

# **Die Schnittstelle Sammlung Online**

## Analyse der Schnittstelle und Weiterverarbeitung der Daten

### **Hausarbeit**

in der Veranstaltung Vertiefung Digital Humanities, Current Topics  
im Studiengang Digital Humanities, M.Sc.

Universität Trier  
FB II - Sprach-, Literatur- und Medienwissenschaften

Betreuerin: Dr. Joëlle Weis

Vorgelegt im Juli 2022 von:

Luisa Schmidt  
Schulstraße 12a  
56412 Ruppach-Goldhausen  
E-Mail: s2lascmi@uni-trier.de  
Matr.-Nr. 1375568

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Grundlagen</b>	<b>2</b>
2.1	Landesmuseum Württemberg . . . . .	2
2.2	Sammlung Online . . . . .	3
<b>3</b>	<b>Forschungsfragen</b>	<b>4</b>
<b>4</b>	<b>Die Schnittstelle Sammlung Online</b>	<b>5</b>
4.1	Datengrundlage und Standards . . . . .	5
4.2	Weiterverwendung und Normdaten . . . . .	6
4.3	Verwendung der Schnittstelle . . . . .	8
4.3.1	Gelungenes . . . . .	8
4.3.2	Optimierungsfaktoren . . . . .	9
<b>5</b>	<b>Visualisierung</b>	<b>12</b>
5.1	Python-Algorithmus und D3.js . . . . .	12
5.2	HTML-Datei . . . . .	15
<b>6</b>	<b>Evaluation und Fazit</b>	<b>16</b>
	<b>Literaturverzeichnis</b>	<b>17</b>
	<b>Anhang</b>	<b>18</b>

# Abbildungsverzeichnis

5.1	Ausschnitt aus der Funktion <code>extract_info()</code> . . . . .	13
5.2	Ausschnitt aus der Funktion <code>create_1800()</code> : Einzelne Objekte werden nach ihrem Entstehungsdatum in Listen gepackt . . . . .	13
5.3	Ausschnitt aus der Funktion <code>create_1800()</code> : Verschachtelung der erstellten Listen und Dictionarys ineinander . . . . .	14
5.4	Ausschnitt aus der entstandenen JSON-Datei . . . . .	14
5.5	Umsetzung der Suchleiste im HTML-Dokument . . . . .	15

# 1. Einleitung

Das Landesmuseum Württemberg mit Standort in Stuttgart ist im Bundesland Baden-Württemberg eine zentrale Institution zum Erhalt württembergischer Kultur und regionaler Werke über viele Jahrhunderte hinweg. Das Landesmuseum ist „eines der größten kulturhistorischen Museen in Deutschland“ (Gesellschaft zur Förderung des Landesmuseums Württemberg e.V., k.D.), dessen Bestand aktuell mehr als eine Million Objekte zählt (Gruber, 2022).

Für den mobilen Zugriff auf die Objekte bietet das Museum die Schnittstelle *Sammlung Online* an. Dort können Informationen zu den Sammlungsobjekten sowie Bilder und Metadaten abgerufen oder heruntergeladen werden. Die Schnittstelle wird laufend um Objekte und Daten erweitert: derzeit umfasst sie ca. 18.000 Objekte, welche über alle Epochen, Orte und Objekttypen verteilt sind (Klingspor, 2022).

Ziel dieser Hausarbeit ist es, die Umsetzung der Sammlung Online anhand verschiedener Parameter tiefgehend zu untersuchen, ihre Nutzungsmöglichkeiten zu bewerten und anschließend eine Visualisierung mit den aus der Schnittstelle abgegriffenen Daten zu erstellen. Dafür werden im nächsten Kapitel zunächst das Landesmuseum Württemberg als auch die Sammlung Online noch einmal im Detail vorgestellt und somit die Grundlagen für weitere Analysen gelegt. In Kapitel 3 folgen die Forschungsfragen, welche zu dieser Arbeit formuliert wurden, bevor dann in Kapitel 4 die Sammlung Online genauer betrachtet und beurteilt wird. Dies geschieht in Bezug auf die Parameter Datengrundlage, verwendete Standards und Normdaten sowie die Weiterverwendungsmöglichkeiten der von der Schnittstelle bereitgestellten Daten. Außerdem wird in diesem Kapitel die Benutzung der Schnittstelle auf positive und negative Faktoren überprüft sowie Optimierungspotenziale aufgezeigt. Zum Abschluss der Ausarbeitung wird ein Visualisierungsvorhaben durchgeführt anhand der aus der Sammlung Online abgegriffenen Daten. Die Visualisierung wird mithilfe eines Python- sowie D3.js-Skripts erstellt und in eine HTML-Seite eingebunden, was in Kapitel 5 ausführlich beschrieben wird.

## 2. Grundlagen

In diesem Grundlagenkapitel wird zunächst das Landesmuseum Württemberg als Institution näher beschrieben. Daraufhin folgt eine kurze Einführung zur Sammlung Online, über die das Landesmuseum Württemberg seine Objekte digital zugänglich macht, sodass im Folgekapitel genauer auf die Schnittstelle und deren Vor- und Nachteile eingegangen werden kann.

### 2.1 Landesmuseum Württemberg

Das Landesmuseum Württemberg (LMW) wurde am 16. Juni 1862 von König Wilhelm I. von Württemberg unter dem Namen *Königliche Staatssammlung vaterländischer Kunst- und Alterthumsdenkmale* gegründet (Gruber, 2022). Die Sammlung wurde zunächst in der Königlichen Landesbibliothek in Stuttgart ausgestellt, welche 1886 erbaut wurde (ebd.). Die Ziele damals waren es, eine breite Sammlung von Objekten über verschiedene Gattungen, Gewerbe sowie Entstehungszeitpunkte hinweg bereitzustellen, die als „zentrale Sammlung (...) württembergische[r] Geschichte“ (ebd.) dienen sollte. Nicht nur die Zusammenstellung von Objekten aus dem damaligen Königreich Württemberg spielte im 19. Jahrhundert eine Rolle, auch sollten die Objekte der breiten Öffentlichkeit präsentiert werden, sodass sie den Bürgerinnen und Bürgern gleichermaßen zugänglich waren (ebd.).

Die Bestände des Museums wurden nach Ende des Ersten Weltkriegs auf das Alte und Neue Schloss in Stuttgart aufgeteilt (Gruber, 2022). Dies war nötig, da die Räumlichkeiten der Königlichen Landesbibliothek nicht mehr genug Platz für die Sammlung boten, da diese über Jahrzehnte gewachsen war und ihren Bestand deutlich ausweiten konnte (ebd.). Zusammengeführt wurden die Bestände dann nach dem Zweiten Weltkrieg im Alten Schloss im Stuttgarter Zentrum, wo sich das LMW auch heute noch befindet (ebd.).

Im Jahr 2022 zählte das LMW über eine Million Objekte zu seinem Bestand, welcher immernoch vor allem darauf fokussiert ist, Objekte und Werke aus dem Raum Württemberg abzubilden (Gruber, 2022). Darüber hinaus werden aber auch Zeitausstellungen geboten, welche andere Regionen zum thematisieren (ebd.). Das LMW hat sich zum Ziel gesetzt, das materielle und kulturelle Erbe der Region Württemberg insgesamt zu bewahren, weshalb der Bestand des Museums nicht auf bestimmte Epochen oder Themen beschränkt ist (ebd.). Das Museum bietet folglich ein breites Sammlungsspektrum, dessen Objekte in der museumseigenen Werkstatt restauriert und so erhalten werden können (ebd.).

