Auswahl und Beschreibung der initialen technischen Workflows und Policies für den Data LifeCycle (R 2.3.1)

Version 4 – 21.07.2015
Cluster 2.3
Verantwortlicher Partner HKI

DARIAH-DE
Aufbau von Forschungsinfrastrukturen für die e-Humanities

Dieses Forschungs- und Entwicklungsprojekt wird / wurde mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF), Förderkennzeichen 01UG1110A bis N, gefördert und vom Projektträger im Deutschen Zentrum für Luft- und Raumfahrt (PT-DLR) betreut.

GEFÖRDERT VOM

Bundesministerium für Bildung und Forschung

Projekt: DARIAH-DE: Aufbau von Forschungsinfrastrukturen für die e-Humanities
BMBF Förderkennzeichen: 01UG1110A bis N

Laufzeit: März 2011 bis Februar 2016
Dokumentstatus: Final
Verfügbarkeit: Öffentlich
Autoren: Johanna Puhl, HKI
Stefan E. Funk, SUB
Danah Tonne, KIT
Martin Haase, DAASI
Tibor Kalman, GWDG

Revisionsverlauf:

Datum	Autor	Kommentare
07.04.2014	Johanna Puhl	First Draft
28.04.2014	Johanna Puhl	Ausbau
09.05.2014	Johanna Puhl	Ausbau
14.05.2014	Stefan E. Funk	Kommentare
19.05.2014	Danah Tonne	Überarbeitung Bitstream Preservation
21.05.2014	Martin Haase	AAI-Text
26.05.2014	Tibor Kalman	Überarbeitung PID
02.06.2014	Stefan E. Funk	Überarbeitung Repository
16.10.2014	Johanna Puhl	Überarbeitung Einleitung, Kapitel 2 und 6
29.10.2014	Johanna Puhl	Überarbeitung insgesamt
11.11.2014	Johanna Puhl	Ergänzung Kapitel Prüfsummen
29.05.2015	Johanna Puhl	Überarbeitung & Abschluss gemäß neustem RDLC Stand
01.07.2015	Stefan Funk	Überarbeitung Repository

Inhalt

1. Einleitung

Aufbauend auf schon existierenden Ergebnissen aus der ersten Projektphase sollen in diesem Report und in enger Zusammenarbeit zwischen Cluster 2 und 3 sowie auf Basis der Ergebnisse der DARIAH AG Research Data Lifecycle Bestandteile und Policies eines Workflows für den Research Data LifeCycle beschrieben werden, welche in der zweiten Projektphase als generische Basis realisiert werden können.

Im weiteren Verlauf der zweiten Förderphase können darauf aufbauend dann Erweiterungen und alternative Lebenszyklen speziell für die Bedürfnisse von einzelnen oder Gruppen von Disziplinen ausgearbeitet werden.

Im ersten Teil dieses Reports werden häufig zitierte und in DARIAH diskutierte Ausprägungen eines "Research Data Lifecycle" vorgestellt, welche im Rahmen der AG "Research Data Lifecycle", zu einer möglichst generischen Definition führen sollen, so dass sich ein erster generischer Forschungsdatenzyklus in der DARIAH Infrastruktur umsetzen lässt.

Im zweiten Teil beschreiben wir, welche schon implementierten Komponenten hierfür sinnvoll verwendet und ggf. modularisiert werden sollten und welche ggf. noch definiert und implementiert werden müssen.

Im letzten Teil dieses Reports soll – aufbauend auf den vorher beschriebenen Grundbausteinen – der DARIAH Forschungsdatenzyklus vorgestellt werden und ggf. fehlende (d.h. noch nicht definierte oder implementierte) Komponenten identifiziert werden. Daneben soll an dieser Stelle diskutiert werden, welche Beschreibungstechnologie zur Implementierung angestrebt wird.

In diesem Report folgen wir der Unterscheidung zwischen den Begrifflichkeiten *Tool* und *Service*[1]:

> *"The term "service" refers to a set of related software functionalities that can be accessed via application programming interface (API). A "tool" covers one specific scholarly activity, e.g. XML editor and it requires user interaction."*

Kapitel 2 enthält alle Ergebnisse aus der AG Research Data LifeCycle, Kapitel 3 beschreibt bereits existierende Komponenten aus der DARIAH Infrastruktur, die einen Workflow im Sinne des Research Data Lifecycle unterstützen,

Kapitel 4 widmet sich explizit dem Problemfeld der digitalen Langzeitarchivierung, einem Gebiet, das zu einem großen Teil automatisierbare Funktionen in einem Research Data LifeCycle abdecken könnte. Kapitel 5 reflektiert mögliche Technologien zur Implementation von Workflows im DARIAH Repository.

[1] Gemäß der Definition in Meilenstein 1.3.4.1 aus DARIAH I: "Service Interoperability – Multi-Modal Interoperability Concept".

2. Research Data Lifecycle – Definition

An dieser Stelle sollen zwei Ziele erreicht werden:

- Einerseits soll eine (keinesfalls vollständige) Übersicht bestehender Konzepte eines geisteswissenschaftlichen Forschungsdatenzyklus erstellt werden und
- soll das darauf aufbauende Ergebnis der Arbeitsgruppe Research Data Lifecycle in DARIAH-DE vorgestellt und als Basis zu Entwicklung eines generischen ersten Workflows in Cluster 2 und 3 verwendet werden.

2.1 Diversität der Konzepte

Viele Bemühungen, die Lebensdauer und Nachnutzung von (geisteswissenschaftlichen) Forschungsdaten möglichst allgemeingültig zu definieren und darzustellen, haben in der Vergangenheit zu ebenso vielen Ergebnissen geführt, aus denen hier ein kurzer Ausschnitt präsentiert werden soll[2]:

Folgende Grafik wurde im Rahmen der zweiten Förderphase von DARIAH in Cluster 2 zur Veranschaulichung eines Forschungsdatenzyklus verwendet.[3]

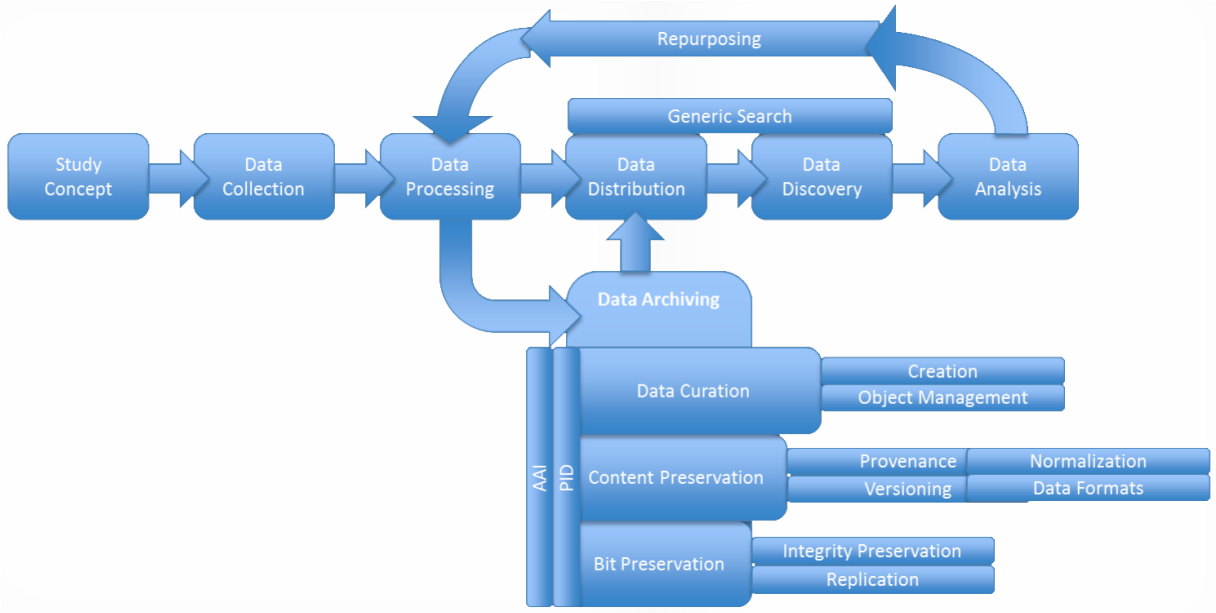

Abb 1. Erweiterter Research Data LifeCycle aus der DDI Allianz

Die aus der DDI Allianz stammende Grafik, welche ursprünglich aus den Sozialwissenschaften stammt[4],

[2] gemäß der Präsentation von Herr Prof. Dr. Thaller auf dem Konsortialtreffen DARIAH II in Darmstadt, 20 März 2014

[3] DDI Structural Reform Group. „DDI Version 3.0 Conceptual Model." DDI Alliance. 2004; angepasst von Daniel Kurzawe und Rainer Stotzka, 13. Feb. 2013.

[4] Vgl. http://opendatafoundation.org/ddi/srg/Papers/DDIModel_v_4.pdf. S.8. Aktuelle Grafik ist in der Version 3.2 stärker vereinfacht unter: http://www.ddialliance.org/sites/default/files/ResearchCycle.gif

wurde für DARIAH vor allem um detailliertere Aufgaben im Feld der (Langzeit-)Archivierung ergänzt, damit eine vollständige Übersicht von erwünschten Merkmalen erzielt wird.

Andere Cluster in DARIAH-DE verwenden hingegen folgendes Schaubild für einen Forschungsdatenzyklus:

Abb 2. Research Data LifeCycle für historische Forschungsdaten

Diese stammt ursprünglich von Boonstra, Breure und Doorn (2004)[5] und bezieht sich auf historische Forschungsdaten.

Bei Boonstra et al. wird explizit darauf hingewiesen, dass im historischen Forschungsprozess nicht alle Schritte in der angegebenen Reihenfolge durchlaufen werden und unter bestimmten Bedingungen sogar ausgelassen werden können (p.21).

Eine weitere Perspektive von Cluster 5 und 6 auf den Lebenszyklus von Forschungsdaten kommt von John Unsworth 2000.[6] Hier wird eine Liste von Tätigkeiten vorgestellt, die gemäß Unsworth den geisteswissenschaftlichen Forschungsprozess beschreibt. Obwohl der Autor explizit auf den auf den nicht erschöpfenden Charakter der aufgeführten Tätigkeiten hinweist und um Ergänzungen bittet, besitzt diese Liste mittlerweile normative Wirkung.

Die von Unsworth genannten Tätigkeiten sind:

- "Discovering
- Annotating
- Comparing
- Referring
- Sampling

[5] Vgl. http://www.dans.knaw.nl/sites/default/files/file/publicaties/Past-present.pdf, S.22.
[6] Vgl. http://people.brandeis.edu/~unsworth/Kings.5-00/primitives.html

- Illustrating
- Representing"

Legt man diese Begriffe auf den Zyklus von Doorn et al, ergibt dies folgendes Bild[7]:

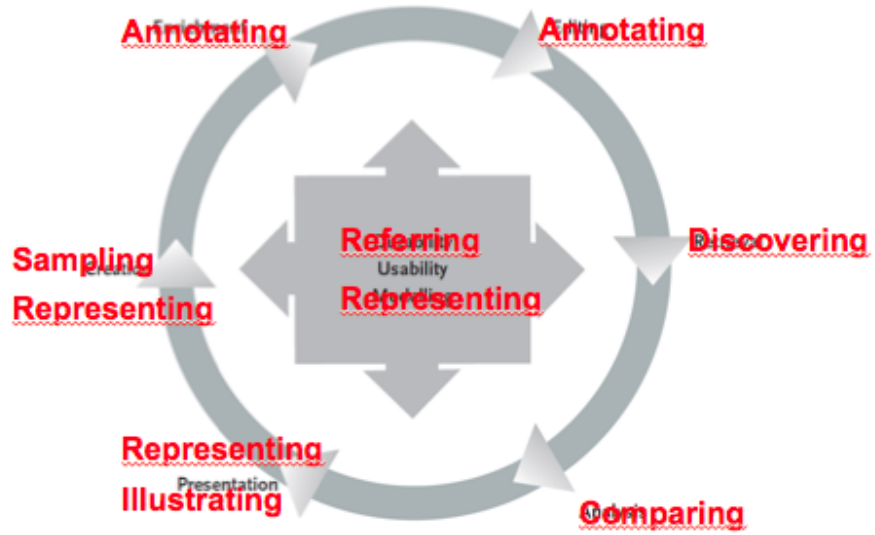

Abb 3. Abbildung der Unsworth'schen Primitives auf den Research Data LifeCycle von Boonstra, Breure und Doorn (zuerst vorgestellt von Thaller, Darmstadt 2014)

Legt man die Unsworth'schen Primitives auf den erweiterten DDI Lifecycle ergibt sich folgendes Bild:

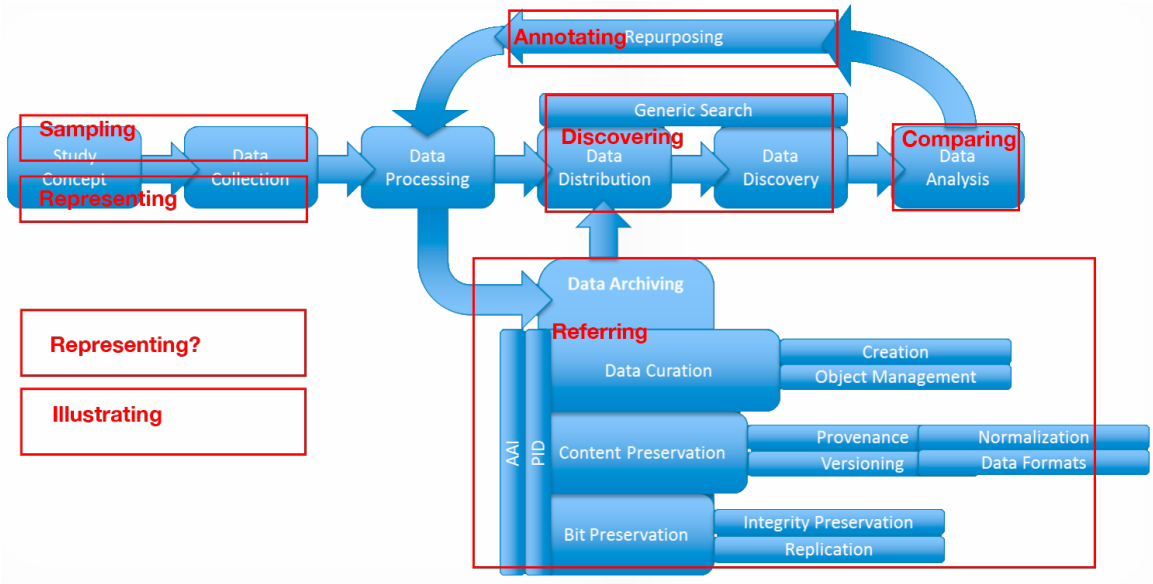

Abb 4. Abbildung der Unsworth'schen Primitives auf den Research Data LifeCycle der DDI (Thaller in Darmstadt, Teil II).

