



Fachkonzept

Konzept Bestandserhaltung

Version:	1.0
Status:	Veröffentlichte Fassung
Vertraulichkeit:	öffentlich
Stand:	23.06.2014

Inhaltsverzeichnis

	Dokumentenlenkung.....	3
	Referenzdokumente.....	4
1	Einführung.....	5
1.1	Gegenstand des Dokumentes.....	5
1.2	Aufbau des Dokumentes	5
2	Voraussetzungen und Umfeld	6
2.1	Funktionseinheit Bestandserhaltung.....	6
2.2	Ausgangssituation und Voraussetzungen in M-V.....	6
3	Anforderungen.....	8
3.1	Abgrenzung des Systemkontextes.....	8
3.2	Anforderungen an die Funktionseinheit Bestandserhaltung	9
4	Entwurf der Systembestandteile	13
4.1	Lösungsidee	13
4.2	Logische Sicht/Szenarien	13
4.3	Aufbau des Systems.....	16
4.3.1	Komponente „Dateiformat-Erkennen“	17
4.3.2	Komponente „Datenobjekt-Validator“	18
4.3.3	Komponenten „Datenobjekt-Transformator“	19
4.4	Laufzeitsicht.....	19
4.4.1	Dateiformate erkennen.....	20
4.4.2	Datenobjekte validieren	22
4.4.3	Datenobjekte transformieren	23
	Abbildungsverzeichnis.....	26

Dokumentenlenkung

Dokumenteninformationen

Dokumententyp:	Konzept
Dokumententitel:	Konzept Bestandserhaltung
Version:	1.0
Dateiname:	Konzept_Bestandserhaltung.docx
Dokumenteneigentümer:	LAKD – Landesarchiv Mecklenburg-Vorpommern
Ablageort:	[https://teamportale.mvnet.de/pz/elamv/Projektakte%20eLA/Fachkonzepte/Konzept_Bestandserhaltung.docx]
Status:	Veröffentlichte Fassung
Vertraulichkeitsstufe:	öffentlich
Verteiler:	
Es tritt außer Kraft:	[xx/yyyy]
Nächste Prüfung:	am 30.06.2015
Ansprechpartner (opt.):	

Änderungskontrolle (ab Version 0.1)

Version	Datum	Bearbeiter	Status	Änderungsgrund	Seiten
0.1	21.10.2013	M. Marten	In Bearbeitung	Erstellung des Konzeptgerüsts	
0.4	23.04.2014	J. Lehmann, R. Lorenz, M. Schmitz	In Bearbeitung	Entwurf	
0.5	23.04.2014	M. Marten	Zur Review	Entwurf für Reviewgruppe	
0.6	30.04.2014	J. Lehmann, M. Marten, M. Schmitz	Zur Abnahme	Einarbeitung Korrekturen der QS	
1.0	20.06.2014	M. Marten	Veröffentlichte Fassung	Einarbeitung Korrekturen der LG	

Referenzdokumente

Dieser Abschnitt führt alle für die Bearbeitung und das Verständnis des Produkts erforderlichen Dokumente an. Die Dokumente sollten hinsichtlich ihrer Verwendung (intern, extern) unterschieden werden. Über die referenzierten Dokumente sind folgende Informationen zu halten: Bezeichnung, Identifikation mit Versionsangabe und Art der Verwendung (z.B. Quelle, weiterführende Literatur)

Dokument	Version	Ablageort
„Formaterkennung und Formatvalidierung Theorie und Praxis“ von Claire Röthlisberger-Jourdan aus „Digitale Archivierung in der Praxis“		https://teamportale.mvnet.de/pz/elamv/Projektakte%20eLA/Quellen/Formaterkennung-Validierung/R%C3%B6thlisberger-Jourdan,%20Formaterkennung%20und%20Validierung.pdf

1 EINFÜHRUNG

1.1 Gegenstand des Dokumentes

Das Konzept Bestandserhaltung bildet die Grundlage für die Erhaltung der Darstellungsfähigkeit des elektronischen Archivguts über lange Zeiträume hinweg. Mit den Anforderungen werden die Rahmenbedingungen festgelegt, mit denen der Zugriff und die Darstellung des AIP-Inhaltes jederzeit möglich sein soll. Im vorliegenden Konzept werden ausschließlich die technischen Mittel beschrieben, nicht die organisatorischen Aufgaben und Prozesse, um die im eLA M-V gespeicherten Informationen langfristig verfügbar und verstehbar zu erhalten. Die organisatorischen Vorgaben sind im Konzept Erhaltungsplanung festgehalten. Komponenten zur authentischen Darstellung des elektronischen Archivguts sind Bestandteil des Zugriffskonzeptes.

1.2 Aufbau des Dokumentes

Das Dokument beschreibt in Kapitel 2 die Grundlagen der Bestandserhaltung und stellt den Einführungsstand geeigneter Software in Mecklenburg-Vorpommern dar. Die Anforderungen an die identifizierten Systembestandteile werden in Kapitel 3 aufgeführt. In Kapitel 4 werden auf Basis der ermittelten Anforderungen eine Lösungsidee entwickelt, die Anwendungsfälle beschrieben sowie der Aufbau des Systems erläutert.

2 VORAUSSETZUNGEN UND UMFELD

2.1 Funktionseinheit Bestandserhaltung

Bestandserhaltung ist keine Funktionseinheit entsprechend der ISO-Norm 14721:2012, sondern eine durch das LAKD – Landesarchiv M-V definierte Funktionseinheit. Sie wurde eingeführt, damit die Software, die zur Umsetzung der Anforderungen an bestandserhaltende Maßnahmen der OAIS-Funktionseinheiten Erhaltungsplanung, Übernahme und Zugriff erforderlich ist, zusammenhängend als funktionale Einheit konzipiert werden kann.

Die Funktionseinheit Bestandserhaltung verwaltet in der Funktionseinheit Datenverwaltung alle Einheiten und Informationen, die zur Erhaltung der Darstellungsfähigkeit des elektronischen Archivguts über lange Zeiträume notwendig sind. Außerdem werden die Informationen zu allen im eLA M-V gespeicherten Primärdokumentformaten verwaltet (z.B. Formatname, Version, PRONOM-ID). Zu jedem Formateintrag sollte mindestens ein für das eLA M-V möglicher Darstellungspfad verzeichnet sein.