## 2.2 Sammlung Online

Das Landesmuseum Württemberg ermöglicht es nicht nur, seine Bestände vor Ort im Alten Schloss in Stuttgart zu begutachten. Es bietet darüber hinaus die Möglichkeit, sich die Objekte in der *Sammlung Online* (SO) über die Webseite des Museums anzuschauen. In dieser digitalen Komponente des LMW sind laut Museum mehr als 12.000 Objekte erfasst, im Suchfeld der SO wird sogar die Anzahl von 18.000 Objekten genannt (Klingspor, 2022). Die Objekte werden mit Fotografien und weiteren, ausführlichen Informationen zur Verfügung gestellt, wodurch die SO auch von Zuhause oder unterwegs aus eine gute Möglichkeit bietet, den Bestand des LMW zu erkunden (ebd.). Ebenso wie der Museumsbestand des LMW ist die SO nicht auf eine bestimmte Epoche, Objektart oder andere Merkmale festgelegt, sodass ein breites Spektrum an Stücken aus der Region Württemberg auch digital ausgestellt wird (ebd.). Darüber hinaus werden online immer wieder Sammlungen zu bestimmten Themenkomplexen gezeigt, wodurch virtuelle Ausstellungen und Ausstellungsräume entstehen, die das LMW dann unter anderem auch auf der Plattform Google Arts & Culture präsentiert (ebd.).

Die Objekte, die in der SO hinterlegt sind, sind durch ihre Digitalisierung rund um die Uhr für Interessierte zugänglich, auch wenn die Originale gerade nicht physisch ausgestellt werden (Klingspor, 2022). Besucherinnen und Besucher des digitalen Museums haben also die Möglichkeit, sich die Objekte jederzeit online anzuschauen. Zudem erfolgt die Ergänzung der SO im laufenden Betrieb, es werden also ständig neue Objekte digitalisiert und zur Sammlung hinzugefügt mit dem Ziel, den Gesamtbestand des Landesmuseums zu repräsentieren (ebd.). Somit ist die SO noch kein abgeschlossenes Projekt und wird perspektivisch in den kommenden Jahren um weitere Sammlungsobjekte erweitert werden. Die SO kooperiert darüber hinaus mit weiteren Sammlungen wie *Museum Digital*, *Landeskunde entdecken online* oder *Gothic Ivories* und ist mit diesen digitalen Angeboten vernetzt (Klingspor, 2022). Im folgenden Kapitel 3 werden zunächst die Forschungsfragen zu dieser Ausarbeitung formuliert. Danach wird in Kapitel 4 die bereitgestellte Schnittstelle *Sammlung Online* genauer nach verschiedenen Parametern analysieren und bewertet, aufbauend auf den in diesem Kapitel bereitgestellten, grundlegenden Informationen.

### 3. Forschungsfragen

In der vorliegenden Ausarbeitung können inhaltlich zwei Teilbereiche voneinander abgegrenzt werden. Im ersten Teil der Arbeit geht es darum, die Schnittstelle *Sammlung Online* des Landesmuseums Württemberg genau zu untersuchen und ihre Stärken und Schwächen herauszustellen. Hierbei sollen auch, wo immer nötig und möglich, konkrete Verbesserungsvorschläge gegeben werden. Aus diesem Vorhaben ergeben sich konkret die beiden Forschungsfragen (FF):

- FF 1: Welche Datengrundlage, Standards und Normdaten werden in der Sammlung Online verwendet und wie können die Daten der Schnittstelle weiterverwendet werden?
- FF 2: Welche Faktoren sind bei der Verwendung der Schnittstelle gelungen und wo besteht Optimierungspotenzial?

Darüber hinaus besteht der zweite Teil der Ausarbeitung darin, ein konkretes Visualisierungsprojekt mit Daten aus der Schnittstelle umzusetzen, wobei beispielsweise auch die eigene Programmierfähigkeiten eingesetzt werden sollen. Aufgrund dieser praktischen Ausrichtung des zweiten Teils der Arbeit werden an dieser Stelle keine Forschungsfragen formuliert. Das Vorgehen und die benötigten Ressourcen für die Visualisierung werden in Kapitel 5 genauer beschrieben und erläutert.

## 4. Die Schnittstelle Sammlung Online

In diesem Kapitel wird die Schnittstelle Sammlung Online (SO) genauer beschrieben, analysiert sowie auf ihre Umsetzung beurteilt. Jeder dieser genannten Schritte erfolgt direkt im jeweiligen Unterkapitel zu den untersuchten Parametern *Datengrundlage und Standards* sowie *Weiterverwendung und Normdaten*. Damit zielen die folgenden Unterkapitel 4.1 und 4.2 auf die Beantwortung der Forschungsfrage 1 ab.

Im Unterkapitel 4.3 erfolgt dann ein Bericht über die *Verwendung der Schnittstelle*. Hier werden gelungene und verbesserungswürdige Faktoren in der Konzeption der Schnittstelle hervorgehoben, wodurch auf die Forschungsfrage 2 eingegangen wird.

### 4.1 Datengrundlage und Standards

Als Datengrundlage der Schnittstelle SO des Landesmuseums Württemberg (LMW) dienen jene Objekte, die zum physischen Bestand des Museums gehören (Klingspor, 2022). Wie bereits im vorherigen Kapitel 2 erwähnt, soll die SO einen Querschnitt aus allen Objekten des Museums darstellen, wodurch sie nicht auf bestimmte Epochen, Werke, Künstler oder ähnliches beschränkt ist (ebd.). Um den großen Bestand von 18.000 Objekten besser greifbar zu machen, ist die Sammlung nach mehreren vorgegebenen Parametern durchsuchbar, wie nach einem Begriff, einem bestimmten Zeitraum oder einer Person.

Die Daten, welche zu jedem einzelnen Objekt angegeben werden, sind recht detailliert. Zu den Sammlungsobjekten wird unter anderem ein Bild des Objekts bereitgestellt, eine Lizenzangabe wird gegeben, ein kurzer Text mit den wichtigsten Informationen über das Objekt sowie Eckdaten in Bezug auf Größe, Material und Maße. Zudem erfolgt eine geschichtliche Einordnung des Objekts. Diese Informationen sind für Interessierte in einem Deckblatt zusammengestellt worden und stehen zum Download bereit.

Zudem können eine LIDO- und eine JSON-Datei zum Objekt sowie eine Gesamt-JSON-Datei mit den Daten aller Objekte heruntergeladen werden. Diese beiden Datenformate sind Standards und daher generell gut weiterverwendbar. *JSON*, was für JavaScript Object Notation steht, ist analog zu einem Python-Dictionary aufgebaut: Es besteht aus sogenannten Key-Value- bzw. zu Deutsch Schlüssel-Wert-Paaren, bei denen der Key einzigartig sein muss, da über diesen der angegebene Value identifiziert wird. Schlüssel und Wert werden dabei durch einen Doppelpunkt voneinander getrennt, der Schlüssel steht dabei vor dem Doppelpunkt. Ein Beispiel für ein solches Key-Value-Paar ist `"objekt_name": "Etruskischer Griffspiegel"`. Da JSON über verschiedenste Anwendungsfälle hinweg weit verbreitet ist, weshalb



im Folgenden nicht tiefer auf die Spezifika des Formats eingegangen wird. Bei *LIDO* - Lightweight Information Describing Objects - handelt es sich um ein XML-basiertes Standardformat, welches vor allem im Museums- und Kunstkontext zu finden ist (ICOM International Committee for Documentation, k.D.). LIDO ist damit ein recht disziplinspezifischer Standard, welcher aber „for all kinds of objects, e.g. art, architecture, cultural history, history of technology, and natural history“ (ebd.) genutzt werden kann. Im Element `<objectDescriptionSet>` können beispielsweise deskriptive Informationen zum Objekt bereitgestellt werden, das Element `<objectMeasurementsSet>` dient der Notation von Maßen (ICOM-CIDOC LIDO Working Group, 2021). Da jedoch eine genauere Beschreibung des Standards, seiner Elemente und Attribute und deren Verwendung über den Rahmen dieser Ausarbeitung hinaus gehen würde, wird an dieser Stelle auf die Dokumentation<sup>1</sup> verwiesen.

