben.

TextGrid SADE Publish


Publizieren von Objekten und Aggregationen aus dem TextGrid Lab

Sind Server und Plugin installiert, kann mit der Publikation von Dokumenten, Bilder und sonstigen Daten begonnen werden. Man startet das im Laboratory konfigurierte Plugin über das Menü „Werkzeuge“ und den dort gelisteten Punkt „SADE Publish“.
Illustration 3: Starten des SADE-Plugins

Dazu benutzt man im TextGridLab den Navigator und wählt die zu übertragenden Daten aus.

Illustration 4: SADE Publish-Tool und Kontextmenü des Navigators

Auch Aggregationen (Kollektionen, Editionen und Aggregationen selbst) können ausgewählt werden und deren kompletter Inhalt wird kopiert, das Skript arbeitet also rekursiv.

**Konfiguration der SADE-Instanz**


**Config.xml**

Derzeit werden die folgenden Konfigurationen benötigt:

```xml
<param key="textgrid.user">[OBSERVER-ACCOUNT-NAME]</param>
<param key="textgrid.password">[OBSERVER-ACCOUNT-PASSWORD]</param>
```

**Menüführung**

Generell handelt es sich bei dem integrierten Template um eine beispielhafte Umsetzung eines Bootstrap-Templates. An dieser Stelle kann ein beliebiges Template eingefügt werden, was lediglich kleinerer Anpassungsarbeiten bedarf.
Menü oben
Das Menü in der Navigationsleiste oben wird über das bereits beschriebene Navigations-Modul in SADE erstellt.

Datenstruktur (Sidebar)

Zukünftige Anbindungsmöglichkeit

DARIAH eXist Hosting
Im Rahmen des Projektes DARIAH-DE werden, derzeit noch im Testbetrieb, die Einrichtung und das Hosting von eXist-Datenbanken angeboten. Diese können von Geisteswissenschaftlern genutzt werden, die dadurch den Vorteil haben, sich nicht selbst um die Hardware, das Betriebssystem, den Internetzugang usw. kümmern zu müssen. Um ein Hosting von SADE bei DARIAH zu vereinfachen, wurde im Rahmen dieses Projektes darauf geachtet, dass die hier programmierte Software innerhalb einer eXist-Installation als XAR-Paket installiert (und aktualisiert) werden kann.

Die Testinstanz
Development

SADE, an acronym for “Scalable Architecture for digital editions”, is a software package that includes many components, which are particularly suitable for developing and displaying a digital edition. It focuses on the presentation of XML documents that follow the guidelines of the Text Encoding Initiative (TEI). Since 2012 a larger developer community has constantly been working on this software, which was originally developed at the Berlin-Brandenburg Academy of Sciences and Humanities (BBAW) as part of the digitization initiative "The Electronic Life Of The Academy" (TELOTA). Contributors are among others the BBAW, the Austrian Academy of Sciences (OeAW; as part of the project CLARIN: Common Language Resources and Technology Infrastructure), the Max Planck Institute for the History of Science (MPI-WG), the Cologne Center for eHumanities (CCEH) and the Goettingen State and University Library (SUB; as part of the project TextGrid). Furthermore, a software component within the package is also referred to as SADE. This component controls the overall output. It is an application that is written in XQuery and is active in the XML database. Therefore the XML database “eXist-db” is the most essential part of the software package.

SADE and TextGrid

Based on SADE a publication platform will be created, which allows projects working with TextGrid to design their own websites and to fill them with the acquired data. In addition, the as generic as possible working system is supposed to provide a search function and a navigation menu. Via the Market Place, user can integrate a plugin into their labinstallation in order to integrate SADE into the TextGrid Laboratory. This plugin provides a connection to a SADE instance that serves as a publication platform, which can be run on a personal computer for testing purposes. It can also be installed on a server accessible on the Internet. Both parts of the software, the plugin as well as the server component, are part of the described and documented development.

Environment (Technical Requirements)

The entire software package works as a web application that has its own servlet container (Jetty). The technical requirements are thereby reduced to a Java Runtime Environment (JRE).

General Components

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 database 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.

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.

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 eXPath 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”.

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 advantages
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.

**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/](http://github.com/ubbo/).

**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.

**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.
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.

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.

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.