Alle Einheiten zur Dokumentation der Daten in einem AIP (z.B. XML-Schema-xsd für die Metadaten) sollten über das Modul verwaltet werden und referenzierbar sein. Das heißt, es muss zu jedem AIP der Zugriff auf die passenden Datendefinitionen möglich sein, um die Darstellung eines AIP-Inhaltes jederzeit gewährleisten zu können (z.B. bei der Archivierung von Datenexporten aus Fachverfahren im XML-Format).

2.2 Ausgangssituation und Voraussetzungen in M-V

Im LAKD – Landesarchiv ist derzeit keine Software zur Bestandserhaltung (Dateiformaterkennung, Validierung und Transformation von Datenobjekten) vorhanden.

Synergieeffekte können sich bei der Zusammenarbeit mit dem Competence Center Domea im Rahmen der Mitnutzung der Software-Werkzeuge Adobe Preflight zur stichprobenartigen Validierung elektronischer Dokumente (ausschließlich PDF/X, PDF/A und PDF/E) sowie Adobe® Lifecycle ES 4 Version 11.0 mit dem Modul Adobe LiveCycle PDF Generator zur Formattransformation ergeben. Beide Werkzeuge werden im Rahmen der Einführung des Dokumentenmanagement- und Vorgangsbearbeitungssystem OpenText DOMEA® von den oberen Landesbehörden genutzt.

Der zentrale Dienst Langzeitspeicherung, mit dem elektronische Dokumente für die Dauer der gesetzlichen Aufbewahrungsfrist in der Behörde gespeichert werden, setzt zur Zeit keine Softwarekomponenten für die Formaterkennung, -validierung oder -transformation ein, wird diese aber perspektivisch ebenfalls benötigen. Mit Fortschritt im Aufbau des eLA M-V werden entsprechende Abstimmungen mit dem Verfahrenseigner, dem Ministerium für Inneres und Sport Mecklenburg-Vorpommern, zur gemeinsamen Nutzung derartiger

Softwarekomponenten als sinnvoll erachtet.

3 ANFORDERUNGEN

3.1 Abgrenzung des Systemkontextes

Der Betrachtungsgegenstand für die Anforderungsanalyse im vorliegenden Konzept wurde, wie in Abbildung 1 dargestellt, auf die Funktionseinheit Bestandserhaltung (weiß hinterlegt) eingeschränkt. Es werden ausschließlich Anforderungen an Software zur Erhaltung von übernommenen elektronischen Akten aus dem DMS/VBS des Landes Mecklenburg-Vorpommern erhoben.

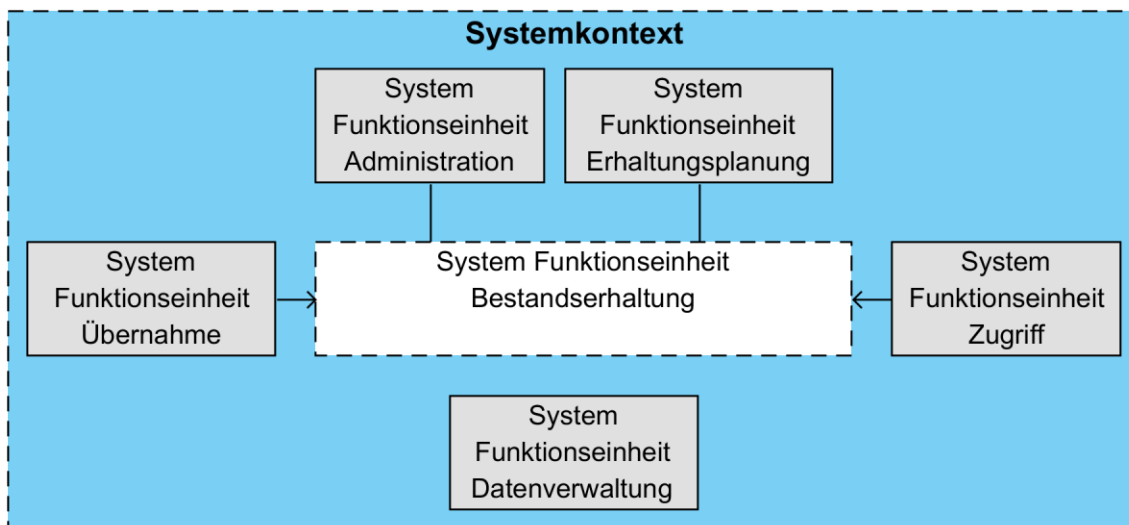


Abbildung 1 Abgrenzung des Systemkontextes für das vorliegende Konzept

Die Bestandserhaltung übergibt Informationen zu Datenobjekten und konvertierte Datenobjekte an die Funktionseinheiten Übernahme und Zugriff. Die Funktionseinheit Übernahme übermittelt ausgewählte Informationen an die Datenverwaltung. Auch wenn die Funktionseinheit Bestandserhaltung nicht selbständig Informationen in der Funktionseinheit Datenverwaltung ablegt, ist das System doch zur Erhebung von Anforderungen relevant und daher im Kontext dargestellt. In direkter Beziehung zur Bestandserhaltung stehen die Funktionseinheit Übernahme, Zugriff, Administration und Erhaltungsplanung. Sie nutzen die Softwarekomponenten der Bestandserhaltung

- zur Erkennung von Dateiformaten,
- zur Validierung von Datenobjekten und
- zur Transformation von Datenobjekten.

3.2 Anforderungen an die Funktionseinheit Bestandserhaltung

Nachfolgend werden die Anforderungen an die Funktionseinheit Bestandserhaltung tabellarisch dokumentiert. Als Anforderungsquellen wurden die ISO-Norm 14721:2012 und der Metadatenstandard PREMIS verwendet. Aus den DIN-Normen 31644, 31645, 31646 und 31647 (Entwurf) und dem Metadatenstandard XDOMEA konnten keine Anforderungen ermittelt werden. Zusätzliche Anforderungen wurden vom LAKD – Landeshauptarchiv auf Basis der Erfahrungen anderer Archive und Veröffentlichungen der Koordinationsstelle für dauerhafte Archivierung elektronischer Unterlagen (KOST) erhoben.

Die Anforderungen sind in der Tabelle nach den drei Hauptfunktionen der Bestandserhaltung „Erkennung von Dateiformaten“, „Validierung von Datenobjekten“ und „Transformation von Datenobjekten“ gruppiert.