Die Verwendung dieser beiden Formate in der Schnittstelle kann durchaus als sinnvoll und folglich positiv evaluiert werden. Da es sich bei beiden Formaten um offene Standards handelt, können die Daten gut von Interessierten eingesehen und weiterverwendet werden. Die bereitgestellten Daten sind somit anknüpfbar und für weiterführende Analysen nutzbar. Besonders JSON ist, wie bereits erwähnt, ein sehr gängiges Format. Da vermutlich ein Großteil der potenziell an einer Weiterverarbeitung Interessierten bereits Erfahrung mit JSON hat, ist eine Einstiegshürde kaum vorhanden. Doch auch ohne Vorkenntnisse ist JSON sehr zugänglich, da die Daten in einer simplen Form strukturiert und für Menschen bereits gut ohne computerbasierte Weiterverarbeitung lesbar sind. Bei LIDO handelt es sich zwar um einen Standard, dieser ist aber recht spezifisch und daher vermutlich nur einem kleinen Teil potenzieller Nachnutzenden bekannt. Hier bedarf es daher einer längeren Einarbeitung in das Schema, auch, weil es recht komplex und umfangreich ist. Da LIDO aber im Museumskontext unabdingbar ist und zudem auch auf JSON zurückgegriffen werden kann, ist über die Einstiegshürde hinwegzusehen.

## 4.2 Weiterverwendung und Normdaten

Unter Weiterverwendung wird in diesem Unterkapitel die potenzielle Nachnutzung der Daten durch Forschenden verstanden und vor allem dahingehend untersucht, inwiefern diese rechtlich möglich ist. Bereits auf der Startseite der Schnittstelle ist angegeben, dass „das Landesmuseum Württemberg (...) alle in der Online-Sammlung publizierten Objektdaten (Bilder und Metadaten) für eine Nachnutzung und Weiterverwendung frei[stellt]“ (Klingspor, 2022). Zusätzlich wird auf die Creative Commons-Lizenz CC BY-SA 4.0 verwiesen, wonach die Daten unter Nennung des Rechteinhabers geteilt, bearbeitet und sogar für kommerzielle Zwecke weitergenutzt werden

---

<sup>1</sup>Die Dokumentation ist zu finden unter <http://lido-schema.org/schema/v1.1/lido-v1.1.html#objectMeasurementsSet>.

dürfen (Creative Commons, k.D.). Die Weitergabe der Daten muss unter gleicher Lizenz erfolgen (ebd.). Die Bereitstellung der Daten unter genannter Lizenz ist stark positiv zu bewerten, denn so können Interessierte die Daten flexibel nachnutzen. Die bereitgestellten Informationen werden für die weitergehende Forschung attraktiv und sind anschlussfähig, was ein durchaus wichtiger Faktor ist.

Die Lizenz ist ebenso bei den einzelnen Objekten in den Downloaddateien und unter den bereitgestellten Bildern zu finden. Durch die häufig angegebenen Lizenzbestimmungen wird für Transparenz in der Rechtefrage gesorgt und die Nutzenden müssen nicht aufwendig nach Angaben zu Weiterverwendungsmöglichkeiten suchen, was Interessierte oftmals bereits davon abhält, die Daten weiterzunutzen. Die Besucher der Schnittstelle wissen immer genau, unter welcher Lizenz das jeweilige Objekt verfügbar ist. Insgesamt ist also die Nachnutzung und Weiterverwendung der Daten in der SO gelungen und transparent organisiert, wodurch die Schnittstelle eine Vorbildfunktion für andere digitale Sammlungen einnimmt.

Auch auf die Verwendung von Normdaten sollte bei der Analyse einer Schnittstelle geachtet werden. Normdaten sind „Daten über Objekte aller Art, deren unmittelbarer Zweck in der Zusammenführung unterschiedlicher identifizierender Benennungen und Begriffe für das jeweilige Objekt besteht“ (Wiechmann, 2016, S. 380). Sie dienen also der Disambiguierung und eindeutigen Identifikation von Objekten über einzelne Institutionen hinweg, weshalb sie von großer Bedeutung sind. Bekannte Normdaten sind die GND-Nummern, Wikidata-IDs oder die VIAF-ID. Jedoch werden weder auf der Detailseite zu den einzelnen Objekten, noch in den bereitgestellten JSON-Dateien die Objekte über Normdaten referenziert. Zur Identifikation wird hier eine museumsinterne ID verwendet, über die die Objekte eindeutig zugeordnet werden können. In den LIDO-Dateien werden teilweise Normdaten verwendet, um Orte oder Institutionen zu verknüpfen, die mit dem Objekt in Verbindung stehen: Neben den drei bereits oben genannten Normdaten werden beispielsweise auch die der Nationalbibliothek Frankreichs sowie der Library of Congress in den USA verwendet. Das Objekt an sich wird nicht über Normdaten referenziert, wodurch über das LMW hinaus keine Möglichkeit zur eindeutigen Referenzierung besteht. Da es sich bei den Objekten im Bestand des LMW um sehr spezifische und an die Region Württemberg gebundene Objekte handelt, welche global kaum eine Bedeutung haben, sollte dieser Faktor nicht allzu stark gewichtet werden. Dennoch wäre zumindest eine Verknüpfung mit den Normdaten der Deutschen Nationalbibliothek über GND-Nummern wünschenswert, um die Objekt auf nationaler Ebene innerhalb Deutschlands referenzieren zu können.

Die beiden vorherigen Unterkapitel zusammenfassend kann die SO in Bezug auf die Parameter *Datengrundlage*, *Standards*, *Weiterverwendung* und *Normdaten* als gelungene Schnittstelle gewertet werden.

## 4.3 Verwendung der Schnittstelle

Nach dem in den vorangegangenen Unterkapiteln 4.1 und 4.2 Grundlegendes in Bezug auf Datengrundlage, genutzte Standards, Weiterverwendung der Daten sowie zur Nutzung von Normdaten in der SO analysiert wurde, folgt nun eine detaillierte Beurteilung der Verwendung der Schnittstelle. Hierbei werden sowohl gelungene Faktoren als auch Bereiche, in denen ein Optimierungspotenzial besteht, hervorgehoben und genauer betrachtet.

### 4.3.1 Gelungenes

Als erster Punkt positiv zu nennen ist die Schnittstellen-Oberfläche. Das Vorhandensein einer Benutzungsoberfläche (GUI), welche in der SO als herkömmliche Webseite aufgebaut ist, über die die Daten abgefragt werden können, vereinfacht die Arbeit mit der Schnittstelle ungemein. Die Nutzenden können Suchanfragen in einem dafür vorgegebenen Feld formulieren und die Ergebnisse präzisieren, indem beispielsweise ein bestimmter Zeitraum für das Entstehungsjahr des Objekts angegeben wird. Somit können die in der SO enthaltenen Daten abgefragt werden, ohne komplexe Befehle in der Kommandozeile des PCs eingeben oder Suchabfragen in der URL der Webseite formulieren zu müssen. Die Einstiegshürde für die Nutzung der SO ist dadurch sehr gering und setzt keinerlei zusätzlichen Kenntnisse seitens der Nutzenden voraus, wie dies oft bei anderen Schnittstellen der Fall ist.