[7] Vgl. Prof. Dr. Manfred Thaller beim Dariah Konsortialtreffen, 20.03.2014

Auffällig ist hier, dass, sowohl "Representing" als auch "Illustrating" keine befriedigenden Entsprechungen finden und nur eventuell noch im Kontext des Feldes "Data Distribution" Sinn ergeben.

Legt man hingegen die Doorn'schen Begrifflichkeiten auf den DDI Lifecycle, entsteht die folgende Abbildung:

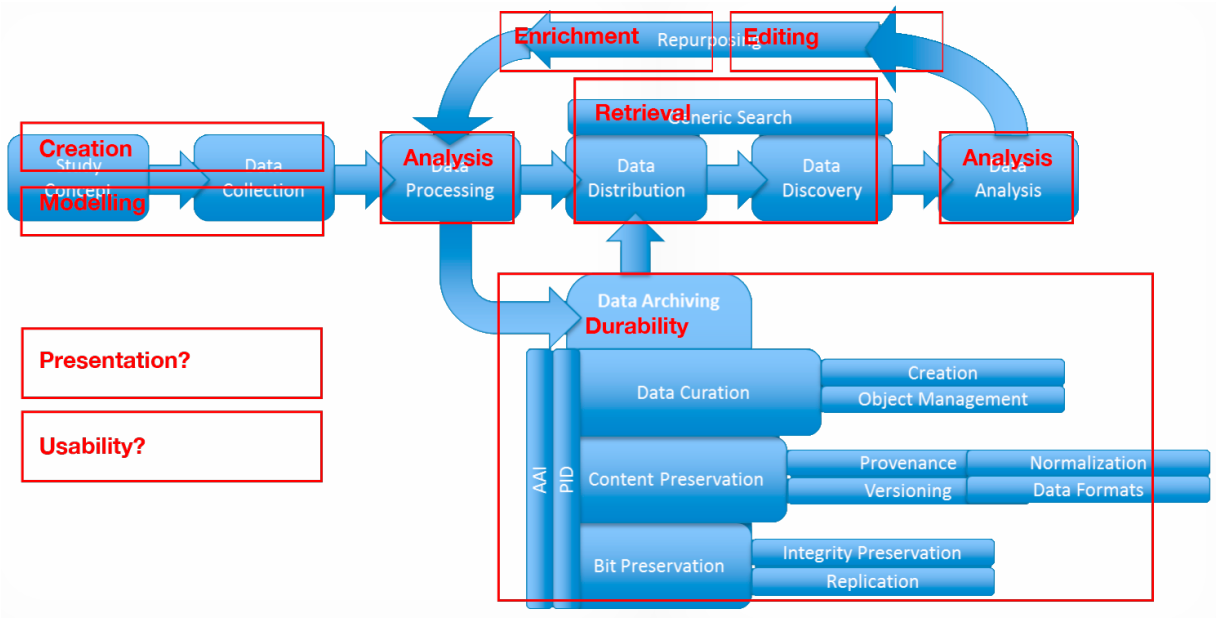

Abb 5. Abbildung der Boonstra, Breure und Doorn Aktivitäten auf den Research Data LifeCycle der DDI Initiative (Thaller in Darmstadt, Teil III)

Hier wiederum sind die Begriffe "Presentation" und "Usability" verwaist.

Aus den hier aufgeführten Beispielen wird deutlich, dass es

- keinen allgemeingültigen Begriff eines Forschungsdatenzyklus gibt,
- dass die Ansätze in Ihrer Granularität stark variieren und dass
- die einzelnen Ausprägungen entscheidend von der Fachdisziplin geprägt werden, aus der die jeweilige Überlegung stammt.

Im Rahmen von DARIAH-DE bedeuten diese Ergebnisse, dass es einerseits notwendig ist, möglichst alle geistes- und sozialwissenschaftlichen Fachdisziplinen in die Ausarbeitung miteinzubeziehen und aus deren Verständnis und Anforderungen dann aber andererseits ein möglichst *generischer* Forschungsdatenzyklus als expliziter "DARIAH Research Lifecycle" zu destillieren ist.

2.2 Ergebnisse der AG "Research Data LifeCycle"

Im Rahmen der zweiten Projektphase wurde die Arbeitsgruppe *Research Data LifeCycle* konstituiert. Durch Überprüfung von Gemeinsamkeiten und Unterschieden in der Sichtweise auf Konzepte und Bestandteile geisteswissenschaftlicher Forschung, kann in dieser AG ein Referenzmodell mit essentiellen

Bestandteilen eines Research Data LifeCycles erarbeitet werden und weiterhin ein Anforderungskatalog für weitere Funktionalitäten, die nur in manchen Disziplinen erwünscht werden, entwickelt werden.

Basierend auf den schon erfolgten Arbeiten zu einem Workflow im DARIAH Repository (siehe folgendes Kapitel) und der Arbeit aus der AG sollte dabei in einem Annäherungsprozess die Lücke zwischen beiden Ansätzen geschlossen werden.

Folgende Aufgaben wurden im Rahmen des ersten AG Research Data Lifecycle Treffens identifiziert:

- Der Bedarf nach einer Definition des Terms "Forschungsdaten" auch im Unterschied zum geisteswissenschaftlichen Sammlungsbegriff
- Die Definition der Dateitypen / Objektklassen, welche in der Implementation des Research Data Lifecycle berücksichtigt werden sollen
- Die Definition der Begriffe "Annotation" (im Unterschied zu Metadaten), “Tools” und “Methoden” – auch und gerade in Abgrenzung voneinander
- Die Definition weiterer Begriffe und Funktionen, die durch einen RDLC erfüllt werden sollen, wie "Curation" im Unterschied "Content Preservation", Rollen- und Rechtemanagement, Identifiervergabe,
- Eine erste LifeCycle Definition

Durch die einerseits möglichst präzise aber andererseit auch möglichst generische für verschiedene Geisteswissenschaften gültige Definition von Begriffen, Funktionen und Abläufen konnte so ein Referenzmodell eines Research Data LifeCycles für die digitalen Geisteswissenschaften erstellt werden.

Auch eine grafische Darstellung von Aktivitäten im Kontext eines zyklischen Ablaufs konnte so generiert werden:

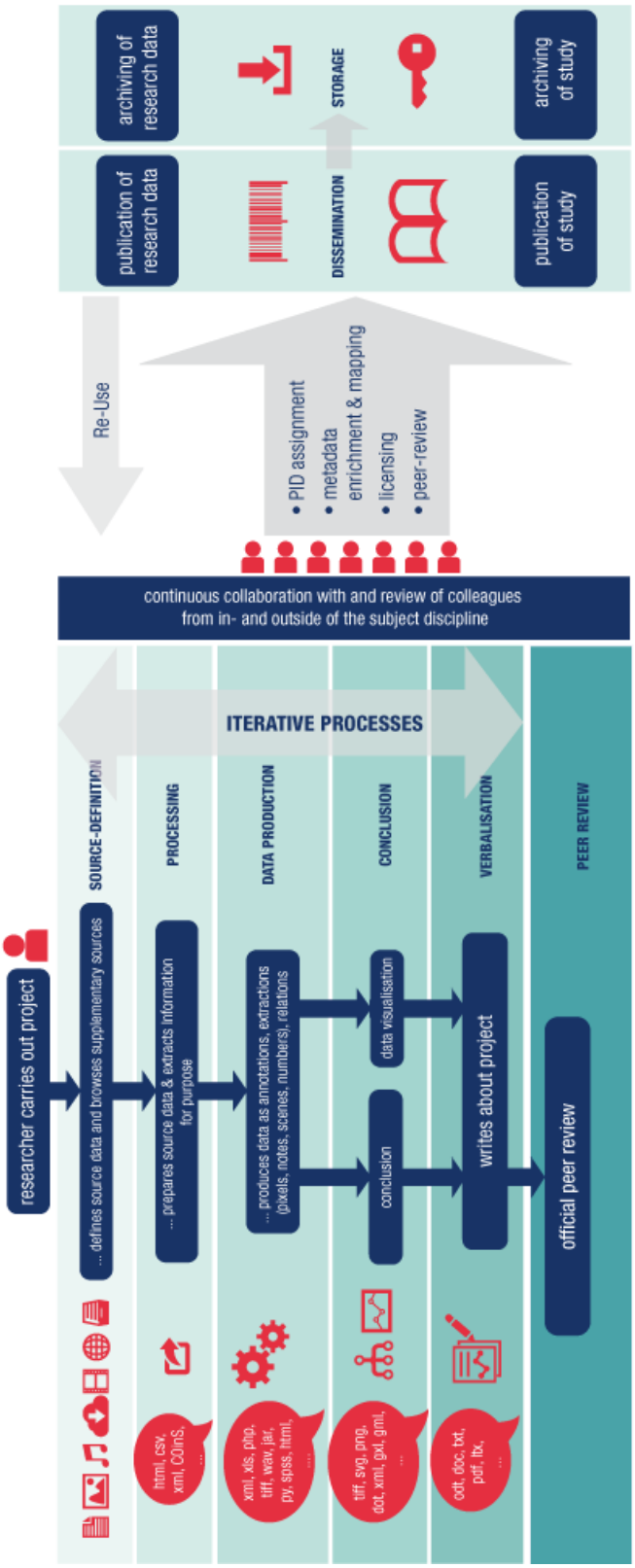

Abb 6. Darstellung eines geisteswissenschaftlichen Research Data LifeCycle in DARIAH-DE

Die folgende Tabelle stellt die Funktionen des Research Data LifeCycle gemäß ihrer Reihenfolge und aus:

Schritt	**Aktivität**	**Formalisierbar Ja/Nein/Vielleicht**	**Notiz**
1	Formulierung Forschungsfrage, Benennung von Methoden	Vielleicht	Anhand einer Basis-Fragestellung lassen sich ggf. anschließende Tätigkeiten oder verwendbare Komponenten formalisieren und maschinell identifizieren.
2	Auswahl Forschungsdaten	Vielleicht	Die Wahl der Forschungsdaten ist eng an die Formalisierung der Forschungsfrage geknüpft: Wenn die Forschungsfrage maschinell auswertbar formalisiert werden kann, so können auch die zu verwendenden Forschungsdaten formell beschrieben werden.
3	Vorbereitung / Verwendung von Tools	Vielleicht	Ist ebenso an die Forschungsfrage und an die Homogenität der vorgefundenen Forschungsdaten geknüpft.
4	Generierung von (Zwischen-) Ergebnissen/ Verwendung von Tools	Ja	Wenn alle Vorbereitungen getroffen und spezifiziert sind: Ja
5	Visualisierung	Ja	"
6	Beschreibung der veröffentlichungs-würdigen Ergebnisse und Erkenntnisse	Nein	Kann sehr begrenzt formalisiert aber niemals maschinell ausführbar beschrieben werden.
7	Kuration / Vorbereitung der Archivierung	Ja	Im Rahmen einer einfachen Langzeitarchivierungsstrategie (Bitstream-Preservation, ggf. Metadaten-Extraktion und Migration) lässt sich formalisiert ein “Preservation Plan” erstellen.

Dabei wurden die folgenden Funktionen:

- Identifiervergabe
- Metadatenanreicherung /-abgleich
- Lizensierung
- Publikation
- Langzeitarchivierung
- Peer-Review

...als iterativ gekennzeichnet, da sie nach jedem Schritt erneut angewendet oder Ihre Anwendung zumindest geprüft werden muss.

Daneben bedürfen einige Schlüsselkonzepte eines DARIAH-Forschungsdatenzyklus der genaueren Spezifikation, welche aktuell noch nicht erfolgt ist:

So sollten sowohl der Begriff der Kuration als auch die in dieser Funktion verarbeitbaren Dateiformate und Objektklassen definiert werden, damit diese technisch implementiert werden können. Auch auf dem Feld der Metadatenvergabe und -normalisierung müssen Entscheidungen zur Nutzung vorhandener Standards getroffen werden, bzw. Anpassungen in Richtung eines eigenen Datenmodells statt finden[8].

Das folgende Kapitel listet die in DARIAH schon existierenden Komponenten auf, um sodann aus diesen noch fehlende Features abzuleiten.

3. Existierende Komponenten

Aus einer Analyse aus der ersten DARIAH-DE Förderphase ist bekannt, dass Geisteswissenschaftler vornehmlich mit heterogenen, aber nicht unbedingt speicherintensiven Dateien unterschiedlichen Medientyps, wie beispielsweise Text, Bild- oder Audiodateien, umgehen und dass daraus der Bedarf nach eindeutigen Identifiern zur Wiederauffindbarkeit und Zitierbarkeit sowie geeigneten Metadaten zur Annotation hervor geht[9].

Ebenso ist der Bedarf nach der langfristigen Verfügbarkeit, d.h. Erhaltung des Bistreams aber auch dessen Nachnutzbarkeit mit aktueller Software explizit Gegenstand der erfolgreichen Implementation eines Forschungsdatenzyklus.

Die folgende Auflistung gibt eine Übersicht über alle aktuell in DARIAH-DE verfügbaren Dienste, die zur Realisierung des diskutierten Forschungsdatenzyklus in einem Workflow eingebunden werden sollten:

[8] Zum aktuellen Stand der Empfehlungen siehe https://dev2.dariah.eu/wiki/pages/viewpage.action?pageId=38080370

[9] Vgl: https://dev2.dariah.eu/wiki/display/DARIAHDE/Langzeitarchivierung

3.1 Das Repository

Zur Unterstützung der beiden Infrastrukturprojekte TextGrid und DARIAH existiert bereits ein Kernsystem mit angeschlossener Infrastruktur. Das Kernsystem firmiert unter der Bezeichnung *TextGrid Repository* bzw. *DARIAH Repository*. Inhaltlich baut das eine auf dem anderen auf, d.h. die in TextGrid initial aufgebaute Infrastruktur wurde und wird in DARIAH weiterentwickelt und nachgenutzt, um Synergien zu schaffen und gemeinsame Anforderungen für beide Projekte erfüllen zu können.