ID	Art	Anforderungsbeschreibung	MSK	Quellen	Kommentar
Req-4.1-001	nfA	Das System muss batchtauglich sein.	M	Röthlisberger-Jourdan (R-J)	Einbindung in automatisierten Prozess.
Req-4.1-002	nfA	Das System soll funktionale, synchrone Schnittstellen zur Integration der Funktionalität in andere Komponenten bieten	S	LA	
Req-4.1-003	nfA	Das System muss fähig sein, ohne Administratorenrechte in Betrieb genommen zu werden.	M	R-J	
Erkennung von Dateiformaten					
Req-4.1-004	fA	Das System soll fähig sein, die jeweils gültigen Übernahmeformate und archivischen Zielformate von Datenobjekten zu erkennen.	S	LA	
Req-4.1-005	fA	Das System muss fähig sein, die im Konzept Betriebsorganisation genannten Übernahmeformate der Übernahme zu erkennen.	M	LA	Konzept Betriebsorganisation, Kapitel 4.5.2
Req-4.1-006	fA	Das System muss fähig sein, die im Konzept Betriebsorganisation genannten archivischen Zielformate zu erkennen.	M	LA	Konzept Betriebsorganisation, Kapitel 4.5.3
Req-4.1-007	fA	Das System muss fähig sein, bei nicht unterstützten Dateiformaten eine automatisiert-auswertbare Fehlermeldung auszugeben.	M	LA	

Req-4.1-008	nfA	Das System soll fähig sein, Formate schnell zu erkennen. Genauigkeit der Erkennung hat Priorität vor Schnelligkeit.	S	R-J	
Req-4.1-009	fA	Das System muss fähig sein, eine Format-Datenbank zu verwenden, die öffentlich zugänglich und editierbar ist.	M	R-J	bspw. Pronom
Req-4.1-023	fA	Das System muss fähig sein, folgende PREMIS-Metadaten nach PREMIS Data Dictionary Version 2.0 für ein Datenobjekt zu ermitteln: ■ format □ formatDesignation - formatName - formatVersion □ formatRegistry - formatRegistryName - formatRegistryKey - formatRegistryRole □ formatNote	M	LA, PREMIS	
Validierung von Datenobjekten					
Req-4.1-010	fA	Das System soll fähig sein, Datenobjekte zu validieren. Ein Datenobjekt ist valide, wenn alle Eigenschaften der Format-Spezifikation erfüllt sind.	S	R-J	
Req-4.1-011	fA	Das System muss fähig sein, die im Konzept Betriebsorganisation genannten Übernahmeformate der Übernahme zu validieren.	M	LA	Konzept Betriebsorganisation, Kapitel 4.5.2
Req-4.1-012	fA	Das System muss fähig sein, die im Konzept Betriebsorganisation genannten archivischen Zielformate zu validieren.	M	LA	Konzept Betriebsorganisation, Kapitel 4.5.3
Req-4.1-013	nfA	Das System muss fähig sein, ein PDF/A-Dokument von einem PDF-Dokument zu unterscheiden (feine Granularität).	M	R-J	
Req-4.1-015	nfA	Das System soll fähig sein, Formate schnell zu validieren. Qualität der Validierung hat Priorität vor Schnelligkeit.	S	R-J	
Req-4.1-016	fA	Das System soll fähig sein, die Nachvollziehbarkeit der Validierung zu gewährleisten.	S	R-J	bspw. durch offen gelegten Quelltext

Req-4.1-017	fA	Das System muss fähig sein, Repräsentationsinformationen für ein positiv validiertes Datenobjekt zu erstellen.	M	ISO 14721:2012	
Req-4.1-018	fA	Das System muss fähig sein, folgende PREMIS-Metadaten nach PREMIS Data Dictionary Version 2.0 für ein Datenobjekt zu ermitteln: ■ significantProperties (ausschließlich page count) □ significantPropertiesType □ significantPropertiesValue □ significantPropertiesExtension ■ fixity □ messageDigestAlgorithm □ messageDigest □ messageDigestOriginator ■ size	M	LA, PREMIS	
Transformation von Datenobjekten					
Req-4.1-019	fA	Das System soll fähig sein, Datenobjekte, die in Übernahmeformaten vorliegen, in archivische Zielformate zu transformieren.	S	LA	
Req-4.1-020	fA	Das System muss fähig sein, die im Konzept Betriebsorganisation genannten Übernahmeformate der Übernahme die im Konzept Betriebsorganisation genannten archivischen Zielformate zu transformieren. Die signifikanten Eigenschaften müssen erhalten bleiben.	M	LA	Konzept Betriebsorganisation, Kapitel 4.5.2 & 4.5.3
Req-4.1-021	fA	Das System muss fähig sein, bei Transformationsfehlern eine automatisiert-auswertbare Fehlermeldung auszugeben.	M	LA	
Req-4.1-022		Das System muss fähig sein, folgende PREMIS-Metadaten nach PREMIS Data Dictionary Version 2.0 für das erfolgreich transformierte Datenobjekt zu erzeugen: ■ creatingApplication □ creatingApplicationName □ creatingApplicationVersion □ dateCreatedByApplication □ creatingApplicationExtension	M	LA, PREMIS	
Req-4.1-		Das System muss den Transformationsvorgang durch ein Protokoll nachvollziehbar machen.	M	LA, PREMIS	

023					
-----	--	--	--	--	--

4 ENTWURF DER SYSTEMBESTANDTEILE

In den folgenden Kapiteln wird aus den Anforderungen aus Kapitel 3 der Entwurf des Systems abgeleitet.

4.1 Lösungsidee

Kernaufgabe der Softwarekomponenten der Bestandserhaltung ist die Erkennung von Dateiformaten, die Validierung und Transformation von Datenobjekten. Die Komponenten werden vornehmlich als Hintergrundsysteme durch die Funktionseinheit Übernahme verwendet. Es ist außerdem ein operativer Einsatz der Komponenten durch einen Mitarbeiter des Archivs im Rahmen der Administration, der Erhaltungsplanung und des Zugriffs denkbar. Daher müssen die Softwarekomponenten sowohl in andere Softwarekomponenten integrierbar sein, als auch eine grafische oder kommandozeilenbasierte Benutzerschnittstelle bieten. Die Benutzerschnittstelle darf nur durch die Mitarbeiter des Landesarchivs nutzbar sein.