Das Vorhandensein einer GUI für die Schnittstelle ist ein zentraler Faktor für die hohe Nutzungsfreundlichkeit der SO. Darüber hinaus ist die Schnittstelle rein optisch gelungen, denn die SO ist übersichtlich aufgebaut, sodass sich Interessierte schnell und vor allem intuitiv zurechtfinden. Für die Nutzung der Schnittstelle ist daher keine Erklärung notwendig, sodass man direkt beginnen kann, die Sammlung zu erkunden und Objekte zu entdecken. Hinzu kommt, dass mit vielen Bildern sowie verschiedenen textuellen Inhaltselementen gearbeitet wurde, die Abwechslung bieten: Neben simplen Textbausteinen für die generellen Informationen zum Objekt finden sich ebenso Stichworte, Tabellen, Akkordeons oder Kästen mit Eckdaten auf den Seiten. Besonders die Akkordeons lassen die Schnittstelle schlanker und weniger textlastig wirken, da sie erst bei Anklicken weiteren Text präsentieren. Zudem wurde darauf geachtet, dass die Seiten zu einzelnen Objekten trotz ihrer Vielfältigkeit an Elementen einheitlich aufgebaut sind, sodass sich die Nutzenden schnell zurecht finden und visuellen Mustern folgen können. Die genannten Punkte tragen alle wesentlich zur hohen Nutzungsfreundlichkeit der Schnittstelle bei, welche somit insgesamt dazu einlädt, länger auf der Seite zu verweilen und die SO zu explorieren. Als dritter Punkt positiv einzuordnen ist die Nutzung einer Verschlagwortung und Einordnung der Objekte in der Schnittstelle. Zu den einzelnen Objekten, die in der SO bereitgestellt werden, sind jeweils mehrere Schlagwörter hinterlegt. Diese

Schlagwörter können dann mit dem Inhalt der Suchanfrage abgeglichen werden und das Objekt letztendlich darüber in der Schnittstelle gefunden werden. Zudem wird eine Einordnung der Objekte vorgenommen. Auch hier werden zusätzliche Daten zum Objekt angegeben, im Unterschied zu den Schlagwörtern sind die Einordnungen jedoch in Kategorien unterteilt. Über die Einordnung *Hergestellt: 1800* wird beispielsweise das betreffende Objekt innerhalb der Schnittstelle im Jahr 1800 angesiedelt und kann über die Verwendung eines Filters auf die Jahreszahl über die Suchmaske gefunden werden. Weitere Einordnungen sind beispielsweise *Wurde abgebildet (Akteur)* oder *Wurde genutzt von*, sodass auch hierunter aufgeführte Personen mit dem Objekt verknüpft werden können. Die Verschlagwortung und Einordnung der Objekte dient dazu, dass die Suchoption in der Schnittstelle funktioniert und die Objekte anzeigt, welche zur gestellten Anfrage passen. Da die Suchfunktion ein essentieller Bestandteil der SO ist und aufgrund der großen Anzahl der Objekte, die in der Sammlung enthalten sind, unverzichtbar wäre, sind hier Verschlagwortung und Einordnung sinnvoll eingesetzt. Zusammenfassend tragen also die beiden Faktoren wesentlich zur Funktionalität und Nutzbarkeit der Schnittstelle bei und werden daher als gelungen betrachtet.

Zuguterletzt und anknüpfend an den vorausgehenden Punkt sind die internen Verlinkungen innerhalb der SO positiv zu erwähnen. Hierbei handelt es sich um Verknüpfungen zwischen einzelnen Objekten mit Personen, Orten sowie Zeitabschnitten. Dies gelingt über die Einordnungen und Schlagwörter, denn über diese können Nutzende zu anderen Objekten springen, bei denen die gleiche Verschlagwortung hinterlegt ist. Durch diese Option der Querverlinkungen und Verknüpfungen zwischen den Objekten in der Sammlung wird eine weitere Art geschaffen, über die man die Sammlung entdecken kann. Sie bietet den Nutzenden die Möglichkeit, sich von den eigenen Interessen von Objekt zu Objekt leiten zu lassen, ohne eine neue Suchanfrage stellen zu müssen. Dadurch ist eine stärkere Exploration der Inhalte gegeben und die SO gewinnt insgesamt an Attraktivität und Funktionalität.

### 4.3.2 Optimierungsfaktoren

Neben den vielen gelungenen Faktoren lässt sich bei der Verwendung der Schnittstelle auch Einiges erkennen, bei dem Optimierungspotenzial besteht. Dies bezieht sich sowohl auf die eigentliche Nutzung der Schnittstelle, als auch auf die Weiterverwendung der aus ihr abgerufenen Daten.

Zunächst fällt auf, dass es keine Übersichtsgrafiken darüber gibt, welche Objekte in der Sammlung enthalten sind. Besucherinnen und Besucher werden direkt mit dem Suchfeld konfrontiert, ohne zu wissen, was in der Sammlung enthalten und recherchierbar ist. Ohne einen solchen Anhaltspunkt ist es schwierig, die Sammlung zu explorieren. Grafiken wären hier eine gute Möglichkeit, anhand ausgewählter Para-

meter eine Übersicht über die Sammlung zu geben und den Nutzenden ein erstes, grobes Verständnis zu vermitteln. Übersichtsgrafiken könnten beispielsweise die Verteilung der Objekte in der SO anhand von Themen, Zeitstempeln oder beteiligten Personen aufbereiten. So könnte die Sammlung explorativ genutzt werden und nicht nur, wenn Interessierte ein bestimmtes Objekt suchen oder eine genaue Vorstellung von dem haben, was sie finden möchten. Entsprechende Grafiken würden die Sammlung nochmals attraktiver machen für Personen, die zufällig auf sie stoßen und dann schauen möchten, was die Sammlung bietet. Zusätzlich würden Übersichtsgrafiken Suchfilter und Schlagworte in ihrer Funktionalität unterstützen, indem sie beispielsweise die Informationen liefern, welche für eine Schlagwortsuche benötigt werden. Die Idee der Übersichtsgrafiken wird in Kapitel 5 weiter aufgegriffen und anhand eines Beispiels ausgeführt.

Ein zweiter Kritikpunkt lässt sich beim Abruf der bereitgestellten Daten aus der Schnittstelle erkennen. Im vorangegangenen Kapitel 4.3.1 wurde die Verwendung von Schlagwortsuchen und Filtermöglichkeiten gelobt, durch die sich Nutzende eine Untersammlung nach ihren Kriterien zusammenstellen können. Ein großes Problem ist jedoch, dass die Untersammlung nicht akkurat weiterverarbeitet werden kann. Es gibt keine Möglichkeit, eine Datei mit allen LIDO- oder JSON-Dateien zu den Objekten herunterzuladen, die in der Teilsammlung enthalten sind, denn es lassen sich weiterhin nur einzelne Dateien abgreifen oder eben die Datei zur Gesamtsammlung. Dies erschwert die Weiternutzung der Daten aus der Schnittstelle erheblich, weil für Nutzende nur folgende Optionen bestehen: Entweder, sie müssen für jedes Objekt aus der Teilsammlung alle JSON- bzw. LIDO-Dateien einzeln herunterladen. Dies mag bei einer Größenordnung von 10 Objekten funktionieren, wenn es jedoch um 50 oder mehr Objekte geht, wird dieses Vorhaben schnell nicht mehr umsetzbar. Weiter können die Nutzenden auch auf ein selbst geschriebenes Skript zurückgreifen, welches automatisiert die einzelnen Dateien herunterlädt oder aus der Gesamt-JSON-Datei eben die herausfiltert, die zur erstellten Untersammlung gehören. Hierfür sind allerdings fundierte Programmierkenntnisse erforderlich, sodass diese Möglichkeit für einen größeren Teil der Nutzenden keine Option darstellt. Um eine gute Weiternutzung der Daten zu ermöglichen, sollte seitens der Schnittstelle die Möglichkeit bestehen, für die erstellte Teilsammlung einen ZIP-Ordner mit allen relevanten JSON- oder LIDO-Dateien herunterzuladen. Erst diese Option macht die Verarbeitung der aus der Schnittstelle gewonnenen Daten wirklich praktikabel als auch attraktiv für Forschende, was für das Landesmuseum Württemberg ein wichtiges Ziel sein sollte.

