

CMIF Creator - digitale Briefverzeichnisse leicht erstellt

Müller-Laackman, Jonas

jonas.mueller-laackman@bbaw.de
Berlin-Brandenburgische Akademie der Wissenschaften,
Deutschland

Dumont, Stefan

dumont@bbaw.de
Berlin-Brandenburgische Akademie der Wissenschaften,
Deutschland

Grabsch, Sascha

grabsch@bbaw.de
Berlin-Brandenburgische Akademie der Wissenschaften,
Deutschland

Der Webservice *correspSearch*¹ wird an der Berlin-Brandenburgischen Akademie der Wissenschaften entwickelt, um digitale und gedruckte Briefeditionen editionsübergreifend zu vernetzen und durchsuchbar zu machen. Darüber hinaus sollen die Briefmetadaten mit *correspSearch* für neue digitale Forschungsmethoden (wie z.B. aus der Historischen Netzwerkforschung) bereitgestellt werden. Mittlerweile sind über 44.000 edierte Briefe in *correspSearch* nachgewiesen.

Ein wesentlicher Bestandteil von *correspSearch* ist die Möglichkeit, den Datenbestand mit Metadaten zu eigenen - digitalen oder gedruckten - Briefeditionen zu erweitern. Durch die damit einhergehende Standardisierung der Metadaten auf der Grundlage der Richtlinien der Text Encoding Initiative (TEI) werden Editionen für die digitale Vernetzung erschlossen. Je mehr Briefmetadaten bereitgestellt und aggregiert werden, desto größer ist der Nutzen dieser Daten für die Recherche und Forschung. Mit dem CMIF Creator erfasste Daten werden im „Correspondence Metadata Interchange Format“ (CMIF) kodiert, das von der TEI Correspondence Special Interest Group entwickelt und gepflegt wird. Anschließend werden die TEI XML-Daten auf Anbieterseite online verfügbar gemacht und vom Webservice per URL abgerufen. Die Metadaten der Briefeditionen werden somit dezentral aggregiert und erfordern nicht den Betrieb einer zentralen Speicherlösung. Der Webservice *correspSearch*, sowie das zugrundeliegende CMIF und das TEI-Elementset *correspDesc* wurden 2018 mit dem *Rahtz Prize for TEI Ingenuity*² ausgezeichnet.

Abbildung 1. Oberfläche des CMIF Creator, Schritt 1: Metadaten.

Jahr	Vom	An	Bis
1793	Georg Arnold Jacob	Friedrich Heinrich Jacob	
1794	Johann Arnold von Clemens	Friedrich Heinrich Jacob	
1795	Georg Arnold Jacob	Friedrich Heinrich Jacob	
1796	Friedrich Heinrich Jacob	Johann Arnold von Clemens	
1797	Johann Wolfgang von Goethe	Friedrich Heinrich Jacob	
1798	Friedrich Heinrich Jacob	Johann Friedrich Krieger	
1799	Georg Arnold Jacob	Friedrich Heinrich Jacob	
1800	Georg Heinrich Ludwig Nicolovius	Friedrich Heinrich Jacob	
1801	Johann Georg Schöner	Friedrich Heinrich Jacob	
1802	Georg Arnold Jacob	Friedrich Heinrich Jacob	

Abbildung 2. Listenansicht der Briefdatensätze.

War es bis vor kurzem nötig, das CMIF manuell zu kodieren, bietet der *CMIF Creator*³ nun eine benutzerfreundliche grafische Oberfläche, um eigene Brief-Metadaten in das nötige Format zu übertragen.⁴ Der mittlerweile in seiner zweiten Version verfügbare CMIF Creator bietet eine einfach zugängliche Möglichkeit, online und browserbasiert Briefeditionen in TEI-konforme CMIF-Dateien zu übertragen. Diese können dann in das Datenbanksystem von *correspSearch* überführt werden. Durch die browserbasierte Anwendung, die nicht auf serverseitige Speicher Routinen angewiesen ist, können Nutzer*innen die gesamte Verarbeitung lokal ausführen. Das bedeutet, dass die Nutzer*innen jederzeit vollständige Kontrolle über ihre Daten haben, da diese ausschließlich lokal gespeichert werden. Das Zwischenspeichern und Laden begonnener CMIF-Dateien ermöglicht weiterhin eine Bearbeitung über einen längeren Zeitraum. Der CMIF Creator unterstützt die Abfrage von Normdaten der *Gemeinsamen Normdatei* (GND) der Deutschen Nationalbibliothek⁵ über die von *lobid* bereitgestellte API⁶