Die Nutzung erfolgt ausschließlich in der Organisation des LAKD – Landesarchiv durch folgende Akteure:

- Administrator (Vertreter des Bereichs Fachadministration)
- VS-Beauftragter
- DV-Beauftragter
- Archivar

Zur Integration von Softwarekomponenten der Bestandserhaltung in Komponenten der Übernahme und des Zugriffs sind funktionale, synchrone Schnittstellen wünschenswert. Da in der Bestandserhaltung vornehmlich bereits existierende Softwarekomponenten zum Einsatz kommen sollen, können, sofern keine funktionalen Schnittstellen existieren, auch die Datenschnittstellen der Tools verwendet werden. Die Softwarekomponenten sollen die zur eigenen Funktionsfähigkeit notwendigen Daten selbständig verwalten. Dateiformat- und Validierungsinformation sowie Metadaten einer durchgeführten Formattransformation, die langfristig gespeichert werden sollen, sind persistent durch die Funktionseinheit Übernahme in der Funktionseinheit Datenverwaltung und im AIP abzulegen.

4.2 Logische Sicht/Szenarien

Das Use-Case-Diagramm in Abbildung 2 zeigt die Use-Cases der Funktionseinheit Bestandserhaltung mit der Beziehung zu den Akteuren und anderen Funktionseinheiten. Die dargestellten Use-Cases werden nachfolgend in weitere Use-Cases detailliert, die aus den Anforderungen abgeleitet wurden. Die nicht funktionalen Anforderungen Req-4.1-001, Req-4.1-002, Req-4.1-003 und Req-4.1-015 können nicht in einem Use-Case dargestellt werden. In den Funktionseinheiten Zugriff, Administration und Erhaltungsplanung existieren aktuell

keine benannten Use-Cases, die die Funktionalitäten der Bestandserhaltung nutzen.

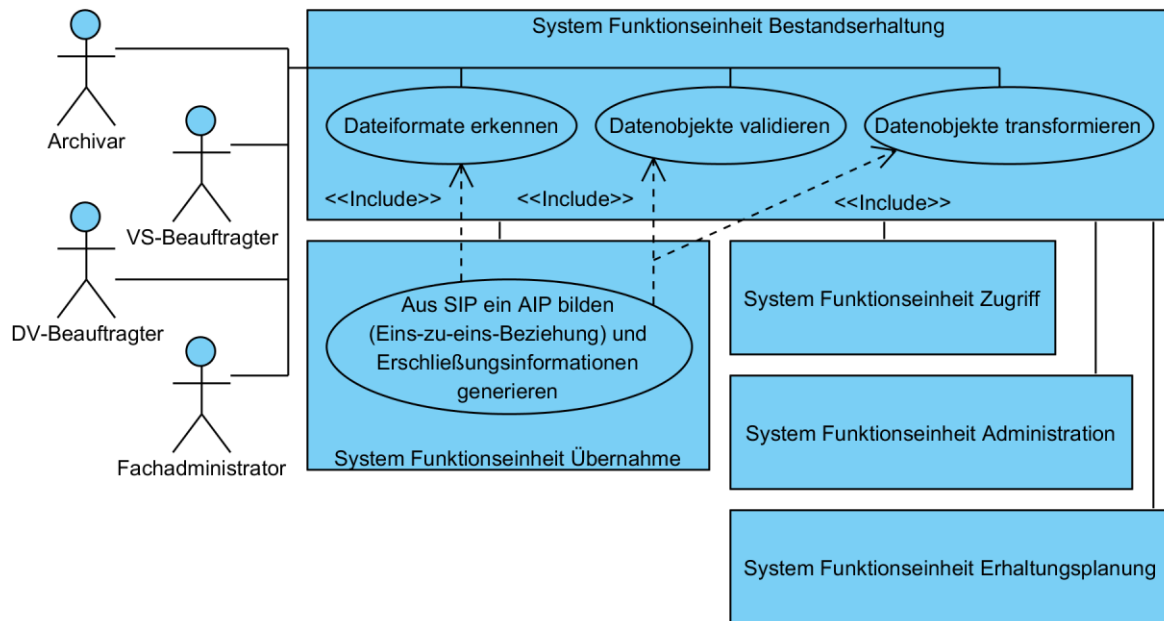


Abbildung 2 Logische Sicht auf die Funktionseinheit Bestandserhaltung

Der Use-Case „Dateiformate erkennen“ kann in folgende weitere Use-Cases konkretisiert werden.

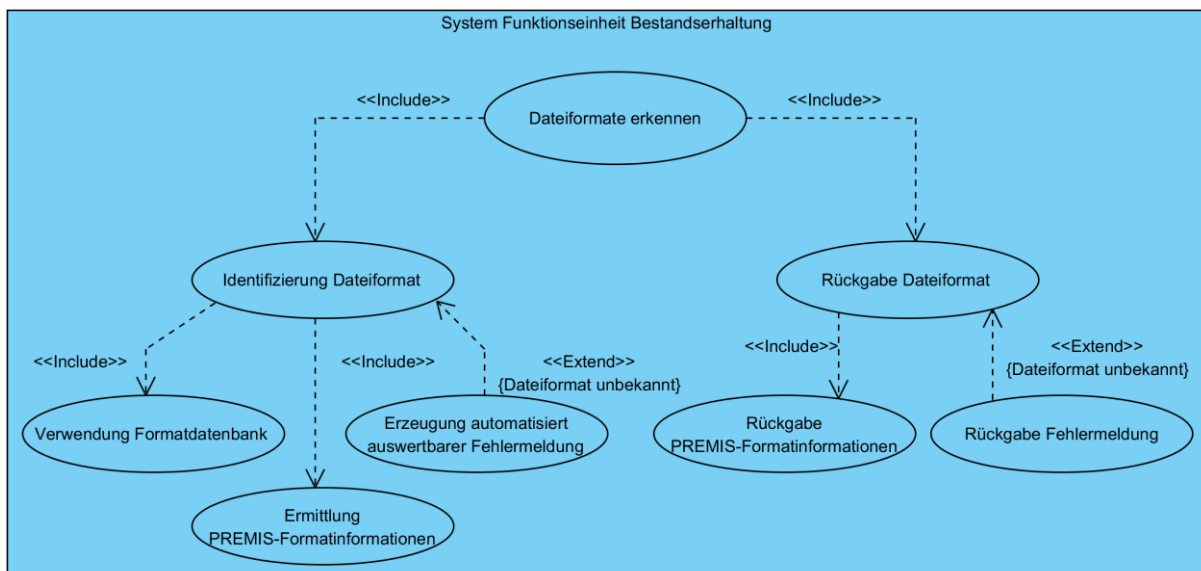


Abbildung 3 Logische Sicht auf den Use-Case "Dateiformat erkennen"

Mit den in Abbildung 3 Logische Sicht auf den Use-Case "Dateiformat erkennen" dargestellten Anwendungsfällen werden die funktionalen Anforderungen Req-4.1-004, Req-4.1-005, Req-4.1-006, Req-4.1-007, Req-4.1-009 und Req-4.1-023 umgesetzt. Die nicht funktionale Anforderung Req-4.1-008 kann nicht in einem Use-Case dargestellt werden.