Für die Weiterverwendung der Daten aus der Schnittstelle ist darüber hinaus kritisch anzumerken, dass in der JSON-Gesamtdatei deutlich weniger Informationen je Objekt angegeben sind, als in den objektspezifischen JSON-Dateien. Dies ist einerseits

verständlich, da die Gesamtdatei über alle 18.000 in der SO enthaltenen Objekte nicht in einer unhandlichen Dateigröße ausufern sollte, um noch gut nutzbar zu sein. Andererseits entstehen jedoch dadurch auch verschiedene Datengrundlagen, was bei einer Nachnutzung der Daten zu bedenken ist. So muss man sich vor der Weiterverarbeitung darüber im Klaren sein, welche Informationen man für die eigene Analyse benötigt und in welcher der Dateien diese zu finden sind. Falls man dann auf die Einzeldateien zurückgreifen möchte, kommt allerdings das Defizit zum Tragen, dass man nicht alle benötigten Einzeldateien gesammelt herunterladen kann. Dies muss - wie bereits im vorherigen Absatz erwähnt - einzeln erfolgen. Für eine gute Nutzung der Daten in der Schnittstelle sollten die Informationsunterschiede in den verschiedenen Dateien ausgeglichen werden. Falls dies aufgrund der Dateigröße nicht praktikabel sein sollte, sollte zumindest ein Hinweis auf die unterschiedlichen Datengrundlagen auf der Webseite platziert werden.

Ein letzter Kritikpunkt an der Schnittstelle bezieht sich ebenfalls auf die Weiterverarbeitung der abgerufenen Daten. Hierbei fällt auf, dass in den JSON-Dateien keine gesonderte Information zum Bau-, Entstehungs- bzw. Findjahr des Objekts ausgezeichnet ist. Die Jahresangabe ist lediglich zusammen mit dem Namen des Objekts unter dem Key `objekt_name` zu finden und somit für die Weiterverarbeitung zunächst nicht greifbar. Um die Jahreszahl zu extrahieren und für weitere Analysen zugänglich zu machen, muss man daher auf ein Skript zurückgreifen, welches die Jahreszahl im Objektnamen identifizieren und extrahieren kann. Dies ist insgesamt recht umständlich und auch hier müssen fundierte Programmierkenntnisse vorliegen, mit denen man sich ein etwaiges Skript schreiben kann. Ein Grund dafür, die Jahreszahl nicht als eigenständiges Key-Value-Paar in den JSON-Dateien aufzunehmen, könnte seitens des LMW gewesen sein, dass nicht bei allen Objekten ein konkretes Jahr zu finden ist. Gerade bei den sehr alten Objekten aus dem ersten oder zweiten Jahrhundert beispielsweise lässt sich häufig kein genaues Objektjahr angeben und so hätte man in den JSON-Dateien oft keinen Wert, der angegeben werden könnte. Ein leerer Wert stellt jedoch für die Erstellung der Key-Value-Paare in JSON-Dateien kein Hindernis dar, weshalb die Angabe eines Keys `objekt_jahr` sinnvoll wäre. Für die Nachnutzung der Daten wäre dann ein immenser Vorteil gegeben, denn die Jahreszahl ist eine Information, die häufig für weiterführende Analysen der Daten verwendet wird und somit für diese zugänglich wäre.

Resümierend zu diesem und dem vorausgegangenen Unterkapitel 4.3.1 lässt sich feststellen, dass die Schnittstelle SO bereits in vielen Aspekten sehr gut aufgestellt ist und daher eine Nutzung interessant erscheint. Es konnten jedoch, gerade in Bezug auf die Nachnutzung der Daten, auch einige Defizite aufgezeigt werden, die es seitens des Landesmuseums Württemberg auszubessern gilt, um eine angenehme Weiterverwendung der bereitgestellten Daten langfristig zu garantieren.

## 5. Visualisierung

Wie bereits im vorigen Kapitel 4.3.2 erwähnt, mangelt es in der Sammlung Online (SO) an Übersichtsgrafiken, die den Nutzenden den Einstieg in die Schnittstelle erleichtern würden. An diesem Punkt setzt das im Folgenden beschriebene Visualisierungsvorhaben an. Es soll eine Grafik in Form eines animierten und interaktiven Kuchendiagramms erstellt werden, welche die in der Sammlung enthaltenen Objekte aus dem 19. Jahrhundert darstellt und in verschiedene Zeitabschnitte gliedert. Die Grafik soll darüber hinaus auch die museumsinterne ID der Objekte beinhalten, so dass man über eine Suchleiste direkt auf die SO gelangen und sich dort das jeweilige Objekt anzeigen lassen kann. Für die Erstellung der Visualisierung ist zum einen ein Python-Algorithmus notwendig, welcher die vorhandenen Daten aufbereitet. Zum anderen wird ein bereits vorhandenes Skript aus der JavaScript-Bibliothek D3.js verwendet, um die finale Visualisierung zu erstellen.

### 5.1 Python-Algorithmus und D3.js

Der selbst geschriebene Python-Algorithmus ist das zentrale Element, mit dem die vorhandenen Informationen aus der Gesamt-JSON-Datei so aufbereitet werden, dass sie schließlich im Kuchendiagramm visualisiert werden können. Der Python-Code kann auf GitHub eingesehen werden<sup>1</sup>.

Der Algorithmus besteht aus mehreren Funktionen, die nacheinander in der **main-Funktion** in den Zeilen 154-158 aufgerufen werden und Parameter übergeben bekommen. Zu Beginn des Algorithmus wird in der Funktion `read_file()` die Gesamt-JSON-Datei aus der SO eingelesen und zurückgegeben. Dann folgt die erste der beiden Schlüsselfunktionen: Die Funktion `extract_info` extrahiert die Informationen zu Entstehungsjahr und Objekt-ID aus der als Parameter übergebenen, eingelesenen JSON-Datei (Code Z. 13). Dafür wird mittels einer Laufvariable `i` über die einzelnen Einträge in der Datei iteriert, die wie Python-Dictionaries als Key-Value-Paare aufgebaut sind. Dabei wählt der Algorithmus den Key `"objekt_name"` aus und zählt die letzten vier Zeichen des hinterlegten Values ab (Z. 21f.) Dort liegt in der Regel die Jahreszahl, welche für das Objekt vermerkt wurde. Diese letzten vier Zeichen werden extrahiert und in der Variable `year` gespeichert. Da nicht bei allen Sammlungsobjekten eine genaue Jahreszahl vermerkt ist, wird in den Zeilen 25 bis 28 mit der Methode `.isdigit()` überprüft, ob es sich bei dem Inhalt der Variable `year` um eine Zahl handelt. Teile der Funktion sind in Abbildung 5.1 visualisiert.

---

<sup>1</sup>Das Repository ist erreichbar unter <https://github.com/s2lascmi/HausarbeitVertiefung>, der Python-Code ist hier hinterlegt [https://raw.githubusercontent.com/s2lascmi/HausarbeitVertiefung/master/Code\\_Hausarbeit\\_Vertiefung.py](https://raw.githubusercontent.com/s2lascmi/HausarbeitVertiefung/master/Code_Hausarbeit_Vertiefung.py).

Abbildung 5.1: Ausschnitt aus der Funktion `extract_info()`

```

18     for data[i] in data:
21         year = data[i]["objekt_name"]
22         year = year[-4:]
25         if year.isdigit():
26             data[i]["objektjahr"] = year
27         else:
28             data[i]["objektjahr"] = 102983

```

Ist dies gegeben, wird zum Dictionary ein neues Paar mit Key "objektjahr" und der Jahreszahl als Wert eingefügt, ansonsten wird ein Platzhalter - hier 102983 - eingefügt. Die Objekt-ID kann über den Key "objekt\_id" herausgefiltert werden. Sie wird nicht unter dem Schlüssel ID gespeichert, was naheliegender wäre, sondern unter dem Schlüssel "name". So kann aber das vorhandene D3.js-Skript einfach weiterverwendet werden. Zuletzt werden in den Zeilen 36 bis 48 die Key-Value-Paare aus dem Dictionary entfernt, die nicht weiter benötigt werden und das bereinigte Dictionary mit den beiden Einträgen für Jahreszahl und Objekt-ID wird zurückgegeben.