Die folgende Grafik gibt einen Überblick über das TextGrid Repository und die darin genutzten Features[10]:

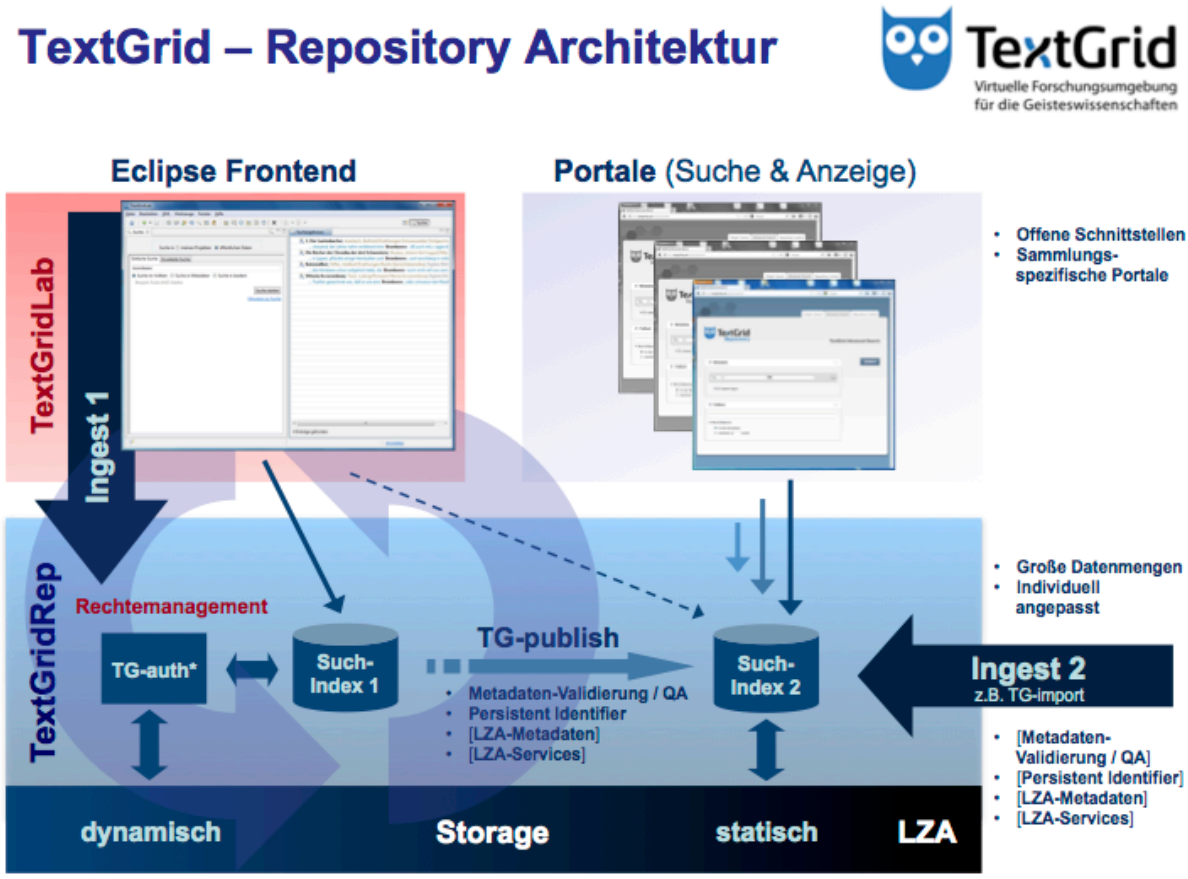

Abb. 7: Abläufe und Funktionen im TextGrid Repository

Neben der Idee, in DARIAH generisch neue Features im Repository anzubieten, bestehen Pläne, einige eher inhaltliche Features des TextGrid Repositories in DARIAH nachzunutzen, hierzu zählt insbesondere das Kollektionskonzept, welches es ermöglicht einzelne Objekte zu so genannten "Kollektionen" zu aggregieren also via Referenzen als "zusammengehörig" auszuzeichnen.

[10] Vgl. https://www.textgrid.de/fileadmin/berichte-2/report-1-2-2.pdf, S. 6

3.1.1. TextGrid

Die in TextGrid realisierte Infrastruktur nutzt gemäß den zu unterstützenden Disziplinen einige Features, die sich nicht 1:1 auf DARIAH übertragen lassen. Insbesondere ist hier das TextGridLab als Einstieg in die Virtuelle Forschungsumgebung (Plattform zum Arbeiten und späteren Publizieren) zu nennen. Weiterhin wird in TextGrid ein sehr spezifischer Objektbegriff inklusive dazugehörigem Metadatenschema verwendet, der sich nicht direkt auf alle Forschungsdaten aus dem Bereich der Digital Humanities, wie sie in DARIAH verwendet werden, übertragen lässt. Auch Identifier werden ggf. anders gebildet und abgelegt als im Schwesterprojekt.

Das Verständnis von aggregierten Objekten, oder "Kollektionen" aus TextGrid ist hingegen ein recht generischer: In TextGrid wird ein Objekt als eine Datei PLUS einem Metadatum gemäß TextGrid Metadata Schema[11] verstanden, dabei kann ein Objekt auch als "Aggregation" zu einem oder mehreren weiteren Objekten verstanden werden, welches mittels Metadatum anzuzeigen ist. Das TextGrid Metadata Schema unterstützt sowohl eigene TextGrid URIs als auch PIDs und weitere Identifier.

3.1.2. DARIAH-DE

Die Gesamtansicht der DARIAH-DE und TextGrid Repositories und die einzelnen Komponenten werden in folgendem Schaubild wieder gegeben:

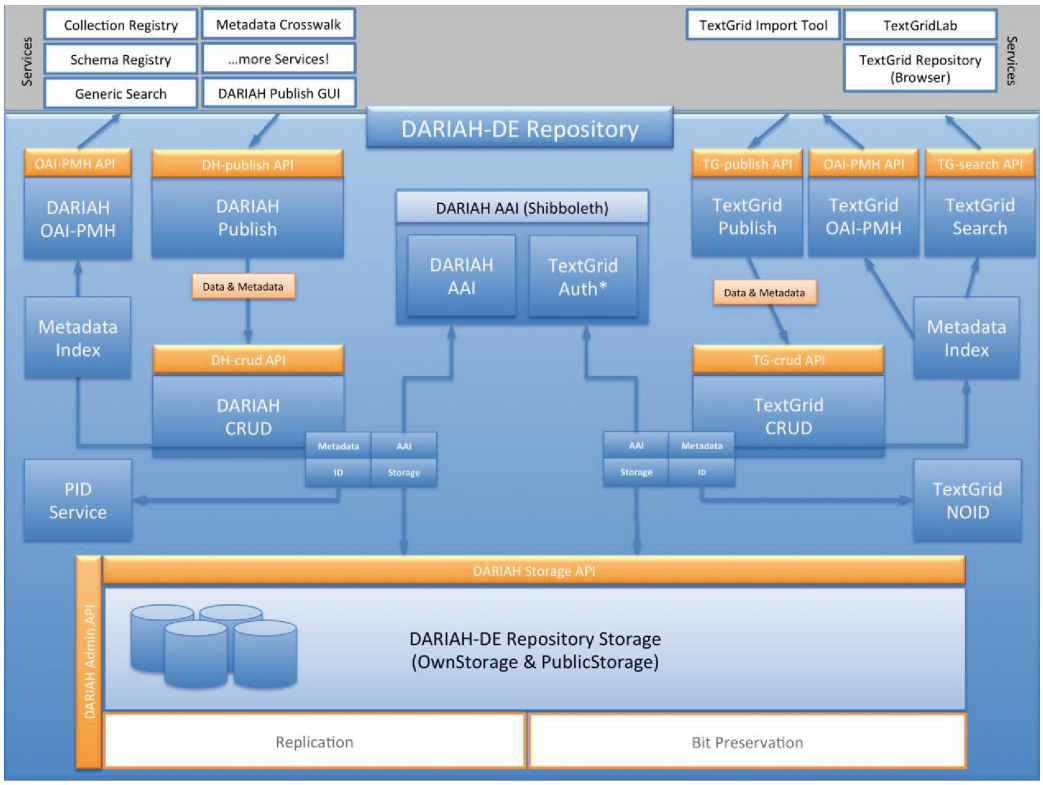

Abb 8. Das DARIAH-DE Repository und das TextGrid Repository – Gemeinsamkeiten und Unterschiede.[12]

[11] Vgl. TextGrid Metadatenschema. http://textgridlab.org/schema/textgrid-metadata_2010.xsd

[12] Vgl. https://de.dariah.eu/der-dariah-ansatz

Interessant ist diese Grafik insbesondere, weil sich aus Ihr ein exemplarischer Workflow der Daten durch das (DARIAH-) Repository nachvollziehen lässt (der Workflow der Daten durch das DARIAH-DE-Repositorium ist nochmal im Einzelnen in Schaubild dargestellt).

Alle genannten Services werden durch den DARIAH AAI (Authorization and Authentication Infrastructure) -Service unterstützt und werden dementsprechend registriert. Die Reihenfolge der verwendeten Services im Workflow ist dabei die Folgende:

- Publish Web-Interface
    - Datenauswahl via Webinterface
    - Beschreibung mit DC-Metadaten
    - Angabe von Abhängigkeiten (z.B. Sub-Kollektionen)
    - Weitergabe an Publish Service
- DARIAH-publish Service
    - Metadatenvalidierung
    - Aktualisierung von Referenzen/Dateipfaden zu mitgelieferten Daten
    - Weitergabe an DARIAH CRUD
- DARIAH-CRUD Service
    - *Steht für* (CREATE, RETRIEVE, UPDATE and DELETE)
    - (Statische) Speicherung in DARIAH Storage
    - Metadatenspeicherung in Indexdatenbank (für OAI-PMH)
    - Erzeugung eines PID
- OAI-PMH Service
    - OAI-PMH Protokoll zur öffentlichen Abfrage der Metadaten
    - Antworten via ElasticSearch-Index.
    - Nachnutzung als personalisierte Suchfunktion auf den Metadaten als "Generische Suche"

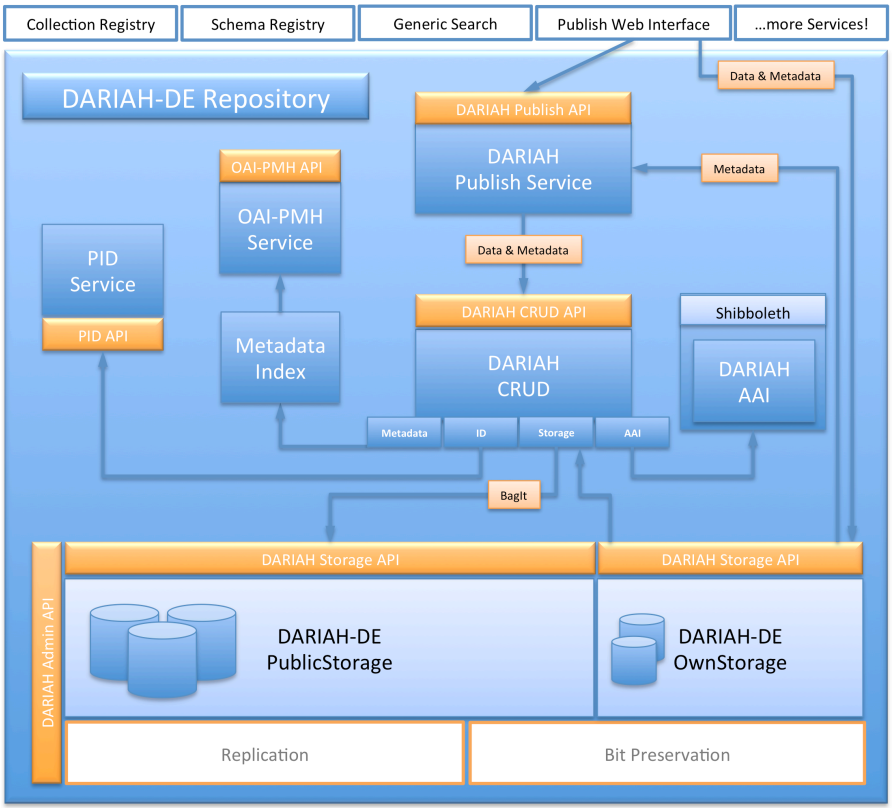

Abb 8.1 Der Prototyp des DARIAH-DE Repositorium

Die soeben aufgelisteten Services werden in Kapitel 3.2 im Detail erklärt.

Nachfolgend einige Erläuterungen:

Im DARIAH Repository sollen – genau wie bei TextGrid im statischen Bereich – EPIC PIDs als Identifikatoren für alle Objekte vergeben werden, es wird lediglich auf die interne ID verzichtet, die in TextGrid als ID für den dynamischen Speicherbereich genutzt wird. Die eingespielten Objekte sollen mittels DublinCore (DC-Simple) mit Metadaten versehen werden. Wie schon im obigen Workflow erwähnt, soll hier der schon aus TextGrid bekannte CRUD Service unter der Bezeichnung DARIAH-CRUD Service genutzt werden, der hierfür angepasst werden muss. Ziel ist eine gemeinsame Codebasis mit an DARIAH und TextGrid angepassten Modulen für z. B. PID-, Metadaten-, AAI- und Storage-Anbindung.

Daneben wird ein Web-Interface, eine OAI-PMH Schnittstelle und eine eigene Publish-API samt Service angeboten. Das Web-Interface ist dabei zusammen mit der Publish-Funktion dafür zuständig, die zu publizierenden Objekte auszuwählen und mit Metadaten anzureichern. Diese werden später vom Publish-Service via DARIAH-crud eingespielt und dort mit persistenten Identifikatoren zur langfristigen Referenzierbarkeit im Repository versehen.

Die in beiden Repositories erwähnten APIs, Features, Komponenten bedürfen ggf. einer genaueren Beschreibung, welche in dem nun folgenden Kapitel erfolgt.