Zur Validierung von Datenobjekten lassen sich aus den Anforderungen die in Abbildung 4 dargestellten Use-Cases ableiten.

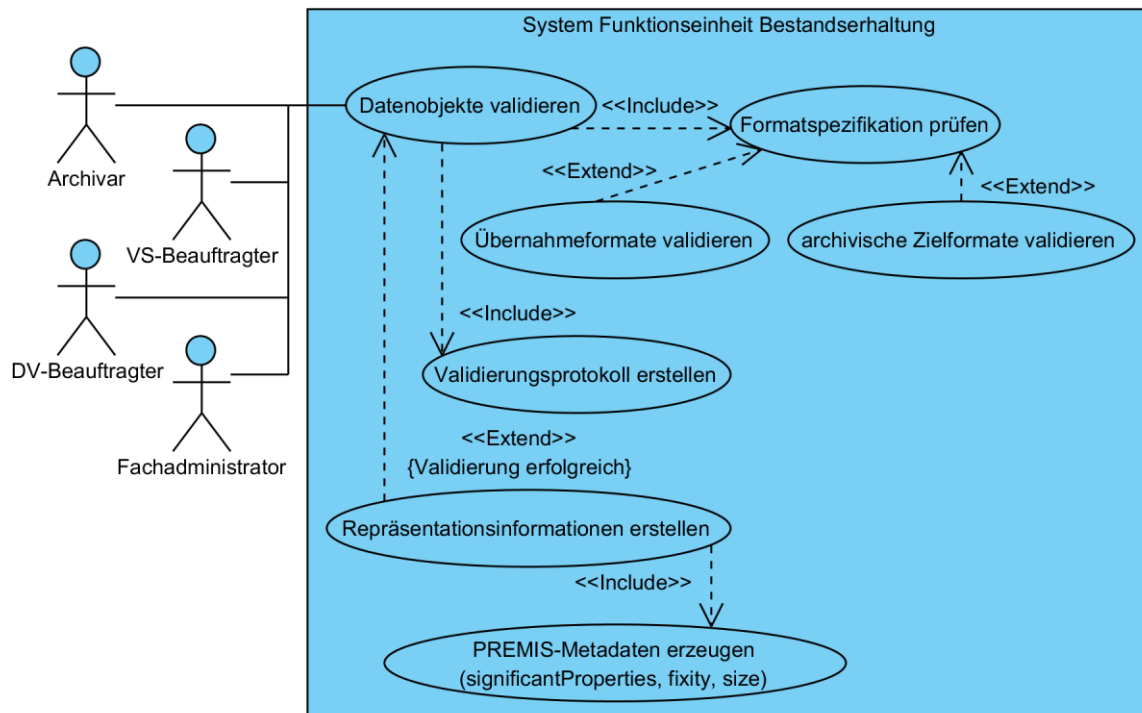


Abbildung 4 Logische Sicht auf den Use-Case "Datenobjekte validieren"

Mit den Use-Cases werden die Anforderungen Req-4.1-010, Req-4.1-011, Req-4.1-012, Req-4.1-017 und Req-4.1-018 umgesetzt. Die Anforderungen Req-4.1-013 und Req-4.1-015 sind Qualitätsanforderungen an den Use-Case „Formatspezifikation prüfen“, die sich nicht als eigener Use-Case darstellen lassen. Zur Umsetzung der Nachvollziehbarkeit der Validierung (Anforderung Req-4.1-016) wird ein Use-Case zur Erstellung eines Validierungsprotokolls definiert. Sofern die Validierung erfolgreich war, müssen Repräsentationsinformationen (u.a. im PREMIS-Format) erzeugt werden.

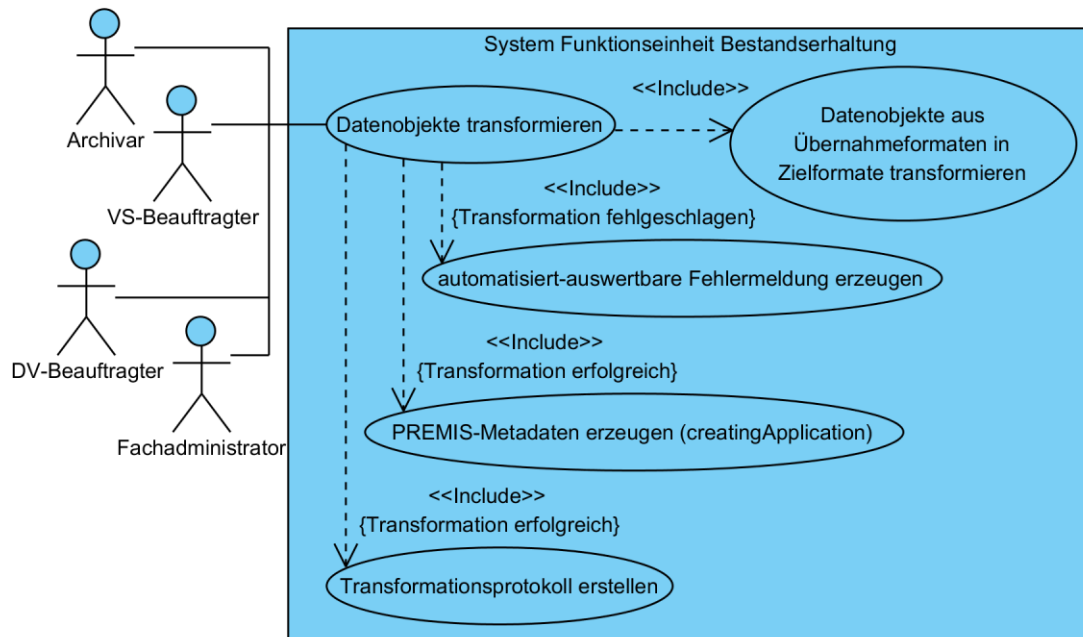


Abbildung 5 Logische Sicht auf den Use-Case "Datenobjekte transformieren"