Das entstandene Dictionary dient als Parameter für die als nächstes aufgerufene Funktion `create_1800`, in welcher die Grundstruktur für die am Ende entstehende JSON-Datei gelegt wird, die dann im D3.js-Skript verwendet wird. Diese Grundstruktur besteht aus den in der Funktion `extract_info()` erstellten Dictionarys, welche dann in Listen und weiteren Dictionarys ineinander verschachtelt werden. Dies ermöglicht letztendlich die verschiedenen detaillierten Zoomstufen, die im fertigen Kuchendiagramm zu sehen sind. In der Funktion `create_1800` wird zunächst je Jahrzehnt des 19. Jahrhunderts eine leere Liste angelegt (Z. 57-66), in die dann in den darauf folgenden Zeilen die entsprechenden Einträge nach dem Wert des Schlüssels "objektjahr" einsortiert werden (Z. 70-92). Es wird ein zusätzliches Key-Value-Paar "value": "1" je Objekt hinzugefügt. Der Wert dieses Paares bestimmt, mit welchem Flächeninhalt die einzelnen Objekte am Ende im Diagramm dargestellt werden sollen. Da im finalen Diagramm allen Objekten der gleiche Platz zugesprochen wird, wird für jedes Sammlungsobjekt der Wert "1" vergeben. Dies zeigt auch Abbildung 5.2.

Abbildung 5.2: Ausschnitt aus der Funktion `create_1800()`: Einzelne Objekte werden nach ihrem Entstehungsdatum in Listen gepackt

```

70     for entry in data:
71         if 1800 <= int(entry["objektjahr"]) <= 1809:
72             liste_dict_1800_1809.append({"name": entry["name"], "
value": 1})

```



In den Zeilen 97 bis 126 werden die entstandenen Listen - getreu der Struktur der Beispieldatei aus dem D3.js-Skript - nochmals je Jahrzehnt in Dictionarys gepackt. Diese werden dann, abhängig davon, ob ihre Objekte zur ersten oder zweiten Hälfte des 19. Jahrhunderts zuzuordnen sind, nochmals in zwei Dictionarys geschachtelt - wie in Abbildung 5.3 zu sehen ist. Schließlich entsteht ein großes Dictionary `dict_total`, welches alle Objekte des Jahrhunderts verschachtelt enthält und zurückgegeben wird (Z. 141-145). Dieses wird letztlich von der Funktion `write_file()` verwendet, welche eine JSON-Datei schreibt. Abbildung 5.4 zeigt einen Ausschnitt dieser Datei.

Abbildung 5.3: Ausschnitt aus der Funktion `create_1800()`: Verschachtelung der erstellten Listen und Dictionarys ineinander

```

131     dict_1800_1849 = {
132         "name": "1800-1849",
133         "children": [dict_1800_1809, dict_1810_1819, dict_1820_1829
, dict_1830_1839, dict_1840_1849]}

141     dict_total = {
142         "name": "flare",
143         "children": [dict_1800_1849, dict_1850_1899]
```

Das gesamte Vorgehen orientiert sich sehr stark daran, dass die entstehende JSON-Datei im D3.js-Skript weiterverwendet werden kann, ohne dieses umzuschreiben. Die Herangehensweise mag daher an manchen Stellen aufwendig wirken, dies wurde jedoch in Kauf genommen, um die bereitgestellten Ressourcen weitzunutzen zu können. Lediglich die Größe der resultierenden Grafik wurde im Skript über das Parameter `width` angepasst, um die Visualisierung gut in einer HTML-Datei einbetten zu können. Außerdem wurde natürlich die vom Python-Algorithmus erstellte JSON-Datei in das D3.js-Skript eingebunden. Das vollständige D3.js-Skript samt Endgrafik ist online verfügbar<sup>2</sup>.

Abbildung 5.4: Ausschnitt aus der entstandenen JSON-Datei

```

7  {
8      "name": "1800-1809",
9      "children": [
10         {
11             "name": "ID: 40980",
12             "value": 1
13         },
```

<sup>2</sup>Das D3.js-Skript kann hier abgerufen werden <https://observablehq.com/@s2lascmi/zoomable-sunburst>.

## 5.2 HTML-Datei

Die erstellte Visualisierung soll als Übersichtsgrafik für die SO dienen. Wie dies geschehen könnte, wird in einer eigens erstellten HTML-Datei simuliert.

Die D3.js-Bibliothek bietet die Option, die erstellten Visualisierungen beispielsweise über ein HTML-Snippet zu exportieren. So können die Interaktivität der Grafik und damit ihr wichtigster Faktor erhalten bleiben, was bei einer .jpg-Datei beispielsweise nicht möglich wäre. Das Snippet wird mit dem Element `<iframe>` in einer HTML-Datei eingebettet, die darüber hinaus Link-Verknüpfungen zur SO und dem Landesmuseum Württemberg enthält. Außerdem werden noch kurze Informationen zur Navigation innerhalb der Grafik angeboten, um eine gute Interaktion zu ermöglichen. Ein zentraler Bestandteil der HTML-Datei ist zuletzt auch die direkte Verknüpfung der Visualisierung mit der Schnittstelle über ein Suchfeld, in welches die ID eines Objekts eingegeben werden kann. Die Umsetzung der Verknüpfung ist in Abbildung 5.5 dargestellt. So kann von der Übersichtsgrafik aus direkt auf die in der Schnittstelle bereitgestellten Informationen zugegriffen werden, ohne die SO aktiv aufrufen zu müssen. Nutzende können folglich mit geringstem Aufwand bequem zwischen Übersichtsgrafik und Museumsseite wechseln.

Abbildung 5.5: Umsetzung der Suchleiste im HTML-Dokument

```

52     <form action="https://www.landeshmuseum-stuttgart.de/sammlung/
sammlung-online/dk-details?dk_object_id=">
53     <label>ID des Objekts:
54     <input type="number" name="dk_object_id" min="1" max="
999999">
55     </label>
56     <button>Sammlung Online aufrufen</button>
57 </form>

```

Insgesamt sollten natürlich weitere solcher Übersichten erstellt werden, um die Nutzung der Schnittstelle noch angenehmer zu gestalten und eine Orientierung zu schaffen. Auch Histogramme oder Karten würden sich beispielsweise anbieten, um die in der SO enthaltenen Informationen anschaulich zu visualisieren und den Nutzenden eine gute Übersicht zu bieten. Die Visualisierung für Objekte des 19. Jahrhunderts dient hier lediglich zur Verdeutlichung bzw. Simulation, denn der Fokus der vorliegenden Arbeit liegt auf der Analyse der Schnittstelle und der Weiterverarbeitung der Daten. Deshalb wird auch kein aufwendiges Design für die HTML-Datei mittels CSS-Regeln definiert, sondern lediglich darauf geachtet, dass die Übersichtsgrafik in ihrer Funktionalität zum Tragen kommt. Das HTML-Dokument ist im Anhang der Hausarbeit hinterlegt und kann zusätzlich im GitHub-Repository<sup>3</sup> abgerufen werden.

<sup>3</sup>Die Datei ist im GitHub-Repository zu finden unter [https://raw.githubusercontent.com/s2lascmi/HausarbeitVertiefung/master/venv/HTML\\_Hausarbeit\\_Vertiefung.html](https://raw.githubusercontent.com/s2lascmi/HausarbeitVertiefung/master/venv/HTML_Hausarbeit_Vertiefung.html).

## 6. Evaluation und Fazit

Wie die beiden vorausgehenden Kapitel 4 und 5 zeigen, ist die Schnittstelle Sammlung Online in großen Teilen ein gelungener Online-Zugriff auf die Objekte des Landesmuseums Württemberg. Zudem konnten die Daten, welche aus der Schnittstelle abgegriffen wurden, mit Programmierkenntnissen gut für eine Übersichtsgrafik weiterverwendet werden.