Die zu einer Kollektion zusammengefassten und publizierten Objekte können schließlich – unter Angabe der OAI-PMH URL, unter dem diese Kollektion zu finden ist – als eine Kollektion in der DARIAH-DE

Collection Registry registriert werden, und somit sind die Daten der gesamten Kollektion mit der DARIAH-DE Generischen Suche nachgewiesen. Ein Schaubild das DARIAH-DE Repositoriums

Eine detaillierte Beschreibung des DARIAH-DE Repositoriums-Prototypen[13] findet sich im öffentlichen DARIAH-DE-Wiki.

3.2 Die einzelnen Komponenten des DARIAH Repositories

3.2.1 CRUD

Das TextGrid Repository beinhaltet einen Service namens CRUD (CREATE, RETRIEVE, UPDATE and DELETE), welche basierend auf dem TextGrid Metadatenschema die Möglichkeit zur Erstellung, Zugriff, Aktualisierung und Löschung von Objekten (hier: TextGrid Objects) im Repository bietet.

Die genannten Services von TextGrid CRUD können via SOAP und RESTful Services verwendet werden.[14] Der TextGrid CRUD Service wird in DARIAH als DARIAH-CRUD nachgenutzt, allerdings ist hierfür die Anpassung und Implementierung der AAI-, Metadaten, PID- und Storage-Module notwendig. Außerdem ist im Gegensatz zu TextGrid der CRUD Service in DARIAH lediglich als statischer Workflow vorgesehen, da speziell zur langfristigen Speicherung im DARIAH Repository die DARIAH BitStreamPreservation verwendet werden soll.[15]

Neben dem Aspekt der Nachnutzung besteht in diesem Kontext der weitere Vorteil, dass für TG-crud schon eine API existiert, welche für DARIAH ggf. lediglich angepasst werden muss.[16]

3.2.2 Bitstream Preservation (DARIAH Storage)

Die DARIAH Bitstream Preservation, auch DARIAH Storage genannt (vgl. Grafik S.11), bildet die Speicherebene, auf die der CRUD Service zugreift. Die DARIAH Bit Preservation wurde zur nachhaltigen, sicheren und persistenten Speicherung heterogener geisteswissenschaftlicher Forschungsdaten entwickelt und ist durch die folgenden Eigenschaften gekennzeichnet[17]:

- Daten werden unabhängig von Größe, Format oder Inhalt gespeichert.
- Nur administrative Metadaten werden erstellt und verwaltet. Inhaltlich erschließende Metadaten werden auf dieser Ebene als Datei behandelt.

[13] Vgl. https://dev2.dariah.eu/wiki/download/attachments/14651583/M%204.3.2.1-DARIAH-Repositorium-Prototyp-final.pdf?version=1&modificationDate=1430220062670&api=v2

[14] Vgl. https://dev2.dariah.eu/wiki/display/TextGrid/TG-crud#TG-crud-SOAPandRESTAPI

[15] Vgl. Das Dariah-Repositorium, V4, https://dev2.dariah.eu/wiki/download/attachments/27330089/DARIAH-Repositorium-v4.pdf?version=1&modificationDate=1395055351485&api=v2), S. 4+5.

[16] Vgl. https://dev2.dariah.eu/wiki/display/TextGrid/TG-crud#TG-crud-SOAPandRESTAPI

[17] Vgl.: Tonne, Danah; Rybicki, Jedrzej; Funk, Stefan E.; Gietz, Peter: "Access to the DARIAH Bit Preservation Service for Humanities Research Data", in: Proceedings of the 21st International Euromicro Conference on Parallel, Distributed, and Network-Based Processing, hrsg. v. Peter Kilpatrick; Peter Milligan; Rainer Stotzka, Los Alamitos 2013, S. 9-15.

- Es werden hauptsächlich CREATE und READ Operationen angewendet. Methoden zum Aktualisieren (UPDATE) oder Löschen (DELETE) sind verfügbar, werden allerdings selten bzw. nur administrativ genutzt.
- Es werden Mechanismen zur Sicherstellung der Datenintegrität bereit gestellt.
- Der Zugriff wird sowohl über intuitive, von Forschern einfach zu nutzende als auch über maschinenlesbare Schnittstellen ermöglicht.
- Durch Nutzung der DARIAH Authentifizierungs- und Autorisierungsinfrastrukture (AAI) werden unerlaubte Zugriffe und Modifikationen verhindert.

Ein besonderer Fokus liegt in der DARIAH Bit Preservation auf Modularität und technologischer Nachhaltigkeit. Durch eine Speicherabstraktionsschicht können die anbietenden Institutionen zum einen ihre vorhandenen Speichersysteme wie beispielsweise HPSS (High Performance Storage System) nutzen, zum anderen werden Datenmigrationen bei veralteter Software ermöglicht. Als Software wird in diesem Kontext zur Zeit iRODS eingesetzt, eine von der Universität North-Carolina (UNC) und der Universität of California at San Diego (UCSD) entwickelte und relativ weit bekannte Open-Source Software zur redundanten Verwaltung und Speicherung großer Datenmengen.

Für die Interaktion mit der DARIAH Bit Preservation wurden zwei Schnittstellen spezifiziert, die DARIAH Storage API und die DARIAH Admin API. Beide Schnittstellen sind HTTP- und REST-basiert, gemäß:

> *"REST components perform actions on a resource by using a representation to capture the current or intended state of that resource and transferring that representation between components. A representation is a sequence of bytes, plus representation metadata to describe those bytes. Other commonly used but less precise names for a representation include: document, file, and HTTP message entity, instance, or variant."*[18]

Die DARIAH Storage API bietet mit ihren Methoden POST, PUT, GET, HEAD, DELETE und OPTIONS Funktionalitäten zur einfachen Speicherung von Dateien. Die DARIAH Admin API erlaubt Interaktion mit der Bitstream Preservation Komponente des Systems:

Es ist möglich, das Bitstream Preservation Level für die jeweiligen Dateien zu bestimmen. Das Bitstream Preservation Level entspricht dabei der Anzahl der Repliken (redundant vorgehaltene Kopien), dem verwendeten Prüfsummenalgorithmus und der Häufigkeit der Integritätsüberprüfungen (Neuberechnung der Prüfsumme und Vergleich mit dem gespeicherten Wert). Die spezifischen Ausprägungen für die unterschiedlichen Level können jedoch nicht von dem Nutzer bestimmt werden, sondern werden von der anbietenden Institution festgelegt.

Dateien können als archivierbar markiert werden. Der Nutzer erhält so längere Zugriffszeiten, da Dateien auf Near-/Offlinespeicher verschoben werden dürfen. Falls nötig können erneute Integritätsüberprüfungen veranlasst werden. Zusätzlich können Informationen zu den Dateien wie Anzahl und Lokation der Repliken, verwendeter Prüfsummenalgorithmus, Häufigkeit der Integritätsüberprüfungen usw. abgefragt werden.

[18] Vgl: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm

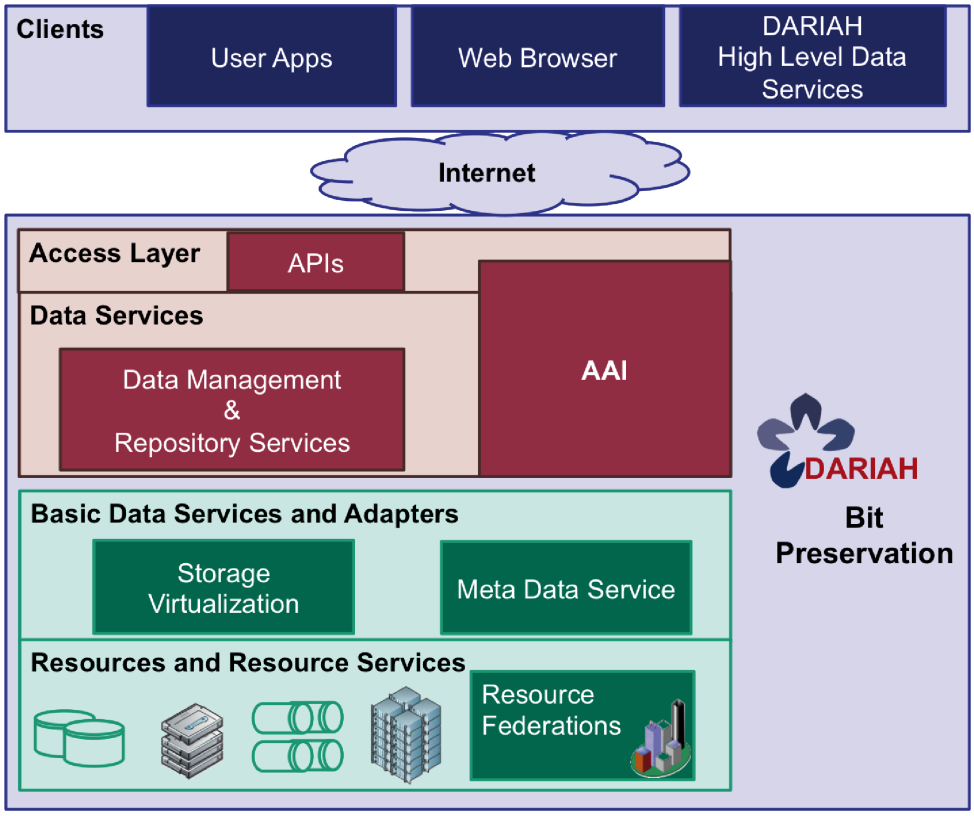

Abb 9. Die DARIAH Storage API

Explizit wird in der Spezifikation 1.0 der Basic Bitpreservation API[19] darauf eingegangen, dass einige Anforderungen auf einer höheren Ebene gelöst werden müssen. Hierbei handelt es sich um Funktionalitäten wie

- das community-spezifische Management erschließender / beschreibender Metadaten,
- das Management technischer Metadaten, bspw. durch Verwendung der Software JHOVE, welche in der Lage ist, diese zu extrahieren,
- Versionierung und Revisionsmanagement,
- gleichzeitiger Ingest einer großen Anzahl von Dateien
- Vergabe und Management von persistenten Identifiern
- und jegliche Form von buchhalterischen Gesichtspunkten.

Die meisten dieser Punkte werden in den folgenden Kapiteln behandelt, in denen weitere Services des DARIAH Repositories beschrieben werden. Einzig der Aspekt der technischen Metadatengenerierung und -beschreibung ist bisher nicht abgedeckt und wird im Rahmen der Langzeitarchivierungsanforderungen am Ende dieses Reports diskutiert.

[19] Vgl: http://handle.gwdg.de:8000/11858/00-1734-0000-0009-FEA1-D

3.2.3 Generische Suche

In Verbindung mit der Collection Registry und dem OAI-PMH Protokoll bietet die DARIAH Infrastruktur die "generische" Suche auf allen Forschungsdaten (-sammlungen), welche im DARIAH-Repository hinterlegt sind, aber auch auf externen Sammlungen von Forschungsdaten an.

Via OAI-PMH ist hier die Abfrage der Metadaten möglich, so dass dem Inhalt dieser Metadaten (DublinCore) beim Ingest der Datensammlungen eine zentrale Bedeutung zukommt...

Speziell der engere, forschungsgetriebene Suchfokus ist hier von besonderer Bedeutung: Ungeachtet der Heterogenität einzelner Sammlungen, soll hier die Möglichkeit nach semantisch tiefer Suche und Auswahl auf Original-Forschungsdaten gegeben werden.

Die Funktionalität der Generische Suche ist somit eng an a) das DARIAH Metadatenschema und b) die Collection-Registry, also das Verzeichnis aller aufgenommenen Forschungsdaten, gekoppelt. Als Basis-Metadatenstandard wird in DARIAH DublinCore verwendet, trotz des Bewusstseins, dass DublinCore gleichzeitig als geringster gemeinsamer Nenner zwischen fachspezifischeren Metadatenstandards wenig inhaltliche Expressivität aufweist[20].

Eine Erweiterung der semantischen Nachnutzbarkeit der DublinCore-Metadaten soll durch die Verwendung von regulären Ausdrücken und Big-Data-Analysten bei der Suche in einzelnen Feldern (Im Text Beispiel: *dc:coverage*) erreicht werden[21].

3.2.4 AAI

Die DARIAH AAI (Autorisierungs- und Authentifizierungs-Infrastruktur) basiert auf SAML. Der OASIS-Standard[22] SAML steht für Security Assertion Markup Language und hat sich im letzten Jahrzehnt sowohl im Hochschulbereich als auch in der Industrie durchgesetzt. Der primäre Anwendungsfall von SAML sind Anwendungen, die eine Authentifizierung erfordern, und Einrichtungen, zu denen Benutzer gehören.

Anwendungen und Einrichtungen sind in der Regel lose innerhalb sogenannter *Föderationen* gekoppelt und gehören oftmals rein administrativ zu verschiedenen Organisationen. Der OASIS-Standard SAML definiert somit das Vokabular, über das diese Organisationen miteinander kommunizieren, damit sich Benutzer der Einrichtungen bei den gewünschten Anwendungen authentifizieren können. Dazu betreibt die Einrichtung einen sogenannten *Identity Provider* (IdP), der an die Benutzerverwaltung angeschlossen ist und ein Login erlaubt, und der Anbieter der Anwendung betreibt einen *Service Provider* (SP), der die Informationen über den Benutzer auf sicherem Wege vom IdP bezieht und an die Anwendung weitergibt.

DARIAH ist Teil der deutschen Hochschulföderation DFN-AAI[23]. Dies bedeutet, dass Mitglieder an deutschen Hochschulen sich für DARIAH-Dienste authentifizieren können. Im Rahmen des europäischen Geant-Projekts nimmt die DFN-AAI an der Meta-Föderation eduGain teil[24], sodass DARIAH-Dienste auch Benutzern aus den dort angebundenen nationalen Föderation offenstehen. Zusätzlich unterhält DARIAH

[20] Vgl: Tobias Gradl, Andreas Henrich: M1.4.2.2: Generische Suche (Facetted Browsing). M1.2.3: Schema Registry, April 2014

[21] Ebd. S. 13-14

[22] Vgl. http://www.oasis-open.org/committees/security

[23] Vgl: https://www.aai.dfn.de/

[24] Vgl. http://www.geant.net/service/eduGAIN/Pages/home.aspx

einen sogenannten *homeless IdP* mit einer eigenen Benutzerverwaltung, der es Benutzern ohne entsprechenden Account an einer Forschungseinrichtung in einer der angebundenen Föderationen ermöglicht, an Projekten und Diensten teilzuhaben.