Abbildung 5 beschreibt den Use-Case zur Transformation der Datenobjekte detaillierter und setzt die Anforderungen Req-4.1-019, Req-4.1-020, Req-4.1-021, Req-4.1-022 sowie Req-4.1-023 um. Die Erhaltung der signifikanten Eigenschaften nach Req-4.1-020 ist eine Qualitätsanforderung an den Use-Case „Datenobjekte aus Übernahmeformaten in Zielformate transformieren“.

4.3 Aufbau des Systems

Aufbauend auf den Anwendungsfällen, die aus den Anforderungen abgeleitet wurden, wird nachfolgend der Aufbau der Funktionseinheit Bestandserhaltung in Form eines UML-Komponentendiagramms beschrieben. Das Diagramm stellt die Struktur und Zusammenhänge zwischen den Komponenten dar. Abbildung 6 zeigt die Funktionseinheit Bestandserhaltung in der Blackbox-Darstellung, in der die drei Hauptfunktionen als Schnittstelle der Bestandserhaltung dargestellt werden. Die Schnittstelle kann bei Bedarf von anderen Funktionseinheiten genutzt werden. Die Komponente Formatdatenbank sowie die zugehörige Schnittstelle zum Abruf von Formatinformationen (siehe Anforderung Req-4.1-009) sind ebenfalls aufgeführt.

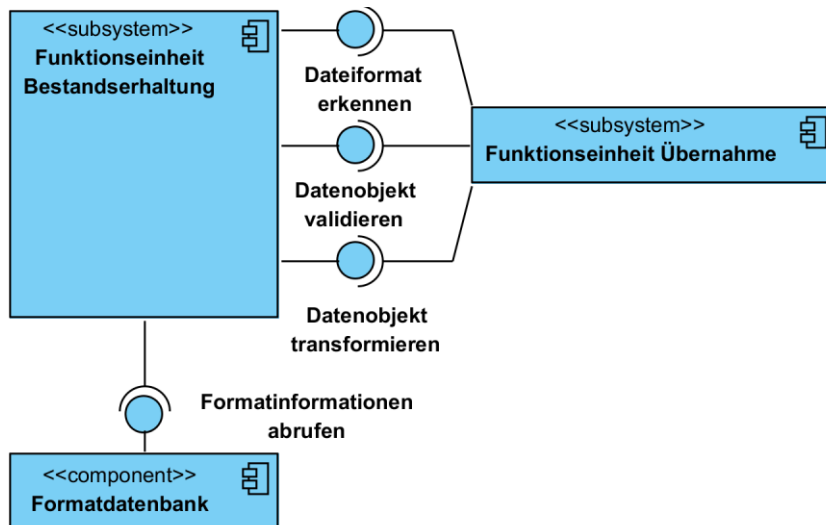


Abbildung 6 Blackbox-Darstellung für die Funktionseinheit Bestandserhaltung

Die Whitebox-Darstellung in Abbildung 7 zeigt den inneren Aufbau der Funktionseinheit Bestandserhaltung. Die Funktionseinheit enthält 1-n Dateiformat-Erkennen, Datenobjekt-Validatoren und Dateiformat-Transformatoren, die für die Behandlung der unterschiedlichen Formate der Datenobjekte zum Einsatz kommen. Koordiniert aufgerufen werden die Komponenten von der Prozesssteuerung, die auch zusätzliche Informationen, etwa Protokollinformationen oder PREMIS Metadaten (creatingApplication), erzeugt.

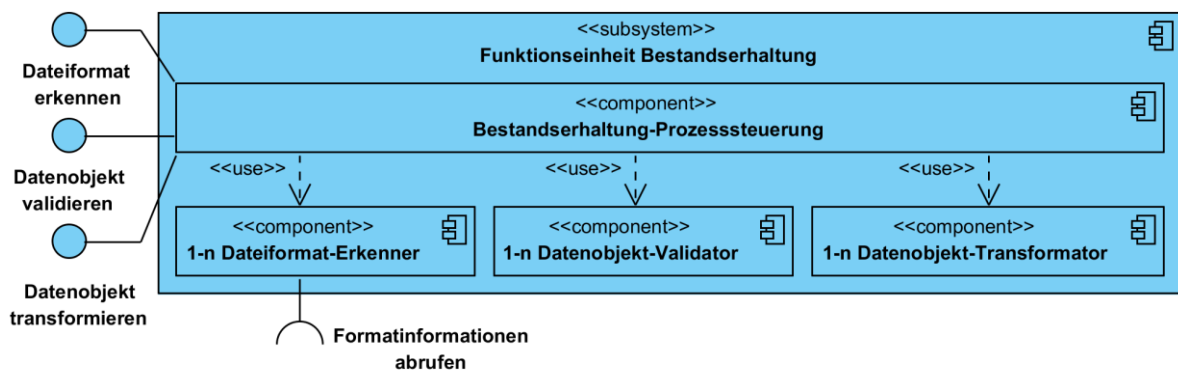


Abbildung 7 Whitebox-Darstellung der Funktionseinheit Bestandserhaltung

4.3.1 Komponente „Dateiformat-Erkennen“

Als Komponenten zur Erkennung von Dateiformaten kommen in der elektronischen Archivierung häufig folgende Tools zum Einsatz:

- DROID (Digital Record Object Identification) – Software zur automatischen Identifikation von Dateiformaten basierend auf der PRONOM-Datenbank

(Formatdatenbank). Entwickelt wurde das Tool durch The National Archives. Der Aufruf erfolgt über eine grafische oder zeichenorientierte Schnittstelle. DROID wird kostenfrei (BSD-Lizenz für DROID v4.0) unter <http://droid.sourceforge.net/> (zuletzt eingesehen am 24.03.2014) zum Download zur Verfügung gestellt.