Besonders positiv fällt die Sammlung Online in Bezug auf ihre Nutzungsfreundlichkeit und die gelungene Benutzungsoberfläche auf. Auch Lizenzangaben zu den einzelnen Informationen machen die Weiternutzung der Daten möglich und transparent. Innerhalb der Sammlung sind die Daten eng miteinander verknüpft und machen die Explorationsmöglichkeiten der Schnittstelle somit vielfältig. Die Verwendung der Standards LIDO und JSON ist ein weiterer positiv zu vermerkender Aspekt, in Bezug auf die Verwendung von Normdaten gibt es Optimierungspotenzial. Die Möglichkeit, die Daten zu einer selbst erstellten Teilsammlung abzurufen, ist nicht gegeben, was die Nachnutzung der Informationen immens erschwert. Insgesamt sind bei der Weiterverwendung der bereitgestellten Daten Programmierkenntnisse von Vorteil, denn einige Informationen, wie beispielsweise das Entstehungs- bzw. Bau- oder Findjahr von Objekten, müssen aus anderen Objektangaben herausgefiltert werden.

Der Mangel an Übersichtsgrafiken, die den Nutzenden den Einstieg und die Exploration der Sammlung Online erleichtern sollen, wird im letzten Kapitel der Ausarbeitung aufgegriffen. Mithilfe eines selbst geschriebenen Python-Skripts und unter der Verwendung eines bereits bestehenden Skripts aus der JavaScript-Bibliothek D3.js wurde ohne großen Aufwand ein interaktives Kuchendiagramm aus den Daten der Schnittstelle erstellt. Dieses zeigt an, welches Objekt welchem Jahrzehnt des 19. Jahrhunderts zugeordnet werden kann und ermöglicht eine direkte Verknüpfung zur Sammlung.

Insgesamt kann die Schnittstelle Sammlung Online des Landesmuseums Württemberg trotz Optimierungspotential als Beispiel für eine gelungene Schnittstelle gewertet werden. Sowohl rein an der Sammlung Interessierte als auch Forschende, welche auf die Daten zugreifen möchten, können die Sammlung Online für ihre Zwecke gut und flexibel nutzen.

# Literaturverzeichnis

Creative Commons. (k.D.). *Namensnennung – Weitergabe unter gleichen Bedingungen 4.0 International (CC BY-SA 4.0)*.

<https://creativecommons.org/licenses/by-sa/4.0/deed.de>  
[06.07.2022].

Gesellschaft zur Förderung des Landesmuseums Württemberg e.V. (k.D.). *Jetzt Fördermitglied werden*.

[https://www.landmuseum-stuttgart.de/fileadmin/user\\_upload/0\\_Downloads/LMW\\_Foerdergesellschaft\\_2020.pdf](https://www.landmuseum-stuttgart.de/fileadmin/user_upload/0_Downloads/LMW_Foerdergesellschaft_2020.pdf) [06.07.2022].

Griebel, R., Schäffler, H., Söllner, K. & Frantz, E. (Hrsg.). (2014). *Praxishandbuch Bibliotheksmanagement*. De Gruyter Saur.

<https://doi.org/10.1515/9783110303261>.

Gruber, N. (2022). *Landesmuseum Württemberg: Erlebe es auf Deine Art!*

<https://www.landmuseum-stuttgart.de/museum/ueber-das-museum>  
[06.07.2022].

ICOM International Committee for Documentation. (k.D.). *What is LIDO?*

<https://cidoc.mini.icom.museum/working-groups/lido/lido-overview/about-lido/what-is-lido/> [06.07.2022].

ICOM-CIDOC LIDO Working Group. (2021). *Lido Lightweight Information Describing Objects Version 1.1*.

<https://lido-schema.org/schema/v1.1/lido-v1.1.html> [05.07.2022].

Klingspor, N. (2022). *Sammlung Online*.

<https://www.landmuseum-stuttgart.de/sammlung/sammlung-online>  
[06.07.2022].

Wiechmann, B. (2014). Normdaten. In R. Griebel, H. Schäffler, K. Söllner E. Frantz (Hrsg.), *Praxishandbuch Bibliotheksmanagement. Praxishandbuch Bibliotheksmanagement* (S. 380–391). De Gruyter Saur.

<https://doi.org/10.1515/9783110303261.380>.

# Anhang

# A. Programmcode

Der vollständige Programmcode ist ebenfalls digital abrufbar unter <https://github.com/s2lascmi/HausarbeitVertiefung>.

```
1 import json
2
3
4 def read_file():
5
6     # Datei einlesen
7     filename = "lmwallobjects.json"
8     with open(filename, "r") as file:
9         data = json.load(file)
10    return data
11
12
13 def extract_info(data):
14
15     # JSON-Datei mit allen Objekten der Sammlung Online einlesen
16     # Je Dictionary (also je Objekt) in JSON-Datei einen
17     # Dateneintrag für (Erstellungs-) Jahr und ID hinzufügen
18     i = 0
19     for data[i] in data:
20
21         # Jahr als Dictionary-Eintrag hinzufügen (aus Name des
22         # Objekts extrahieren über letzte vier Zeichen)
23         year = data[i]["objekt_name"]
24         year = year[-4:]
25
26         # Prüfen, ob year auch wirklich Jahreszahl ist oder
27         # nicht, wenn nicht "102983" einfügen
28         if year.isdigit():
29             data[i]["objektjahr"] = year
30         else:
31             data[i]["objektjahr"] = 102983
```

---

```

29
30     # ID extrahieren und für Enddatei als "Name" speichern ,
    um Struktur des vorhandenen D3JS-Skripts zu imitieren
31     # (in Enddatei wird das Sammlungsobjekt über die ID
    referenziert)
32     id = data[i]["objekt_id"]
33     data[i]["name"] = "ID: " + str(id)
34
35     # nicht nötige Key-Value-Paare entfernen
36     data[i].pop("objekt_id")
37     data[i].pop("objekt_name")
38     data[i].pop("objekt_inventarnr")
39     data[i].pop("objekt_erfasst_am")
40     data[i].pop("objekt_beschreibung")
41     data[i].pop("institution_id")
42     data[i].pop("institution_name")
43     data[i].pop("image")
44     data[i].pop("total")
45     i += 1
46
47     # bereinigte Daten zurückgeben , bestehend aus Objektjahr und
    ID
48     return data
49
50
51 def create_1800(data):
52
53     # Erstellt die Gesamtstruktur der JSON-Datei , die dann in
    D3JS-Skript eingelesen wird
54     # Die Struktur besteht aus ineinander geschachtelten
    Dictionarys , welche letztendlich die einzelnen Detailebenen
    des Kuchendiagramms ergeben
55
56     # Leere Listen für Jahrzehnte erstellen , in die die Objekte
    nach Jahr einsortiert werden
57     liste_dict_1800_1809 = []
58     liste_dict_1810_1819 = []
59     liste_dict_1820_1829 = []
60     liste_dict_1830_1839 = []
61     liste_dict_1840_1849 = []
62     liste_dict_1850_1859 = []
63     liste_dict_1860_1869 = []

```

```

64     liste_dict_1870_1879 = []
65     liste_dict_1880_1889 = []
66     liste_dict_1890_1899 = []
67
68     # jeden Dateneintrag zu Liste mit zugehörigem
    Entstehungsdatum hinzufügen
69     # Objekt wird hier schon als Dictionary eingefügt mit Key "
    name" und Value 1, um Struktur des D3JS-Skripts zu imitieren
70     for entry in data:
71         if 1800 <= int(entry["objektjahr"]) <= 1809:
72             liste_dict_1800_1809.append({"name": entry["name"],
    "value": 1})
73         elif 1810 <= int(entry["objektjahr"]) <= 1819:
74             liste_dict_1810_1819.append({"name": entry["name"],
    "value": 1})
75         elif 1820 <= int(entry["objektjahr"]) <= 1829:
76             liste_dict_1820_1829.append({"name": entry["name"],
    "value": 1})
77         elif 1830 <= int(entry["objektjahr"]) <= 1839:
78             liste_dict_1830_1839.append({"name": entry["name"],
    "value": 1})
79         elif 1840 <= int(entry["objektjahr"]) <= 1849:
80             liste_dict_1840_1849.append({"name": entry["name"],
    "value": 1})
81         elif 1850 <= int(entry["objektjahr"]) <= 1859:
82             liste_dict_1850_1859.append({"name": entry["name"],
    "value": 1})
83         elif 1860 <= int(entry["objektjahr"]) <= 1869:
84             liste_dict_1860_1869.append({"name": entry["name"],
    "value": 1})
85         elif 1870 <= int(entry["objektjahr"]) <= 1879:
86             liste_dict_1870_1879.append({"name": entry["name"],
    "value": 1})
87         elif 1880 <= int(entry["objektjahr"]) <= 1889:
88             liste_dict_1880_1889.append({"name": entry["name"],
    "value": 1})
89         elif 1890 <= int(entry["objektjahr"]) <= 1899:
90             liste_dict_1890_1899.append({"name": entry["name"],
    "value": 1})
91         else:
92             print("")
93