DARIAH-Dienste können neben den Benutzerdaten, die sie von den IdPs der jeweiligen Einrichtung bekommen, zur *Autorisierung* (d.h. der Entscheidung des Zugriffs eines konkreten Benutzers auf eine bestimmte Ressource) noch weitere Attribute von einer zentralen Instanz abfragen. Dazu fungiert der DARIAH homeless IdP als sogenannte *SAML Attribute Authority*, der die Mitgliedschaft von Benutzern in Autorisierungsgruppen (z.B. alle Mitglieder, die im internen TextGrid-Bereich des DARIAH Wikis Schreibrecht haben) an die Dienste weitergibt. Dazu muss ein Föderationsbenutzer einmalig im Zuge des Erstzugriffs auf einen DARIAH-Dienst bei der Attribute Authority registrieren. Ein Schaubild dieser Infrastruktur ist nachfolgend abgebildet.

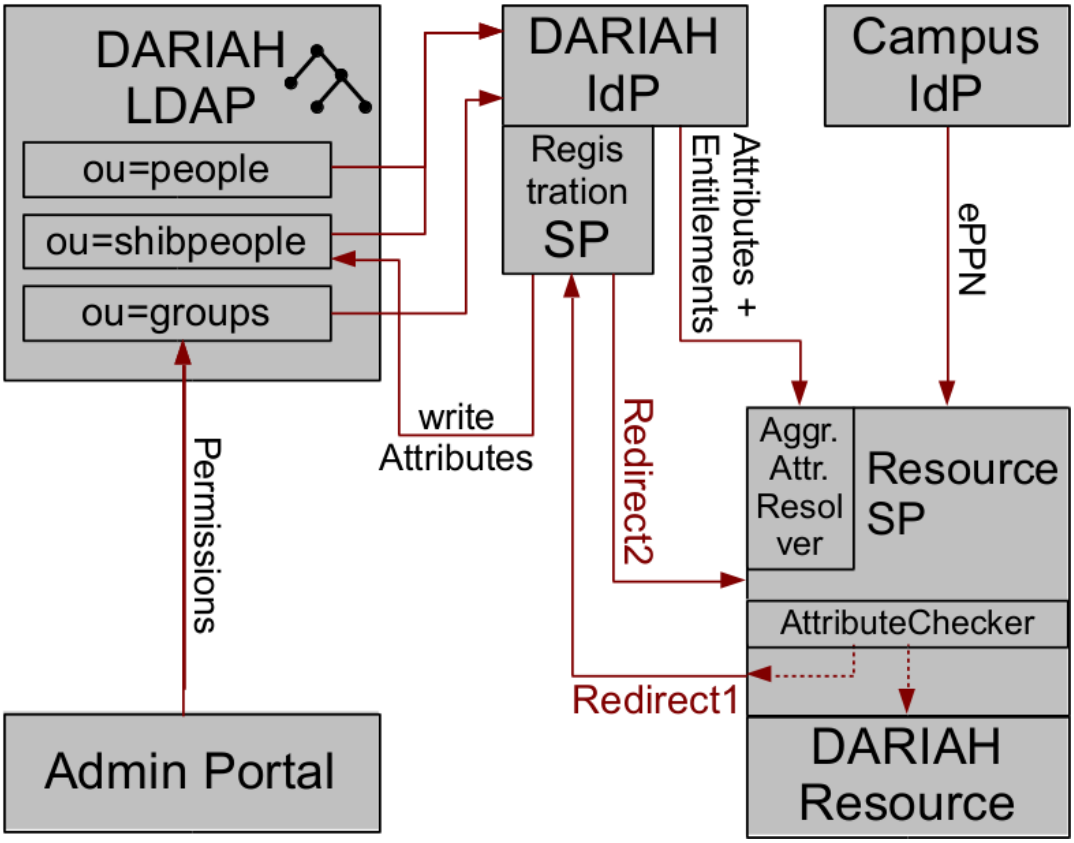

Abb 10. Die DARIAH-DE Autorisierungs- und Authentifizierungs-Infrastruktur. Quelle: DAASI International / DARIAH

3.2.5 Epic PID – Persistente Identifikatoren

Der in DARIAH verwendete EPIC-PID-Service ist ein Dienst zur Erzeugung von persistenten Identifikatoren basierend auf dem Handle System.

Das European Persistent Identifier Consortium (EPIC), welches den Epic PID Service bereitstellt, ist ein 2009 im Rahmen der EU gegründetes Konsortium bestehend aus der Gesellschaft für wissenschaftliche Datenverarbeitung (GWDG), Surf SARA, CSC – IT center for science und dem Deutschen Klima Rechenzentrum, DKRZ.

Im Gegensatz zu anderen persistenten Identifier Ansätzen, hat der EPIC Standard den Vorteil, dass die vergebenen Identifier relativ flexibel in Ihrer Gültigkeit sind, da bei EPIC der Gültigkeitszeitraum eines Identifiers selbst bestimmt werden kann. Dieses System kann zum Beispiel zur Behandlung von Objekten unterschiedlicher Granularität und deren flexiblen Kombination zu neuen Objekten / Kollektion von besonderem Interesse im DARIAH Repository sein.

Das Handle-System, auf dem EPIC aufbaut, wurde von der US-amerikanischen Wissenschaftsorganisation *Corporation for National Research Initiatives* (CNRI) u.a. für das eigene „Digital Object Architecture"-System eingeführt[25]. Handle basiert auf einem Set aus Protokollen mit Referenzimplementation, beispielsweise für *Digital Object Identifier* (DOI) oder die Library of Congress, somit ist auch automatisch das DOI System im DARIAH Repository nachnutzbar.

3.2.6 Publish-Webservice

Die Funktion des DARIAH Publish Services ist es, Daten von diversen DARIAH Publish Clients (zunächst hauptsächlich dem Publish Web-Interface, also der Publish GUI) annehmen, die eingegebenen DC-Metadaten zu validieren, Referenzen auf Objekte innerhalb der einzuspielenden Kollektion von Dateipfaden auf persistente Identifikatoren umzuschreiben und evtl. technische Metadaten zu generieren; weitere Funktionen sind denkbar.

3.2.7 OAI-PMH

OAI-PMH steht für Open Archival Information-Protocol for Metadata Harvesting und basiert auf dem Paradigma des OAIS Modells, vonach archivierte Daten (bzw. deren Metadaten) zum Zwecke der Dissemination möglichste generisch abrufbar sein sollen.

Mithilfe dieses Protokolls können in DARIAH Metadaten bereits archivierte Objekte bereitgestellt und durchsucht werden[26].

Zentraler Gedanke hinter diesem Protokoll ist der Gedanke "open", was ein nicht zu unterschätzender Mehrwert aber auch Limitierung zur Folge hat, so dass bspw. keine Daten mit ungeklärten Urheber- oder Verbreitungsrechten über eine solche Schnittstelle zur Verfügung gestellt werden sollten.

Im Rahmen von DARIAH wird eine OAI-PMH Schnittstelle zur "Generischen Suche" auf den im Ingest generierten DC-Metadaten bereitgestellt, welche dann per ElasticSearch durchsuchbar gemacht werden sollen.

So greift zum Beispiel die Generische Suche auf die Sammlungen der Collection Registry zu (soweit dort eine OAI-PMH-Schnittstelle angegeben ist), und indiziert alle Objekte, die durch OAI-PMH in den einzelnen Kollektionen angefordert werden können.

4 Langzeitarchivierung: Anforderungen und Gapanalyse

Neben der schon existierenden Lösung zur BitStream-Preservation in der Storage-API bestehen zur *langfristigen* Zugänglichkeit und Verarbeitung der geisteswissenschaftlichen Forschungsdaten weitere Anforderungen nach entsprechenden technischen Lösungen in DARIAH, denn:

[25] Vgl. http://www.handle.net/

[26] Vgl: Eine Demoabfrage: http://demo2.dariah.eu/colreg/OAIHandler?verb=ListRecords&metadataPrefix=dclap

1. Der Bitstream einer exemplarischen CorelDraw Datei ist vermutlich in 10 Jahren[27] zwar noch vollständig erhalten aber womöglich nicht mehr durch lauffähige Software interpretierbar.
2. Es ist weiterhin zu vermuten, dass die in DARIAH gesammelten Forschungsdaten auch über 10 Jahre hinaus von solcher Relevanz sind, dass Geisteswissenschaftler an Ihnen Interesse haben werden, und sie diese daher für Ihre Forschungsfragen nachnutzen möchten.
3. Es existiert bisher kein Formatmigrations- oder Emulationswerkzeug in der DARIAH-Infrastruktur, welches eine Datei in einem veralteten Format (s.o. das Beispiel mit CorelDraw) in ein aktuelles, interpretierbares Dateiformat migrieren oder eine emulierte Version veralteter Interpretationssoftware zur Darstellung anbieten könnte. Hier lassen sich womöglich schon an anderer Stelle eingesetzte Lösungen, wie Kolibri, nach nutzen. Ein solcher Ansatz bedarf aber noch der Evaluation der beteiligten Partner.
4. Der verwendete Metadaten-Standard DublinCore enthält keine Informationen über die Art der Datei, die gespeichert wird, ihre Versionen oder ihren rechtlichen Zustand.
5. Bisher werden keine weiteren Metadaten erfasst, welche technische Dateiinformationen enthalten und durchsuchbar machen. Hier ist der Einsatz von JHOVE 2 geplant.
6. Neben technischen Metadaten sind interne Strukturmetadaten insbesondere dann sinnvoll, wenn die gespeicherten Dateien nicht mit einer inhaltlichen 1:1 Beziehung zu Ihren Metadaten abgebildet werden können (bspw: Seiten eines Buches, jede als Bild gespeichert).
7. Gerade zur Abbildung eines Gesamtworkflows für einzelne Forschungsdaten eignen sich Metadatenstandards, wie Premis o.ä., welche für diese eine geeignete Struktur anbieten, besonders.

Zur vollständigen Abdeckung des Research Data Lifecycle ist daher eine sinnvolle Unterstützung von Features aus dem Langzeitarchivierungsbereich unumgänglich, wie sie auch in der erweiterten Darstellung des Research Data Lifecycle genannt werden:

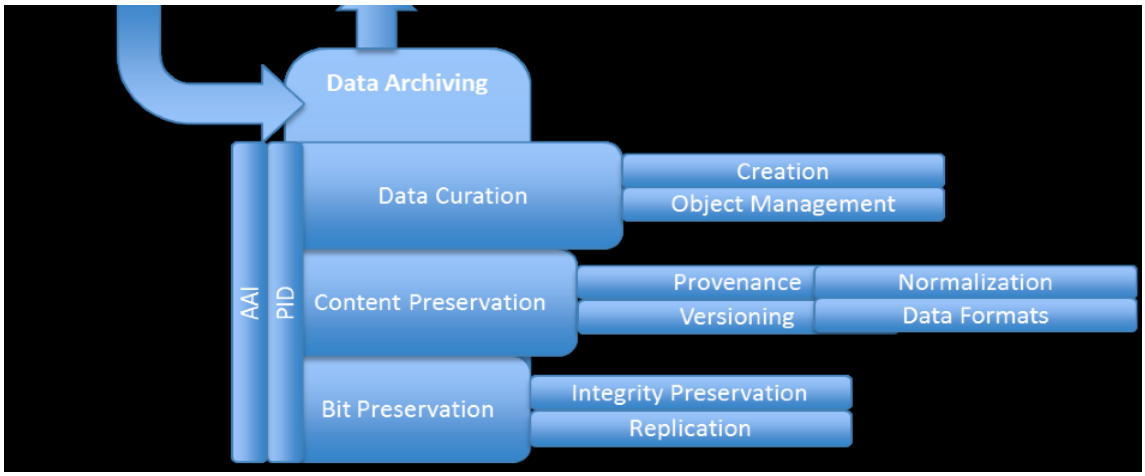

Abb 11: Research Data LifeCycle aus der Data Documentation Initiative - Erweiterung von DARIAH-DE. Der erweiterte Fokus um Langzeitarchivierungskomponenten.

[27] Die DFG gibt in Ihren Richtlinien zur Sicherung guter wissenschaftlicher Praxis eine Mindestaufbewahrungsdauer für Primärdaten von 10 Jahren an. Vgl. http://www.dfg.de/foerderung/grundlagen_rahmenbedingungen/gwp/

Die Grafik enthält hier neben der Bitstream-Preservation, die im DARIAH Repository abgedeckt ist, zwei weitere Schichten, nämlich die langfristige Administration (Data Curation) und die langfristige inhaltliche Erhaltung (Content Preservation).

Im Rahmen der AG Research Data LifeCycle wird für beide Termini eine Definition erarbeitet. An dieser Stelle wird eine erste Anforderungssammlung formuliert werden, die ggf. nach Definition der Termini durch die entsprechenden Gremien angepasst werden kann.

4.1 Grundlagen

Erhaltung und Pflege von archivischem Material sind grundsätzlich Tätigkeiten im Umfeld der Einrichtungen, die zur Bewahrung des kulturellen Erbe dienen. Bei analogem Material ist dies also in erster Linie Aufgabe von Bibliotheken, Archiven und Museen.

Während aber im analogen Bereich die Aufbewahrungsdauer und Materialhaltbarkeit des zu archivierenden Materials relativ klar zu bestimmen sind, sowie die zu ergreifenden Maßnahmen als eindeutig definiert gelten, ist dies im Bereich des Digitalen nicht der Fall:

So legt beispielsweise die DFG in Ihren Richtlinien "zur guten wissenschaftlichen Praxis" eine Aufbewahrungsfrist für wissenschaftliche Primärdaten in den von Ihr geförderten Projekten von 10 Jahren fest[28]. Es ist allerdings an den geförderten Einrichtungen häufig nicht festgelegt, welche Abteilung sich um solch eine Sicherung bemüht (Bibliothek oder Rechenzentrum) und erst recht fehlt in der Regel das Bewusstsein und die Überprüfung, die die Einhaltung solch einer Aufbewahrungspflicht sicherstellen würden.