- File – UNIX-Programm zur Identifikation von Dateiformaten basierend auf der Formatsammlung magic numbers (ASCII-Textdatei). Das Programm steht unter <ftp://ftp.astron.com/pub/file/> (zuletzt eingesehen am 24.03.2014) separat zum Download zur Verfügung, ist aber auch Bestandteil von UNIX-basierten Betriebssystem und kann mit der Nutzungslizenz für das Betriebssystem verwendet werden.
- Fido (Format Identifier for Digital Objects) – Tool (CLI-Tool) zur automatischen Formaterkennung unter Windows und Linux. Greift auf die PRONOM-Signaturen zurück, wandelt diese in reguläre Ausdrücke und wendet sie anschließend an. Fido ist kostenfrei (Apache 2.0 Lizenz) unter <https://github.com/openplanets/fido> (zuletzt eingesehen am 24.03.2014) erhältlich.
- Tika (Apache Tika™ Toolkit) – Tika erkennt und extrahiert Metadaten und strukturierte Text-Inhalte aus verschiedenen Dokumenten in einer vorhandenen Parser-Bibliothek. Für die Formaterkennung wird der MimeMagic Detection Teil verwendet. Tika steht kostenfrei (Apache 2.0 Lizenz) unter <https://tika.apache.org/download.html> (zuletzt eingesehen am 24.03.2014) zum Download zur Verfügung.

4.3.2 Komponente „Datenobjekt-Validator“

Für die unterschiedlichen Datenformate, unabhängig ob Übernahmeformat oder archivisches Zielformat, muss ein Validator mit festgelegter Konfiguration bestimmt werden. Dazu müssen die unterschiedlichen, erhältlichen Validatoren geprüft und die Konfiguration dokumentiert werden. Die schweizer Koordinierungsstelle für die dauerhafte Archivierung elektronischer Unterlagen (KOST) hat in einer Studie verschiedene PDF/A-Validatoren einer genaueren Prüfung unterzogen. Dabei bezieht sie sich auf Vorarbeiten der Firma PDFlib. Ein Übersicht der getesteten Produkte und der Testergebnisse wird in der folgenden Tabelle dargestellt.

Eigenschaft/ Produkt	getestete Version	getestet durch/ im Jahr	Geschwindigkeit & Robustheit	Genauigkeit
Adobe Acrobat	9.1.0	PDFlib/ 2009	Ausreichend	Gut
Callas pdfaPilot	1.2.077	PDFlib/ 2009	Ausreichend	Gut
Intarsys PDF/A Live	5.0.4	KOST/ 2010	Gut	Sehr gut
PDF Tools 3Heights PDF	1.8.32.1	PDFlib/ 2009	Sehr Gut	Gut

Validator Shell				
Seal Systems PDF Longfile Suite/ PDF-Checker	2.1.1.2	PDFlib/ 2009	Gut	Ausreichend
Solid Documents Solid Framework	5.1.168	PDFlib/ 2009	Ausreichend	Mangelhaft
PDFTron PDF/A Manager	1.00 (CLI)	KOST/ 2010	Sehr gut	Sehr gut

Geschwindigkeit & Robustheit (Sehr gut = < 1 Minute & ohne Absturz; Gut = 1 bis 5 Minuten & ohne Absturz; Ausreichend = > 5 Minuten & ohne Absturz; Mangelhaft = Absturz) Genauigkeit (Sehr gut = Mittelwert > = 95 %; Gut = Mittelwert 90 % bis 94 %; Ausreichend = Mittelwert 75 % bis 89 %; Mangelhaft = Mittelwert < 75 %)

4.3.3 Komponenten „Datenobjekt-Transformator“

Für die unterschiedlichen Datenformate muss analog zum Validator ein präferiertes Transformationstool festgelegt werden. Dazu müssen die unterschiedlichen, erhältlichen Transformatoren für die unterschiedlichen Datenformate geprüft und die Konfiguration, mit der das beste Transformationsergebnis aus archivischer Sicht erzielt wird, dokumentiert werden. Diese Konfiguration ist dann im Rahmen der Bestandserhaltung zu verwenden und regelmäßig zu prüfen. Auch hier hat die KOST im Rahmen einer Studie zu PDF/A-Konvertern folgende Produkte näher untersucht:

- dmstools Rendering Service von dmstools (<http://www.dmstools.de>) – Kostenpflichtige Software zur Konvertierung nach TIFF, PDF und PDF/A.
- PDF/A Live! von intarsys (<http://www.intarsys.de>) – Kostenpflichtige Software zur Konvertierung von PDF- und Bilddateien nach PDF/A.
- LuraDocument PDF Compressor Enterprise mit Born Digital Module von LuraTech (<http://www.luratech.com>) - Kostenpflichtige Software zu Konvertierung nach PDF/A.
- PDFCreator 0.9.9 von pdfforge (<http://www.pdfforge.org>) – Open Source Software zur Konvertierung nach PDF/A, PDF, TIFF, PNG, JPG, BMP, PCX, PS und EPS.
- 3-Heights Document Converter Service von PDF Tools (<http://www.pdf-tools.com>) – Kostenpflichtige Software zur Konvertierung nach TIFF, PDF und PDF/A.
- PDF/A Manager von PDFTron (<http://www.pdftron.com>) – Kostenpflichtige Software zur Konvertierung von PDF-Dateien in PDF/A.

Weitere Informationen (u.a. der Preis) sind der KOST-Studie zu entnehmen.

4.4 Laufzeitsicht

In den nachfolgend dargestellten Sequenzdiagrammen werden die Aktivitäten und deren Abfolge der drei Use-Cases „Datenformate erkennen“, „Datenobjekte validieren“ und „Datenobjekte transformieren“ dargestellt. Die in den Sequenzdiagrammen beschriebenen Aktivitäten müssen vollständig abgeschlossen sein, bevor eine neue Aktivität beginnt. Parallelitäten von Aktivitäten sind explizit dargestellt.

4.4.1 Dateiformate erkennen

Für das Erkennen des Dateiformates ist grundsätzlich folgender Ablauf vorgesehen, der primär durch Aufruf einer automatisierten Schnittstelle durch die Funktionseinheit Übernahme gestartet wird.