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):

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

![Illustration 5](image_url)

*Illustration 5: menu created with the navigation module. Currently, no submenus can be nested at this point.*

**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/](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.

```xml
<module key="multiviewer">
  <param key="xslt">
    <!-- Nutze das eigene Stylesheet aus dem xslt Ordner im Projekterverzeichis (local=true) für den TEI namespace -->
  </param>
  <stylesheet
    namespace="http://www.tei-c.org/ns/1.0"
    local="true"
    location="xslt/my-tei-stylesheet.xsl"/>
```
SemToNotes-Integration

Software components from the project "SemToNotes" ([https://github.com/HKIKoeln/SemToNotes](https://github.com/HKIKoeln/SemToNotes) and [http://hkikoeln.github.io/SemToNotes/](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. ...

Illustration 6: Published data created with the text image link editor
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 . 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

```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>
```
**Installation**

**Server**

**Download**

This report refers to the SADE version `Latest-stable.tar.gz` (SADE server) provided in the TextGrid Wiki. This software is still in development. The daily build can be retrieved at [http://dev2.dariah.eu/jenkins/view/TextGrid/job/SADE/lastSuccessfulBuild/artifact/dist-utils/antbuild/build/Sade textgrid.tar.gz](http://dev2.dariah.eu/jenkins/view/TextGrid/job/SADE/lastSuccessfulBuild/artifact/dist-utils/antbuild/build/Sade textgrid.tar.gz). Windows user may need an application such as WinRAR or 7-Zip to unpack the compressed folder.

**Start**

The compressed archive needs to be unzipped after its download. Afterwards the “bin” folder can be found in the unpacked directory. Startup applications for the different operating systems are available. Windows user open the files with the extension ".bat", Linux and Mac user the ones with ".sh".

<table>
<thead>
<tr>
<th>filename</th>
<th>funktion</th>
</tr>
</thead>
<tbody>
<tr>
<td>backup</td>
<td>creates a backup of the database</td>
</tr>
<tr>
<td>client</td>
<td>startup script for eXist interactive client</td>
</tr>
<tr>
<td>run</td>
<td>with startup options for Linked Java Object Builder and YourKit Java Profiler with Launcher and tray icon</td>
</tr>
<tr>
<td>server</td>
<td>startup script for Jetty + eXist</td>
</tr>
<tr>
<td>shutdown</td>
<td>stop Jetty + eXist</td>
</tr>
<tr>
<td>startup</td>
<td>startup script for Jetty + eXist with the option &quot;Java debugging via JDWP on port 4000 in server mode&quot; without Launcher and tray icon</td>
</tr>
</tbody>
</table>

In case of installations with a graphical user interface (Windows or Linux on personal computers) "run.sh" (Linux / Mac) or "run.bat" (Windows) is recommended because it contains a launcher, runs in the background and can be accessed or terminated via a tray icon.

To configure a remote server, the documentation of eXist provides an auto-start-guide, accessible via [http://exist-db.org/exist/apps/doc/advanced-installation.xml](http://exist-db.org/exist/apps/doc/advanced-installation.xml). By default, the communication to the instance runs via the port 8080. The "conf.xml" file has to be edited with the appropriate entry.
in order to use the port 80, which is the established standard for http and for websites. Mainly a privileged user can open port 80 for incoming requests. That's why the application has to be started as an administrator or super user. The configurations described below have to be adapted accordingly.

**TextGrid Laboratory Plugin (Client)**

The installation of a plugin is described in detail in the TextGrid manual and is available at [https://dev2.dariah.eu/wiki/display/TextGrid/Installing+Additional+Tools](https://dev2.dariah.eu/wiki/display/TextGrid/Installing+Additional+Tools). In this case you have to select and install the plugin "SADE publish tool". Afterwards a new icon is displayed in the toolbar and the option "Publish to SADE" appears at the context menu of the objects shown in the navigator. User can configure the plug-in in the settings.

**Configuration of the plug-in**

To use the plugin, an end point in the form of a URL must be specified, which leads to an already started SADE server. In case of local installation on your PC, the URL is: [http://localhost:8080/exist/apps/textgrid-connect/publish/textgrid/](http://localhost:8080/exist/apps/textgrid-connect/publish/textgrid/)

As an "Authorized User" you enter "admin" and the password field can be left blank in the default configuration.