Neben der ordnungsgemäßen Aufbewahrungsdauer von 10 Jahren auf Datenträgern sind weitere Risiken zu nennen, wenn man annimmt, dass die Daten über die Frist von 10 Jahren hinaus verfügbar und nutzbar sein sollen:

- Grundsätzlich besteht bei langfristigen Speicherung die Gefahr der Bit Deterioration, also die Gefahr, dass aufgrund von Materialermüdung oder Schäden, Teile einer Datei / eines Datenstroms nicht mehr lesbar / interpretierbar sind. Dieser Gefahr sollte grundsätzlich durch mehrere Maßnahmen begegnet werden, nämlich
  - Hardwaremigration
  - Redundante Speicherung
  - Vielfalt unterschiedlicher Speichersysteme
- Neben der Gefahr, dass der Verschleiß von Datenträgern zu einer Zerstörung einzelner Daten führt, besteht die wesentlich weniger einschätzbare Gefahr für den Zugriff auf archivierte Dateien in der Überalterung von Soft- & Hardware, in der diese zu interpretieren sind.

Im Kontext des Digitalen und im Fokus von diesen prekären Umständen, sind solche kuratorischen Aufgaben inhärent in jede Form eines Repositories mit dem Anspruch langfristiger Zugänglichkeit einzubinden.

Als Grundlage eines Archivierungsprozess hat sich mittlerweile das weithin bekannte OAIS

[28] Vgl. http://www.dfg.de/download/pdf/dfg_im_profil/reden_stellungnahmen/download/empfehlung_wiss_praxis_1310.pdf

Referenzmodell durchgesetzt, dessen Termini speziell für die Objekte, welche aufgenommen und am Ende ausgegeben werden, hier aufgegriffen werden sollen:

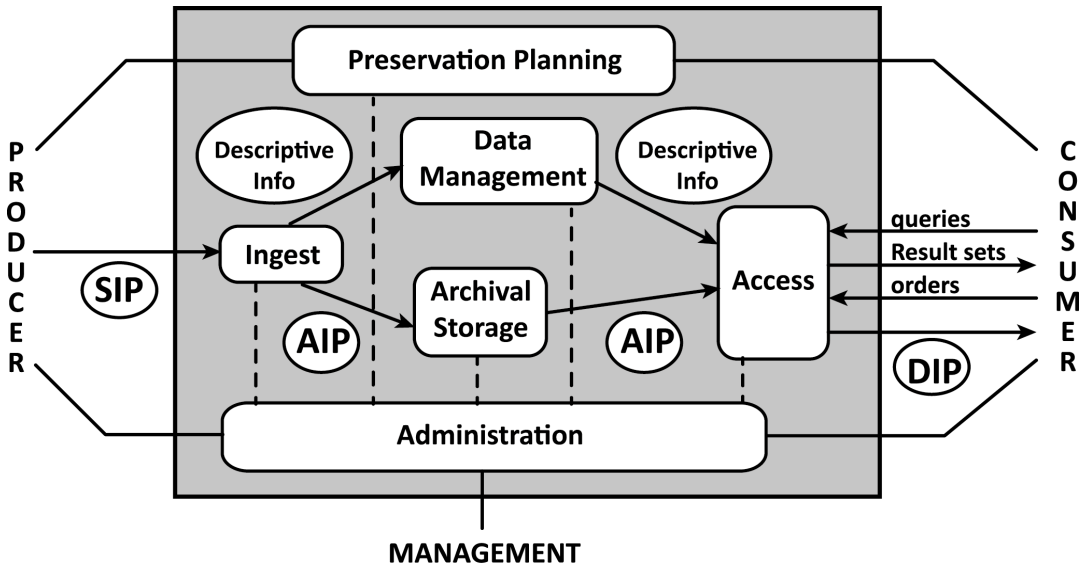

Abb. 12: The OAIS functional model, http://public.ccsds.org/publications/archive/650x0m2.pdf. S. 4-1

SIP, AIP und DIP bezeichnen Datenpakete, deren interne Struktur hier nicht festgelegt sein soll, die aber die inhaltlich gleichen Informationen an unterschiedlichen Positionen des Verarbeitungsprozess spiegeln:

- Ein Datenpaket, welches in den Verarbeitungsprozess aufgenommen wird, wird als Submission Information Package (SIP) bezeichnet und enthält alle Dateien, die als eine Einheit dem archivischen Prozess übergeben werden
- Das gleiche Datenpaket wird innerhalb des archivischen Ablagevorgangs als Archival Information Package (AIP) bezeichnet. Das AIP bezeichnet also das (möglichst) identische inhaltliche Dateipaket, wie das im "Ingest"-Prozess verarbeitet SIP, allerdings u.U. in anderer Zusammensetzung und Ausprägung. Das AIP wird langfristig abgelegt.
- Der Counterpart zum "Ingest" ist im OAIS Modell der "Access", also der Zugriff auf archivierte Pakete, diese (abrufbaren) Pakete werden als Dissemination Information Package (DIP) bezeichnet. Sie müssen nicht zwangsläufig alle Dateiobjekte in allen Formaten enthalten, wie sie im Archiv als AIP abgelegt wurden. Es kann u.U. sinnvoll sein, auflösungsreduzierte Versionen von Dateien an Endnutzer herauszugeben, sei es aus rechtlichen, sei es aus Gründen langsamer Datenleitungen. Auch sind bestimmte administrative Metadaten, welche im AIP zu speichern sinnvoll ist, nicht zwangsläufig an Endnutzer herauszugeben.

4.2 Verarbeitung

Die folgenden Kapitel listen mögliche Verarbeitungsschritte in einem Langzeitarchivierungssystem auf. Grundsätzlich wird angestrebt, die genannten Verarbeitungsschritten den beiden Terme "Data Curation"

und "Content Management" zuzuordnen.

Dabei sind alle Arbeitsschritte, die der manuellen (d.h. menschlichen) Betreuung bedürfen, dem Term "Data Curation" zuzuordnen, während alle Tätigkeiten, die maschinell erfolgen können (z.B. Registrierung und Normalisierung von Objekten), "Content Management" genannt werden.

Grundsätzlich konzentriert sich dieses Kapitel auf Arbeitsschritte, die (halb-)automatisch erfolgen können, also eher dem Feld des "Content Management" zuzuordnen sind. Tätigkeiten, die der Data Curation zuzuordnen sind, sind maximal teilautomatisierbar. Hierunter fallen:

- Zuordnen von Dateien zu Sammlungen,
- Hinzufügen von rechtlichen Bestimmungen (Lizenzen) zu Sammlungen oder einzelnen Dateien
- Alle Arten von manueller Bearbeitung der Metadaten zum Zwecke der erweiterten Suchbarkeit, Administration, Datenmanagement

4.2.1 Formaterkennung und -validierung

Grundsätzlich ist es notwendig, die in ein Archiv aufgenommen Datenpakete (SIPs) in Ihre Bestandteile zu zerlegen und das Dateiformat der einzelnen enthaltenen Dateien zu erkennen. Denn nur eine Datei, deren Dateiformat bekannt ist, kann auch mit einer entsprechenden Software verarbeitet werden.

Zur Erkennung der Daten wird standardmäßig die Software Pronom, JHOVE bzw. JHOVE2, empfohlen.

Man unterscheidet genaugenommen zwischen der Identifikation eines Formats (Hier handelt es sich um eine JPEG-Datei gemäß Standard XY) und deren Validierung (Die Datei ist *valide* gemäß Spezifikation XY). Validierung wird folglich nicht von allen genannten Tools geboten, sondern u.a. von JHOVE, einem von der Harvard Library entwickeltem Tool, welches relativ umfangreiche Funktionen – aber nicht für alle Dateiformate bietet.

Die Software FITS[29] ist der Versuch des gleichen Herstellers, die Funktionalität dieser und weiterer Werkzeuge (wie Tikka und Droid) zu bündeln, indem sie alle in ein Tool integriert werden.

4.2.2 Metadatenerzeugung

Das Ergebnis der soeben genannte Funktion "Formaterkennung und -validierung" sollte standardmäßig in Metadaten gespeichert werden, damit die so erhobenen Informationen dauerhaft zum Abruf bereitstehen.

Neben diesen eher technischen Metadaten gibt es offensichtliche weitere Informationen, die im administrativen Bereich interessant sind. Hierzu gehören

- die Version der Datei (wenn sie bspw. in ein neues Format migriert wurde)
- die Herkunft der Datei (Institution, Person…)
- rechtliche Informationen (Lizenzname…)
- verschiedene kalendarische Informationen (Datum der Erstellung, Datum der einzelnen Versionen, Datum der Metadatenerhebung)

[29] Vgl. http://projects.iq.harvard.edu/fits

- Abbildung struktureller Informationen: wenn es sich um mehr als eine Datei im SIP handelt: wie stehen diese Dateien miteinander in Beziehung? Ist das eine die Metadatei der anderen?
- Ggf. Informationen über Prüfsummen oder eindeutige Identifier (Im Rahmen von DARIAH offensichtlich PIDs, aber ggf. auch URNs oder andere)

All diese administrativen Informationen lassen sich beispielsweise mit Hilfe des Metadatenstandards "PREMIS" der Library of Congress abdecken. PREMIS bietet außerdem die Möglichkeit als Container für andere Metadatenstandards zu dienen, so können bspw. in DublinCore angelegte Felder in eine PREMIS Datei übernommen werden.

Auch die Aggregation weiterer externer Informationen, wie z.B. das Ergebnis einer Formatvalidierung oder der Output einer Extraktion technischer Metadaten (Bildbreite, Bildhöhe, Farbtiefe, wie mit der Software JHOVE möglich) ließe sich so aufnehmen.

Grundsätzlich muss an dieser Stelle zwischen manueller und automatischer Metadatenerzeugung unterschieden werden: Manuelle Metadatenerzeugung ist immer dann sinnvoll, wenn Daten inhaltlich beschrieben und erschlossen werden sollen, bspw. durch Ergänzung von Schlüsselwörtern zur besseren Auffindbarkeit oder durch Anfügen von inhaltlichen Anmerkungen zum Material. Generell sollte es der Anspruch einer Langzeitarchivierungsfunktion sein, möglichst viele Metadaten automatisch zu generieren.

Im Falle von DARIAH könnten u.U. selbst händisch generierte Metadaten aus den Daten mithilfe eines Crosswalks automatisiert in standartisierte Metadaten zu einzelnen Objekten für die Langzeitarchivierung überführt werden. Der aktuelle Entwicklungsstand des DARIAH Repositories sieht bisher einzig im Ingest eine manuelle Erzeugung von Metadaten in DublinCore Felder vor.

4.2.3 Emulation und Migration

Standardmäßig wird der Gefahr der Überalterung von Software und Dateiformaten mit den Methoden "Migration" und "Emulation" begegnet. Dabei stellt Migration das Vorgehen zu Überführung eines Dateiformates in ein anderes dar (="Konvertierung"), während Emulation die Verarbeitung von Originaldateien in "emulierter" (d.h. nachempfundener) Software beschreibt.

Beide Ansätze bergen Vor- und Nachteile, auf die an dieser Stelle nicht näher eingegangen werden soll. Generell stellt Migration die unproblematischere Strategie zur Erhaltung der Lesbarkeit von Dateien dar, da hier weniger Implementationsaufwand besteht.

4.2.4 Dateiformate

Ein nicht zu unterschätzendes Problem bei der Langzeitarchivierung von Dateien ist die (Aus-) Wahl der zu unterstützenden Dateiformate (in die beispielsweise migriert werden soll). Grundsätzlich wird hier zwischen Quell- und Zielformaten unterschieden.

Quellformate sind alle die Dateiformate, die in das System eingespielte Objekte beinhalten können. Bei einer momentan kolportierten Anzahl von 1674[30] auf der Welt existierenden Dateiformaten, besteht

[30] http://www.dmoz.org/Computers/Data_Formats/. Rosetta hat zur Zeit 146 (weit verbreitete) Dateiformate registriert, vgl: http://www.fileformat.info/format/, Wikipedia zählt weit mehr: http://en.wikipedia.org/wiki/List_of_file_formats

hier die Notwendigkeit sehr genau einzugrenzen, welche Dateiformate ein System maximal unterstützen soll. Zielführend sind hier meist Stakeholder-Umfragen nach den meist genutzten Formaten.

Ein weiteres Augenmerk gilt der Auswahl der Ziel-Dateiformate. Folgende Kriterien sollten u.A: hier besonders relevant sein[31]:

- Keine oder zumindest keine verlustbehaftete Kompression
- Frei verfügbare Spezifikation
- Frei von Patenten und Lizenzen
- Weite Verbreitung (eine große Anzahl gut bekannter Software, die die genannten Formate verarbeiten kann)

Generell ist es zielführend für jeden Medientyp (Textdatei, Bilddatei, Audiodatei, Videodatei) mindestens ein *Zielformat* zu bestimmen, in welches migriert wird. Dabei stellen Mischformate, wie HTML oder auch einige Formen von PDF eine besondere Herausforderung dar und müssen ggf. weiter unterteilt behandelt werden. Häufig empfohlen werden für die gängigsten Dateitypen Bild und Text: PDF/A für Office-Dokumente und Tiff bzw. JPEG-2000 für Bilder[32]. Weitere Formatempfehlungen wurden in der AG RDLC erarbeitet.[33]

4.2.5 Bedarf nach administrativer Verarbeitung

Neben den für eine Langzeitarchivierung offensichtlich relevanten Funktionen, ist eine Vielzahl von administrativen Features für eine vollständige Abdeckung eines Workflows notwendig.

Hierzu können Funktionen zählen, wie

- die Registrierung von Datenobjekten in einem System,
- die Registrierung von unterschiedlichen Versionen desselben Dateiobjekts (vor und nach einer Migration),
- die Registrierung komplexer Objekte, die sich aus Komponenten unterschiedlicher anderer Objekte zusammensetzen
- das Versehen mit Identifiern,
- die Verzeichnung von URI's in den Metadaten komplexer Objekte
- u.U. die maschinelle Berücksichtigung unterschiedlicher Berechtigungseinstellungen, sowohl für die spätere Weiterverarbeitung, als auch für die Publikation

Generell gilt die eindeutige Definition von Pakettypen (wie werden SIPs, AIPs, DIPs definiert) als nicht-triviale Aufgabe und sollte strategisch geplant werden.

Zu dieser Definition gehört die Beantwortung der Frage, wie abgeschlossen oder wie veränderbar ein solches Datenobjekt sein soll:

---

[31] Vgl. Brown, Adrian: Selecting File Formats for Long-Term Preservation. The Nation Archives UK, 2008. http://www.nationalarchives.gov.uk/documents/selecting-file-formats.pdf

[32] Vgl. Empfehlungen der Florida Digital Archives: http://fclaweb.fcla.edu/uploads/Lydia%20Motyka/FDA_documentation/recFormats.pdf

[33] Vgl. https://dev2.dariah.eu/wiki/pages/viewpage.action?pageId=38080370

- Sollen Änderungen an archivierten Daten im Nachhinein erlaubt werden?
- Wie eng sollen Beziehungen zwischen Datenobjekten erlaubt sein?
- Wie werden diese abgebildet (in Datenbanken, in dazugehörigen Metadaten?)
- Ob und welche Containerformate sollen als "Verpackung" um die einzelnen Datei-Aggregationen, die Pakete oder Objekte genannt werden, fungieren?