, sowie das Abrufen von ortsbezogenen Normdaten aus der Datenbank des Ortsnamendienstes *GeoNames*⁷, und ermöglicht eine in das User Interface integrierte Auswahl und Zuweisung von Normdaten zu Korrespondent*innen und Orten. Die Unterstützung von Normdaten ist essentiell, da *correspSearch* für die Auflösung der Ambiguitäten von Personen- und Ortsnamen wesentlich auf die Nutzung von eindeutigen Normdaten-IDs zurückgreift. Die Oberfläche des Editors ist bewusst einfach und gleichzeitig so flexibel wie möglich gestaltet, um das CMIF Format im Frontend möglichst gut abzubilden und auch für weniger technisch versierte Nutzer*innen verfügbar zu machen.

Abbildung 3. Listenauswahl von Normdaten.

In den letzten Jahren hat sich JavaScript durch die breite Verwendung von hochentwickelten JavaScript-Frameworks zu einer der zentralen Skriptsprachen im Bereich des Frontend-Development entwickelt. Das zugrundeliegende Datenformat JSON bietet seinerseits eine bewährte und einfache Möglichkeit der Datenspeicherung im Frontend. Da XML-Frameworks oder -Bibliotheken für einen browserbasierten Editor dieses Umfangs als weitestgehend veraltet⁸ und unzureichend performant evaluiert wurden, erfolgte die Entscheidung schließlich zugunsten einer JavaScript- und JSON-basierten Anwendung. Durch die Crossplatform-Tauglichkeit von JavaScript bietet sich in der Perspektive darüber hinaus großer Spielraum für die Weiterentwicklung des CMIF Creators analog zum Format und den Bedürfnissen der Nutzer*innen.

Abbildung 4. Ansicht eines Briefdatensatzes.

Während der Arbeit im Editor verbleiben die Daten im JSON-Format und ermöglichen so eine dynamische Nutzung im Kontext des dem Editor zugrundeliegenden reaktiven JavaScript-Frameworks *vue.js*⁹. Durch die Verwendung von *vue.js* in Verbindung mit dem CSS-Framework *Bootstrap*¹⁰ wird jede Eingabe und Änderung der Nutzer*innen unmittelbar im JSON-Datenmodell abgebildet. Beim Speichervorgang werden die JSON-Daten über die *correspSearch*-eigene API in valides XML serialisiert, das dann später von *correspSearch* abgerufen werden kann. Im Falle von invaliden Eingaben enthält der CMIF Creator interne Validierungsfunktionen, die sowohl nach der Eingabe in einzelnen Feldern, als auch vor dem abschließenden Speichervorgang, die Eingaben auf technische Korrektheit überprüft. Finden sich Fehler, so werden diese nicht nur lokalisiert, sondern auch verlinkt, sodass Nutzer*innen direkt zum entsprechenden Eintrag springen können. Beim Laden von validen XML-Dateien in den Editor findet die Umwandlung in JSON ebenfalls über die *correspSearch*-API statt. Der CMIF Creator wird als OpenSource-Software entwickelt und unter der GNU LGPL lizenziert. Die Quellen sind auf der GitHub-Seite von *correspSearch*¹¹ frei zugänglich.