![Configuration of the SADE publish tool](https://dev2.dariah.eu/wiki/display/TextGrid/Installing+Additional+Tools)

**TextGrid SADE Publish**

In order to better integrate SADE, TextGrid provides an own publication module based on SADE. It is programmed in JavaScript and displayed in the integrated web browser of the TextGridLab. It transfers the objects selected in the TextGridLab-navigator to the SADE installation. This module, originally implemented in Java, has been rewritten in XQuery. One advantage is, that the module can be directly deployed on the server in any eXist installation as XAR package, even without admin rights. Further advantages are an easier use of the DARIAH-eXist-Hosting and an easy
access to the user authentication of eXist. As part of the new implementation, the user interface of
the module has been redesigned with Bootstrap.

**Publishing objects and aggregations from the TextGrid Lab**

The publication of documents, images and other data can be started after having installed the
server and the plugin. To start the plugin configured in the Laboratory go to the tool menu and
select “SADE-Publish”.

![Illustration 6: Starting the SADE-plugin](image)

Go to the navigator in the TextGridLab and select the data that have to be transmitted.
You can select aggregations (collections, editions and aggregations themselves) and their complete content will be copied. The script works recursively. In the context menu of the objects (accessible via the right mouse button) you select "Publish to SADE", Linux users can also use drag and drop. Afterwards you go to the publish-tool and click the button "Publish" in order to start the transfer process. The XML documents (e.g. TEI documents as well as objects that were created with the text-image-link-editor) and the associated metadata will be transmitted. If you select images that are supposed to be displayed in the SADE portal, only the metadata are transferred and a link to each image will be created within the resulting menu structure.

The publication process is successfully completed, when a green checkmark is added to the listed objects. In case of bigger data records, the process can take some minutes (even hours for several thousand objects). When the process is completed, you can find the transferred objects at local installations via http://localhost:8080/exist/apps/SADE/textgrid/.

**Configuration of the SADE instance**

In order to be able to make changes to the documents and to the configuration within the SADE-instance, you can use the tool “eXide” provided by eXist-db. It is available at http://localhost:8080/exist/apps/Exide/index.html. An editor is provided. You can click the button "OPEN" to search the database in the dialog window and then open the documents in question. As the configured URL (http://localhost:8080/exist/apps/textgrid-connect/publish/textgrid/, see 5.2.1), especially the last part indicates, the data are stored in a "textgrid" project. (Here, the URL
can also be changed and a separate folder can be used. Before that, the corresponding data structure within the XML database has to be created.) The documents are now stored within the main directory "sade-projects" as a collection referred to as "textgrid". There you will find further files that are responsible for the layout and generate further content. The necessary configurations are specified in the XML document "config.xml".

**Config.xml**

Currently, the following configurations are required:

```xml
<param key="textgrid.user">[OBSERVER-ACCOUNT-NAME]</param>
<param key="textgrid.password">[OBSERVER-ACCOUNT-PASSWORD]</param>
```

**menu navigation**

In general the integrated template exemplifies an implementation of the bootstrap templates. Here, any template can be inserted, which only requires minor adjustments.

**menu at the top**

The menu in the top navigation bar is created with the previously described navigation module in SADE.

**Data structure (Sidebar)**

The data model of TextGrid and the structure of the navigator from the TextGridLab are completely copied. The titles of all transmitted documents are displayed in a sidebar and their content made accessible via respective links. Two functions used at the end of the publishing process are crucial. The first one is responsible for the further separation of data and layout and initially creates an XML file, similar to that of the navigation module. Here, the hierarchical structure is mapped and the necessary metadata summarized. This includes the elements object type, title and the TextGrid URI (with an adjusted prefix) including a revision number. Finally, this structure can be made compliant with the template in use by using a simple XSLT. It is stored in a new document, which is integrated into the template.

**Further connecting features**

**DARIAH eXist Hosting**

Within the project DARIAH-DE, the configuration and hosting of eXist databases are offered (currently in a test mode). They can be used by humanities scholars with some advantages: they do not have to worry about hardware, operating system and internet access, etc. In the context of this project the aim was to facilitate the hosting of SADE via DARIAH. Therefore, the programmed software can be installed and updated as XAR package within a eXist installation.
The Reference Instance

Via http://sade.textgrid.de/ a test instance is available. User can start there without a local installation and try out the documents of interest. We do not guarantee for the safety and availability of the documents stored in the reference instance and all data is erased occasionally. To use this service register via http://sade.textgrid.de/exist/apps/SADE/textgrid/sign-up.html. We send an email to the provided address with all data necessary to configure the Lab plug-in.