4.2.6 Hardware und ortsunabhängige Speicherung

Generell gelten die folgenden Grundsätze auf dem Feld der Hardwarespeicherung:

Zum einen ist es so, dass aufgrund der häufigen Paradigmen- und Technologiewechsel ein steter Migrationsfluss eingeplant werden muss: Alle paar Jahre müssen aufgrund von Verschleißerscheinungen oder Technologieobsolszenz Hardwarebausteine ausgetauscht werden. Solche Operationen beinhalten immer einen Kopierprozess Folge, bei dem sicher gestellt werden muss, dass die Integrität der Dateien erhalten bleibt (normalerweise über Prüfsummen gelöst).

Zweitens gelten Dateien nur dann als sicher aufbewahrt, wenn Sie mehrfach an verschiedenen Orten gespeichert werden: Alleine aufgrund von Bitfehlern können nur einmalig gespeicherte Dateien durch eine einfach Routine-Übertragung auf einen neuen Server beschädigt und im schlimmsten Falle unbrauchbar werden.

Eventuelle Naturkatastrophen, Brände etc. können zudem ALLE Hardware an einem Standort gleichzeitig zerstören, weswegen Back-Ups von Dateien grundsätzlich an mehreren Orten (mit möglichst einigen Kilometern Distanz dazwischen) vorzuhalten sind.

Hier sollte außerdem auf eine gewisse Vielfalt unterschiedlicher Hardware und Betriebsysteme geachtet werden, so dass bei der unbeabsichtigten Wahl eines besonders Bug-anfälligen Systems dies nicht die vollständige Unbrauchbarkeit aller Kopien einer Datei zur Folge hat, sondern an anderer Stelle die gleiche Datei auf einem anderen stabileren System intakt bleibt.

Drittens sollte bei der Wahl von Hardware und Betriebssystem solchen Systemen Vorzug gegeben werden, die Dateien "unfragmentiert" also nicht auf unterschiedliche Speicherblöcke verteilt, bewahren oder regelmäßig in diesen Zustand zurück versetzen.[34]

Auf diese Weise erhöht sich die Wahrscheinlichkeit, dass auch Dateipakete vollständig wieder hergestellt werden können, wenn Teile der Hardware zerstört sind.

4.3 Fazit

Folgende gestaffelte Reihenfolge von Lösungsansätzen sind für ein LZA-System in einem DARIAH Research Data LifeCycle denkbar:

- Fall 1: (mehrfach redundante) BitStream-Preservation + Formaterkennung + Generierung administrativer Metadaten

[34] Sears,R.,& van Ingen, C. (25.12.2006). Fragmentation in Large Object Repositories – Experience Paper. University of Winsconsin - Madison Database Systems Group:http://wwwBdb.cs.wisc.edu/cidr/cidr2007/papers/cidr07p34.pdf

- Fall 2: (mehrfach redundante) BitStream-Preservation + Formaterkennung + Generierung administrativer Metadaten + Generierung technischer Metadaten
- Fall 3: (mehrfach redundante) BitStream-Preservation + Formaterkennung + Generierung administrativer Metadaten + Generierung technischer Metadaten + Migration einer begrenzten Anzahl von Dateiformaten
- Fall 3: (mehrfach redundante) BitStream-Preservation + Formaterkennung + Generierung administrativer Metadaten + Generierung technischer Metadaten + Migration einer begrenzten Anzahl von Dateiformaten + erneute Formatvalidierung

5. Workflowbeschreibung – Paradigmen und Technologien

In diesem Kapitel soll ein Überblick über aktuell verwendete und weit verbreitete Paradigmen und Technologien zu Beschreibung und Austausch von modularisierten, web-basierten Diensten vermitteln. Ziel dieses Überblicks ist es, die Entscheidung für eine Technologie zur Ansprache der APIs in einem Workflow möglichst transparent und nachvollziehbar zu machen.

5.1 WSDL

Bei WSDL handelt es sich um einen XML-basierten Standard zur Beschreibung und Ansteuerung von Webservices in einem SOAP-Framework.

SOAP bezeichnet ein XML-basiertes Protokoll, mithilfe dessen eine Menge von Programmen (Webservices) angesteuert werden kann sowie zwischen diesen Webservices Informationen ausgetauscht werden können.

So stehen in einer WSDL-Datei maschinenlesbare Informationen zur angesprochenen Schnittstelle, zum Protokoll und Deployment, sowie alle weiteren Formate und Port-Informationen, die notwendig sind, damit Programme sinnvoll Daten zwischeneinander weitergeben können.

5.2 WADL

Neben WSDL für SOAP-Architekturen, existiert das im Vergleich relativ neue WADL (Web Application Description Language) für Architekturen gemäß dem REST Paradigma, welches als Alternative zu SOAP gehandelt wird. Die Definition von WADL ist verglichen mit WSDL recht offen:

> *"WADL is designed to provide a machine processable description of HTTP-based Web applications."*

Auch ist REST im Gegensatz zu SOAP nicht als einheitliche technische Spezifikation definiert sondern bezeichnet ein Paradigma oder "Design-Konzept" zur Ansprache von netzbasierten Services und der Weitergabe von Informationen zwischen diesen. Das REST Paradigma beschreibt die Basis zur Charakterisierung und Beschränkung der Interaktionsmöglichkeiten auf der Makroebene:

> *"REST exemplifies how the Web's architecture emerged by characterizing and constraining the macro-interactions of the four components of the Web, namely origin servers, gateways, proxies and clients, without imposing limitations on the individual participants"*[35]

WADL ist also die "Beschreibungssprache" für Webanwendungen, welche gemäß dem REST-Paradigma gestaltet wurden. Ein wichtiger Unterschied zwischen WADL und WSDL sind MIME-Types:

WADL verwendet MIME-Types zur Erkennung der zu verarbeitenden Dateiformate. Je nach hinterlegtem MIME-Type in einer WADL-Datei stehen unterschiedliche Operationen zur Verarbeitung durch andere Webservices bereit.

Weiterhin unterscheiden sich die beiden Sprachen durch die Architekturen, auf die sie sich beziehen: Das REST-Paradigma richtet den Fokus stark auf Kommunikation zwischen Clients und Servern, während SOAP eher von "services" ausgeht [36].

Insgesamt lässt sich festhalten, dass REST und damit WADL als die flexiblere und state-of-the-art Alternative zur Beschreibung und Ansprache von Webservices gilt[37]. Auch ist zu bedenken, dass alle APIs, welche im Kontext des DARIAH Repositories implementiert worden sind, entweder schon RESTful sind oder aber mit diesem Paradigma geplant werden.

5.3 BPEL und GWES

Zur Verbindung von mit WSDL beschriebenen Webservices zu einem komplexeren Workflow wurde eigens WS-BPEL[38] entwickelt. Hier handelt es sich um eine aus dem Umfeld der Wirtschaftsinformatik stammende XML-basierte Sprache, mit Hilfe derer Webservices gemäß einer SOA in bestimmte Geschäftsabläufe gekoppelt, also "orchestriert", werden können. Aus diesem Grund eignet sie sich perfekt zur Verbindung von mit WSDL beschriebenen einzelnen Services zu einem oder mehreren Workflows.

Eine Übersicht bestehender open-source Software zur Implementation von BPEL in eine Infrastruktur wurde in TextGrid erstellt und kritisch beleuchtet[39]. Allerdings wurde aufgrund der prekären Marktlage bei Open-Source Tools, die diesen Standard umsetzen, im TextGrid-Lab ein eigener Ansatz entwickelt, der auf dem Fraunhofer entwickelte GWES basiert[40]. Weiterhin wurde bereits in TextGrid ein Workflow-Tool entwickelt, welches basierend auf GWES die Möglichkeit vorhält, in einer GUI Services miteinander zu kombinieren.[41]

---

[35] Vgl: Todd Arias: The Cloud Computing Standards Handbook - Everything you need to know about Cloud Computing Standards. 2007. Tebbo. S.43

[36] http://bitworking.org/news/193/Do-we-need-WADL

[37] Vgl. Vgl: http://www.torsten-horn.de/techdocs/jee-rest.htm#Vergleich-REST-SOAP

[38] Vgl. http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf

[39] Vgl. http://www.textgrid.de/fileadmin/berichte-1/report-1-5.pdf

[40] Vgl. https://dev2.dariah.eu/wiki/display/TextGrid/TG-workflow

[41] Vgl: https://dev2.dariah.eu/wiki/display/TextGrid/Workflow+Tool

5.4 Fazit

Aus der geschilderten Übersicht geht hervor, dass – wenn Workflows über eine gewisse Komplexitätsgrenze hinaus in DARIAH technisch implementiert werden sollten – dies am ehesten mit einem auf WSDL / WADL basierenden Ansatz geschehen sollte. Hier ist auf die Erfahrungen in TextGrid aufzubauen und – wenn fachlich nichts dagegen spricht – können auch technische Lösungen direkt übernommen werden. Es ist also eine Lösung mithilfe von WSDL / WADL und BPEL anzustreben. Da GWES seit einiger Zeit nicht mehr weiter entwickelt wird und auch von der Fraunhofer-Gesellschaf weder gepflegt noch angeboten wird, ist es zunächst nicht sinnvoll, diese Komponente in DARIAH zu übernehmen. Eine Zusammenarbeit mit TextGrid ist jedoch wünschenswert, wenn neue Workflow-Services evaluiert und/oder entwickelt werden sollen.

6. Workflows für den Basis-Research Data Lifecycle

Dieses Kapitel soll zum einen den Basisfall des Forschungsdatenzyklus durch das DARIAH-Repository mit all seinen oben geschilderten Komponenten veranschaulichen und zum anderen Richtlinien erwartbarer Ergebnisse für die einzelnen Schritte definieren.

6.1 Der Basisfall

Die folgende Tabelle enthält eine erste Übersicht aller erwünschten Funktionen des Research Data LifeCycle bezogen auf alle vorhandenen Komponenten der DARIAH-DE Infrastruktur.

Schritt	**Aktivität**	**Notiz**	**Korrespondierender DARIAH-DE Service**
1	Formulierung Forschungsfrage, Benennung von Methoden	Die Formulierung der Forschungsfrage als auch die Benennung der dazu verwendeten Methoden sind ein kreativer Akt und können nicht automatisiert werden.	Generische kollaborative Werkzeuge
2	Auswahl Primärdaten	Ist eng an die Forschungsfrage geknüpft und damit nicht automatisierbar	Suche in der Collection Registry
3	Vorbereitung / Verwendung von Tools	Ist prinzipiell (teil) automatisierbar. Kommt auf den Kontext an	Ggf. Nutzung einzelner in DARIAH-DE bereitgestellter oder weiter entwickelter Tools, wie eCodicology, Digivoy, MEISE,

			Geobrowser, Monasterium...
4	Generierung von (Zwischen-) Ergebnissen/ Verwendung von Tools	Wenn alle Vorbereitungen getroffen und spezifiziert sind: Ja	Nutzung externer Tools
5	Visualisierung	"	Nutzung externer Tools, Geo-Browser
6	Beschreibung der veröffentlichungs-würdigen Ergebnisse und Erkenntnisse	Ist ein kreativer Akt.	Generische kollaborative Werkzeuge
7	Kuration	Iterativ	DARIAH-DE, Publish GUI

Darauf aufbauend enthält die folgende Tabelle alle iterativen Funktionen eines Research Data LifeCycle und den korrespondierenden Service in der DARIAH-DE Infrastruktur:

A	**Identifiervergabe**	**Iterativ**	**EPIC PIDs**
B	Metadatenanreicherung /-abgleich	Iterativ (Deskriptive Metadaten müssen manuell vergeben werden)	Content Metadaten: Schema-Registry. Admin-Metadaten: DublinCore bzw. DARIAH-DE Collection Level Description Application Profile (DCCAP-based)
C	Lizensierung	halb automatisch	Work in Progress, vorerst mit CC-BY 4.0 lizensiert
D	Publikation	Iterativ	Collection Registry, Publish GUI
E	Peer-Review	Iterativ	Ggf. durch Nutzung von kollaborativen Tools, wie Wikis
F	Langzeitarchivierung	Iterativ	DARIAH-DE Bit-Preservation Service

6.2 Datenmodell / Konzeptionelles Modell

Neben der grundsätzlichen Ausbaufähigkeit des Langzeitarchivierungszweiges kristallisiert sich immer weiter der Bedarf nach einem Datenmodell bzw. einem konzeptionellen Modell für die Abbildung des Research Data lifeCycle heraus:

Damit eine Infrastruktur zwischen der ersten Version eines Forschungsdatums und seinen Derivaten in anderen Forschungsprojekten und Forschungsdatensammlungen unterscheiden kann, muss möglichst in den dazu bereitgestellten Metadaten (also unabhängig von den in einem solchen System meist eingesetzten Datenbanken) die Provinienz einer Datei und Ihre "Bearbeitungsgeschichte" abgebildet werden. Auch ist eine möglichst automatisierte Implementation von Workflows oder mindestens Workflowbestandteilen mithilfe eines Datenmodells erstrebenswert.

Im Gegensatz zum TextGrid-Lab, in welchem Daten in einem geführten Modus bearbeitet werden können und gemäß vorgeschriebener Standards mit Metadaten angereichert worden sind, kann in der DARIAH-DE Infrastruktur nicht auf solche Features zurück gegriffen werden. Hier muss mit der Anlieferung von allen Arten von Metadaten gerechnet werden:

Informationen über die Herkunft eines Scans können sowohl im TEI-Header in der dazugehörigen XML-Datei als auch in METS, LIDO oder EAD stehen. Neben einer Vielzahl weiterer Standards sind auch gänzlich unspezifizierte Metadaten oder eingebettete Metadaten in Form von EXIF-Daten in einem Scan selbst möglich.