The screenshot shows the 'Speichern' (Save) step of the CorrespSearch interface. At the top, there are tabs for 'Start', '1: Metadaten', '2: Bibliographie', '3: Briefe', and '4: Speichern'. Below the tabs, a message explains that the digital letter index can be downloaded and saved on a computer, and that the CMIF file must be prepared online. The main content area shows three sections: '1: Metadaten' with a green bar and a checkmark, '2: Bibliographie' with a green bar and a checkmark, and '3: Briefe' with a red bar and a checkmark. Under '3: Briefe', there is a message '1 fehlerhafter Brief' (1 erroneous letter) and a link 'Zu Schritt 3: Briefe springen'. Below this, a specific error message is shown for 'Brief-ID 45 von Friedrich Heinrich Jacobi', stating that the 'Absender*in' (Sender) and 'Empfänger*in' (Recipient) fields are required and must be filled with a name. At the bottom, there is a checkbox 'Im JSON-Format speichern' and a button 'Als Entwurf speichern' (Save as draft).

Abbildung 5. Bei fehlerhaften Einträgen kann die CMIF-Datei nur als JSON-Entwurf gespeichert werden.

The screenshot shows the 'Speichern' (Save) step of the CorrespSearch interface. At the top, there are tabs for 'Start', '1: Metadaten', '2: Bibliographie', '3: Briefe', and '4: Speichern'. Below the tabs, a message explains that the digital letter index can be downloaded and saved on a computer, and that the CMIF file must be prepared online. The main content area shows three sections: '1: Metadaten' with a green bar and a checkmark, '2: Bibliographie' with a green bar and a checkmark, and '3: Briefe' with a green bar and a checkmark. At the bottom, there is a checkbox 'Im JSON-Format speichern' and a button 'Speichern' (Save).

Abbildung 6. Gibt es keine Fehler, können die Daten als valide .xml-Datei gespeichert werden.

10. <https://bootstrap-vue.js.org/>
11. <https://github.com/correspSearch>

Bibliographie

Andert, Martin / Frank Berger / Paul Molitor / Jörg Ritter (2015): *An Optimized Platform for Capturing Metadata of Historical Correspondence*, *Literary and Linguistic Computing Advance Access* 30 (4): 471–480. <https://doi.org/10.1093/lc/fqu027>.

Dumont, Stefan (2016): *CorrespSearch – Connecting Scholarly Editions of Letters*, *Journal of the Text Encoding Initiative*, Nr. Issue 10 (Dezember). <https://doi.org/10.4000/jtei.1742>.

Stadler, Peter (2012): *Normdateien in der Edition*, *editio* 26: 174–83.

TEI Consortium, Hrsg. o. J. *TEI P5: Guidelines for Electronic Text Encoding and Interchange. Version 3.4.0 vom 23.07.2018*. TEI Consortium. <http://www.tei-c.org/Guidelines/P5/>. [zuletzt abgerufen: 12. Oktober 2018]

TEI Correspondence SIG, Hrsg. (2015): *Correspondence Metadata Interchange Format (CMIF)*, <https://github.com/TEI-Correspondence-SIG/CMIF>.

Fußnoten

1. <https://correspsearch.net/>
2. <http://www.bbaw.de/presse/pressemittelungen/pressemittelungen-2018/Rahtz-Prize-for-TEI-Ingenuity-2018>
3. <https://correspsearch.net/creator/index.xql>
4. Der CMIF Creator basiert auf der schon in Andert et al. 2015 geäußerten Annahme, dass die Eingabe von Briefmetadaten möglichst effizient gestaltet sein sollte. Im Gegensatz zu dem von Andert et al. skizzierten Tool arbeitet der CMIF Creator allerdings standardbasiert. Darüber hinaus überlässt der CMIF Creator die Daten in der Obhut ihrer Ersteller*innen, die jederzeit vollständige Einsicht in die und volle Kontrolle über die (lokal) gespeicherten XML-Daten haben.
5. <http://www.dnb.de/>
6. <http://lobid.org/gnd/api>
7. <http://www.geonames.org/>
8. So lief z.B. die Unterstützung für betterForm in existdb Anfang 2018 aus, siehe hierzu auch den Eintrag in der Mailingliste <https://sourceforge.net/p/exist/mailman/message/36239166/> bzw. den entsprechenden Issue <https://github.com/eXist-db/exist/pull/1736>
9. <https://vuejs.org/>