```



```

94
95     # Dictionaries dritter Ebene erstellen aus vorher
generierten Listen der Objekte
96     # für Enddarstellung in Kuchendiagramm: zweite,
detailliertere Zoomstufe
97     dict_1800_1809 = {
98         "name": "1800–1809",
99         "children": liste_dict_1800_1809}
100     dict_1810_1819 = {
101         "name": "1810–1819",
102         "children": liste_dict_1810_1819}
103     dict_1820_1829 = {
104         "name": "1820–1829",
105         "children": liste_dict_1820_1829}
106     dict_1830_1839 = {
107         "name": "1830–1839",
108         "children": liste_dict_1830_1839}
109     dict_1840_1849 = {
110         "name": "1840–1849",
111         "children": liste_dict_1840_1849}
112     dict_1850_1859 = {
113         "name": "1850–1859",
114         "children": liste_dict_1850_1859}
115     dict_1860_1869 = {
116         "name": "1860–1869",
117         "children": liste_dict_1860_1869}
118     dict_1870_1879 = {
119         "name": "1870–1879",
120         "children": liste_dict_1870_1879}
121     dict_1880_1889 = {
122         "name": "1880–1889",
123         "children": liste_dict_1880_1889}
124     dict_1890_1899 = {
125         "name": "1890–1899",
126         "children": liste_dict_1890_1899}
127
128
129     # Dictionaries zweiter Ebene erstellen, indem die vorherigen
Dictionaries zusammengefasst werden
130     # für Enddarstellung in Kuchendiagramm: erste Zoomstufe
131     dict_1800_1849 = {
132         "name": "1800–1849",

```

```

133         "children": [dict_1800_1809, dict_1810_1819,
dict_1820_1829, dict_1830_1839, dict_1840_1849]]
134     dict_1850_1899 = {
135         "name": "1850-1899",
136         "children": [dict_1850_1859, dict_1860_1869,
dict_1870_1879, dict_1880_1889, dict_1890_1899]]
137
138
139     # oberste Ebene erstellen, indem die vorherigen Dictionaries
zusammengefasst werde
140     # für Enddarstellung in Kuchendiagramm: Anfangsansicht
141     dict_total = {
142         "name": "flare",
143         "children": [dict_1800_1849, dict_1850_1899]
144     }
145     return dict_total
146
147
148 def write_file(data):
149
150     # JSON-Datei schreiben, die dann in D3JS-Skript verwendet
wird
151     file_name = "objects-1800-1899.json"
152     file = open(file_name, "w")
153     json.dump(data, file)
154     file.close()
155
156
157 def main():
158     raw_data = read_file()
159     cleaned_data = extract_info(raw_data)
160     final_data = create_1800(cleaned_data)
161     write_file(final_data)
162
163
164 if __name__ == "__main__":
165     main()

```

## B. HTML-Dokument

Der vollständige HTML-Dokument ist ebenfalls digital abrufbar unter <https://github.com/s2lascmi/HausarbeitVertiefung>.

```
1 <!DOCTYPE html>
2 <html lang="de">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>Übersichtsgrafik Sammlung Online , Objekte des 19.
    Jahrhunderts</title>
7
8   <style>
9     body {
10       font-family: sans-serif;
11       text-align: center;
12       padding-left: 50px;
13       padding-right: 50px;
14     }
15
16     button {
17       background-color: #CAFF70;
18     }
19
20     .Erklärung {
21       padding-left: 200px;
22       padding-right: 200px;
23     }
24
25     .Suchfeld {
26       background-color: beige;
27       padding: 20px 0px 20px 0px;
28     }
29   </style>
30 </head>
```

```

31
32 <body>
33   <div class="Erklärung">
34     <h1>Objekte des 19. Jahrhunderts</h1>
35     <h2>Übersichtsgrafik zu Objekten der <a
36       href="https://www.landesmuseum-stuttgart.de/sammlung/
sammlung-online">Sammlung
37       Online</a> des <a href="https://www.landesmuseum-
stuttgart.de">Landesmuseums Württemberg</a></h2>
38     <p>Klicken Sie auf einen der Zeitbereiche im Diagramm, um
auf eine detailliertere Ebene der Visualisierung zu
39     gelangen. Klicken Sie in die Mitte des Diagramms, um auf
die nächsthöhere Ebene zurück zu springen.</p>
40     <p>Die Flächenverteilung auf den einzelnen Ebene des
Kreisdiagramms stellt den prozentualen Anteil der Objekte aus
41     der
42     Sammlung für den dargestellten Zeitraum dar. Die
Ausgangsvisualisierung zeigt also beispielsweise, dass aus
dem
43     gesamten 19. Jahrhundert ca. zwei Drittel der in der
Sammlung enthaltenen Objekte zwischen 1800 und 1849
entstanden und ca. ein Drittel
44     zwischen
45     1850 und 1899.
46   </p>
47
48   <div class="Suchfeld">
49     <p>Wenn Sie ein Objekt der Sammlung genauer recherchieren
möchten, geben Sie die Ziffern der Objekt-ID aus der Grafik
in das
50     unten
51     stehende Suchfeld ein. Sie werden direkt weitergeleitet
.</p>
52     <form action="https://www.landesmuseum-stuttgart.de/
sammlung/sammlung-online/dk-details?dk_object_id=">
53       <label>ID des Objekts:
54       <input type="number" name="dk_object_id" min="1" max="
999999">
55       </label>
56       <button>Sammlung Online aufrufen</button>
57     </form>
58     <p>Alternativ können Sie diesen <a

```

```

59         href="https://www.landmuseum-stuttgart.de/sammlung/
sammlung-online/dk-details?dk-object_id=">Link</a>
60     öffnen.
61     Auf der sich öffnenden Webseite hängen Sie dann einfach
die im Diagramm angezeigte Objekt-ID an die URL an.</p>
62     </div>
63 </div>
64
65 <<iframe width="70%" height="1010" frameborder="0"
66 src="https://observablehq.com/embed/@s2lascmi/zoomable-
sunburst?cells=chart"></iframe>
67
68 <div class="Erklärung">
69     <p>Die Grafik wurde erstellt auf Basis des Skripts <a
70         href="https://observablehq.com/@d3/zoomable-sunburst">
Zoomable Sunburst</a>
71     der <a href="https://d3js.org">D3JS-Bibliothek</a>. Das
leicht angepasste Skript zu dieser Grafik kann
72     <a href="https://observablehq.com/@s2lascmi/zoomable-
sunburst">hier</a> abgerufen werden.
73     </p>
74     <br>
75     <p>Diese Grafik wird unter der <a href="https://
creativecommons.org/licenses/by/2.0/de/">Creative Commons
Lizenz CC
76     BY</a> bereitgestellt.</p>
77 </div>
78 </body>
79
80 </html>

```

# Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich diese Hausarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und die aus fremden Quellen direkt oder indirekt übernommenen Gedanken als solche kenntlich gemacht habe. Die Arbeit habe ich bisher keinem anderen Prüfungsamt in gleicher oder vergleichbarer Form vorgelegt. Sie wurde bisher auch nicht veröffentlicht.

L. Schmitt

Ruppach-Goldhausen, den 28. Juli 2022