Nichtsdestotrotz besteht natürlich auch in DARIAH-DE der Wunsch, die Nachnutzung von Forschungsdaten zu analysieren und zu quantifizieren. Daher wird als Minimallösung zumindest die Extraktion und der Abgleich von schon vorhandenen Metadaten sowie die Überprüfung der Nachnutzung bestehender Forschungsdaten auf der Ebene von persistenten Identifiern empfohlen: Da Identifier pro Datei genau **einmal** vergeben werden sollten und sich eine Datei nach einer Publikation inhaltlich nicht mehr ändern sollte, können anhand von Identifiern "Lebenszyklen" von Dateien überprüft werden. Darüber hinaus ist natürlich eine automatisierte Analyse der stattgefundenen Operationen auf den Forschungsdaten, sowie Ihre Kontextualisierung mit anderen Daten und Akteuren wünschenswert. In DARIAH-DE hat bereits eine Untersuchung dieses Themenkomplexes statt gefunden.[42]
Hier wurde der Bedarf nach Provinienzmetadaten untersucht. Im Kontext des Research Data LifeCycle ließe sich auch vom Bedarf nach einem Datenmodell sprechen.

Ein Provinienzsystem wird in dem zitierten Dokument wie folgt definiert:

> *"Wir bezeichnen einen Komplex aus Speichermedien und Software dann als "Provenienzsystem", wenn es in der Lage ist, Provenienzinformationen zu diesem Projekt bzw. zu Forschungsfragen aufzunehmen, zu verwalten und entsprechende Recherchen zu unterstützen bzw. zur Beantwortung diesbezüglicher Fragen zu führen. Dieses System muss zugleich die Sicherheit gewährleisten, die für den gegebenen Arbeitsbereich sowohl von Entwicklern als auch geisteswissenschaftlichen Nutzern gefordert wird."*[43]

Ein solches Provinienzsystem kann in DARIAH-DE insofern unterstützt werden, als das alle in DARIAH-DE

[42] Vgl. https://dev2.dariah.eu/wiki/download/attachments/2295782/DARIAH-DE%20Report%201.3.3%20%E2%80%93%20Provenance.pdf?version=1&modificationDate=1361467022440&api=v2

[43] Vgl. Ebd S. 5

bereitgestellten Services Metadaten gemäß eines festen Datenmodells vergeben und überprüft werden, so dass sowohl das Repository als auch die Collection Registry und der Storage Bereich, mit dem gleichen Standard basierend auf dem gleichen PID System arbeiten.

Als Standards werden in dem Dokument diskutiert: PREMIS, W3C PROV, Open Provenance Model (OPM). OPM wird aufgrund zu dem Zeitpunkt nicht erreichter Spezifikationstiefe bei der Entscheidung außen vorgelassen.

Als Empfehlung bleiben grundsätzlich die Standards: PREMIS und W3C PROV übrig[44]. Welche der beiden Standards präferiert wird, hängt nicht zuletzt auch an persönlichen Einschätzungen der Beteiligten. Tiefere Analysen des Themas – insbesondere auch hinsichtlich eines umfassenden Datenmodells zusammen mit Workflowbeschreibungssprachen und korrespondierenden Ontologien geisteswissenschaftlicher Tools und Methoden – sind noch nicht abgeschlossen. Diese werden in der AG Research Data Lifecycle zur Zeit bearbeitet.

7 Richtlinien (Policies) für den Basis-Research Data Lifecycle

Dieses Kapitel gibt einen Überblick über alle technischen Richtlinien, die für die Verarbeitung von Forschungsdaten in der DARIAH-Infrastruktur und die Implementation eines Workflows relevant sein könnten.

7.1 Identifier

Als persistente Identifier wird in DARIAH-DE vermutlich EPIC 2[45] als Standard Verwendung finden. Wichtig ist hier, dass die DARIAH-Infrastruktur in der Lage ist, bereits vergebene PIDs schon im Ingest zu erkennen, damit eine korrekte Abbildung der Versionsgeschichte in den Metadaten erfolgen kann.

Außerdem hat die Vergabe der EPIC-PIDs auf Dateiebene zu erfolgen. Weitere Komplexitätsebenen gemäß eines Provinienzmodells sind denkbar und wünschenswert.

[44] Ebd. S. 13-14

[45] Vgl. https://github.com/CatchPlus/EPIC-API-v2/wiki/Core-API

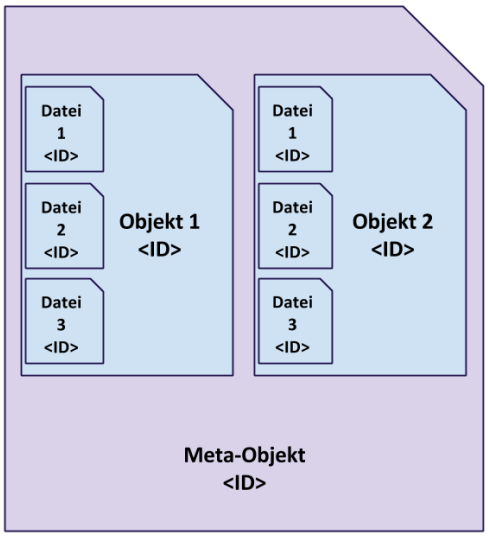

Abb. 12: Identifiervergabe bei komplexen Objekten in den Metadaten aus der AG Research Data LifeCycle

7.2 Prüfsummen

Zur Überprüfung der Datenintegrität nach der Verwendung von Software auf Daten oder aber nach Kopierprozessen empfiehlt sich in einer technischen Infrastruktur die Verwendung von sogenannten Hashwerten[46]. Dabei wird für einen Binärstrom (eine Datei) mittels eines Algorithmus eine Zeichenkette (=ein Hashwert) errechnet, welche zusammen mit dem Binärstrom übetragen und gespeichert wird.

Durch automatische Prozesse, die für die Datei am Zielort oder nach einer für Sicheitsaspekte relevanten Zeitspanne maßgeblich sind, wird mit dem gleichen Algorithmus für den Binärstrom erneut ein Hashwert gebildet und mit dem mitgeschickten / mitgespeicherten Hashwert verglichen. Stimmen beide Hashwerte mit einander überein, ist die der Binärstrom (die Datei) intakt und damit integer. Stimmen beide Hashwerte nicht miteinander überein, hat eine Korruption, d.h. eine bewusste Manipulation oder ein einfacher Bitfehler im Kopierprozess, des Binärstroms (der Datei) statt gefunden.

Um also zu gewährleisten, dass gespeicherte und übertragene Dateiströmen in der DARIAH-DE Infrastruktur nicht verändert wurden, erscheint die Verwendung von Hashwerten oder Prüfsummen sehr sinnvoll. Hier sind verschiedene Standards gleich weit verbreitet. Generell wird aber momentan SHA-1 etwas mehr Sicherheit gegen zufällige Entschlüsselung zugestanden als beispielsweise MD5.

Aus diesem Grund wird für DARIAH-DE die Verwendung von Prüfsummen empfohlen und hier nach Möglichkeit SHA-1 oder neuere SHA-Algorithmen, wie SHA-256 oder SHA-512.

7.3 Verarbeitbare Dateiformate

Eingeliefert werden kann prinzipiell jedes Dateiformat. Hier können keine Einschränkungen gemacht werden, da (digitale) Geisteswissenschaftler mit einer zu hohen Varianz an Dateitypen und -formaten

[46] Vgl. bspw: Nestor: Eine kleine Enzyklopädie der Langzeitarchivierung. Version 2.3. Kapitel 5.3.1 Hashverfahren und Fingerprinting. http://nestor.sub.uni-goettingen.de/handbuch/nestor-handbuch_23.pdf

arbeiten.

Eine erste – nicht erschöpfende – Liste von zu unterstützenden Dateiformaten für Geisteswissenschaftler[47] wurde von der AG RDLC in DARIAH-DE vorgelegt[48]

Das für das Repository vorgeschlagene Tool zur Extraktion technischer Metadaten und Erkennung von Dateiformaten wird JHOVE 2[49] sein. JHOVE 2 erkennt Dateiformate anhand ihres MIME-Type[50]. Existiert für ein Dateiformat kein MIME-Type, wie bspw. bei Containerformaten aus der Microsoft Office-Reihe (DOCX, XLSX, PPTX) wird eine solche Datei im Repository als unbekannt markiert und kann entsprechend von manchen Funktionen, wie eventuellen Migrationsoperationen, nicht bearbeitet werden. Da eine erfolglose Erkennung von Officeformaten natürlich sehr unbefriedigend ist, gilt es hier nach ergänzenden Lösungen zu suchen.

7.4 Datenmodell

Grundsätzlich wird als maximaler Konsens mit minimaler Aussagekraft in DARIAH-DE DublinCore bzw. das Dublin Core Collection Application Profile zur Modellierung von Datensammlungen als Metadatenstandard zu internen Verwaltung verwendet.

Informationen, welche mit DublinCore beschrieben sind, können in der DARIAH-DE Infrastruktur ausgelesen und verarbeitet werden. Hierzu zählen u.a. auch das Feld mit dem Identifier eines Dokuments, im Falle von DARIAH-DE also bevorzugt der EPIC-PID.

Wie im Kapitel 6.2 Datenmodell schon geschildert, reicht zur Administrierung komplexer Dateizusammenhänge und Provinienzmodellierung DublinCore bei weitem nicht aus. Eine Einbettung oder Nachnutzung von DublinCore Metadaten kann und sollte aber in einem dafür geeigneteren Standard (ob PREMIS und W3C PROV) erfolgen.

7.5 Metadaten

Neben diesen schon diskutierten administrativen und oder technisch-strukturellen Metadaten kann auf der Ebene inhaltlicher Metadaten mit der Einreichung von Metadaten gemäß weiterer Standards gerechnet werden.

Eine Übersicht von beobachteten und empfohlenen Metadatenstandards aus den Fachdisziplinen ergibt dabei folgendes Bild[51]:

[47] Vgl. Kapitel 3.2.2 aus dem Papier der AG Research Data LifeCycle, https://docs.google.com/document/d/12tSyZdByWH7I0wb2xGAbh38cw78OezRdjHEGmPIiYIM/edit#, allerdingsbereinigt um kombinierte Formate, wie OCR scannend Bilder(...)

[48] Vgl. https://dev2.dariah.eu/wiki/pages/viewpage.action?pageId=38080370

[49] Vgl. https://bitbucket.org/jhove2/main/wiki/Home

[50] Vgl. http://www.sitepoint.com/web-foundations/mime-types-complete-list/

[51] Die Originalversion dieser Tabelle war Bestandteil der Arbeitsergebnisse von AP 3.2 “Fachspezifische Standards und Empfehlungen” auf https://dev2.dariah.eu/wiki/display/DARIAHDE/Sammlung+der+Metadatenformate

Metadatenformat	Disziplin	Einsatzzweck
TEI Header	Musikwissenschaft	Kodierung der Metadaten von Textdaten für wissenschaftliche Zwecke
TEI Header	Epigraphik/Judaistik	Kodierung der Metadaten von epigraphischen Objekten
TEI Header	Geschichtswissenschaft/Theologie	Kodierung der Metadaten von Textdaten (Druckschriften) für wissenschaftliche Zwecke(Editionen)
TEI Header	Archäologie	ENRICH was the first Arachne project whose TEI-P5-data was harvested via OAI PMH. The ENRICH project is the joint digital manuscript library of the European Union.
Dublin Core Simple	Musikwissenschaft	Kodierung der Metadaten für die XHTML Darstellung
Dublin Core Simplehttps://dev2.dariah.eu/wiki/display/DARIAHDE/Dublin+Core	Archäologie	Standard für OAI Repository
Dublin Core Simplehttps://dev2.dariah.eu/wiki/display/DARIAHDE/Dublin+Core	Geschichtswissenschaft/Theologie	Standard der Quellen (Objekt-) Beschreibung der VRE-IEG, Metadatencodierung
CIDOC-CRM	Archäologie	Kontextualisierung von Objekten zu Topographie, Literatur usw.
MARC21	Archäologie	OPAC Zenon
METS/MODS	Archäologie	Datenaustausch mit Propylaeum
(OAI-PMH		Schnittstelle zum Austausch von Metadaten)
COinS		Kodierung von bibliografischen Metadaten
EAC-CPF		Encoded Archival Context – Corporate Bodies, Persons and Families
MEI Head	Musikwissenschaft	Kodierung der Metadaten von Musiknotation für wissenschaftliche Zwecke
Prometheus	Archäologie, Kunstgeschichte	"Metadatenstandard" für das Bildportal

		Prometheus
ArchaeoML	Archäologie	Ontology used for several excavation documentations in the US
ADeX	Archäologie	Standard developed by the Verband der Landesarchäologen
Midas	Archäologie	Developed in the UK
CARARE	Archäologie	exchange standard developed by project CARARE
LIDO	museum, art, cultural heritage	
Spectrum	museum, art, cultural heritage	
CDWA	museum, art, cultural heritage	
museumdat	museum, art, cultural heritage	

Die hier aufgeführten Metadatenstandards erfüllen in einer Infrastruktur das Ziel von deskriptiven Metadaten, da sie vor allem zum Zwecke der inhaltlichen Strukturierung von fachspezifischen Informationen in den einzelnen Disziplinen eingesetzt werden. Viele der genannten Standards enthalten allerdings auch administrative, d.h. für die generische Abbildung von Forschung relevante Informationen.

Inwiefern diese administrativen Informationen für eine Infrastruktur relevant sind (weil sie beispielsweise Identifier oder andere Dateireferenzen enthalten) und daher ggf. ausgelesen werden müssen, damit ein vollständiges Bild des Forschungsvorhabens maschinell abgebildet werden kann, muss Gegenstand weiterer Diskussionen bleiben.

7.6 Rollen- und Rechtemodell

Im Rahmen von DARIAH-DE wurde in den vergangenen Jahren eine Autorisierungs- und Authentifizierungs-Infrastruktur aufgebaut, die die Vergabe von unterschiedlichen Rollen und Rechten an verschiedene Nutzergruppen – gerade auch für Nutzergruppen aus dem DARIAH-EU Kontext – ermöglicht.[52] Im Fokus des Research Data Lifecycle und auch im Rahmen eines ggf. festzulegenden Datenmodells wird zu klären sein, inwieweit die bereits durchgeführten und zukünftig geplanten Arbeiten an der DARIAH-DE AAI ausreichen, um die hieraus spezifischen Anforderungen zu erfüllen.

[52] Eine technische Beschreibung der DARIAH-DE AAI findet sich u.a. auf der DARIAH-DE Webseite: https://de.dariah.eu/aai