1. Die Erkennung des Dateiformates beginnt mit dem Empfangen eines Datenobjektes.
2. Das empfangene Datenobjekt wird sukzessive mit allen in der Funktionseinheit Bestandsverwaltung vorhandenen und aktivierten Formaterkennungswerkzeugen (1-n Dateiformat-Erkennen) verarbeitet. Hierbei kann zusätzlich zu der eigentlichen Formaterkennung durch das jeweilige Werkzeug ebenfalls eine Abfrage bei einer öffentlichen Formatdatenbank erfolgen. Diese Abfrage muss durch mindestens einen der verwendeten Format-Erkennen durchgeführt werden.
3. Im Anschluss eines Formaterkennungsdurchlaufs wird das ermittelte Ergebnis mit den bereits vorhandenen Ergebnissen vorheriger Durchläufe auf Widersprüchlichkeit geprüft. Werden in diesem Zusammenhang widersprüchliche Angaben festgestellt, wird eine strukturierte und automatisiert auswertbare Fehlermeldung generiert und der Prozess der Dateiformaterkennung mit Rückgabe der Fehlermeldung abgebrochen. Sofern die Angaben widerspruchsfrei sind, werden die PREMIS-Formatinformationen extrahiert und die Ergebnisse des aktuellen Erkennungsdurchlaufs mit den bisherigen Ergebnissen zusammengeführt.
4. Nach Verarbeitung des empfangenen Datenobjektes durch alle Dateiformat-Erkennen wird das Ergebnis der Erkennungsdurchläufe einschließlich der ermittelten PREMIS-Formatinformationen an die aufrufende Funktionseinheit zurückgegeben und der Prozess der Formaterkennung beendet.

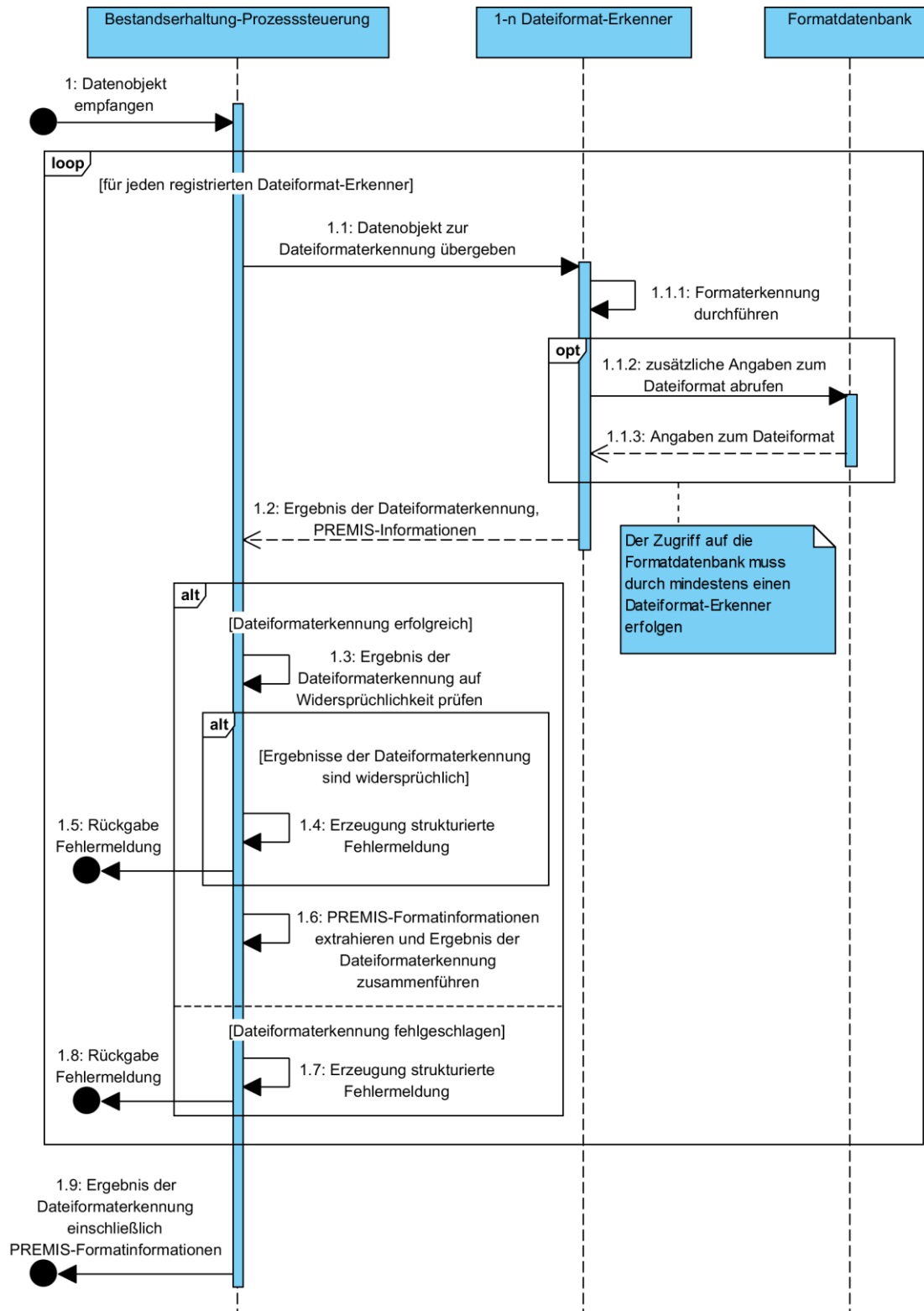


Abbildung 8 Sequenzdiagramm für den Use-Case „Dateiformat erkennen“

4.4.2 Datenobjekte validieren

Für den Use-Case „Datenobjekte validieren“ ist folgender Ablauf vorgesehen, der durch die Akteure Archivar, VS-Beauftragter, DV-Beauftragter und Fachadministrator manuell gestartet werden kann oder über die Schnittstelle „Datenobjekt validieren“ durch eine andere Funktionseinheit gestartet wird.

1. Die eingehende Validierungsanfrage muss das zu validierende Datenobjekt sowie das über den Use-Case „Dateiformate erkennen“ identifizierte Dateiformat für das Datenobjekt enthalten.
2. Da unterschiedliche Validatoren für unterschiedliche Datenformate zum Einsatz kommen können, aber auch mehrere Validatoren für ein Datenformat (zur Erhöhung der Qualität der Validierung) müssen die Validatoren für das Dateiformat ermittelt werden.
3. Jeder Validator, der für das Dateiformat ermittelt wurde, muss aufgerufen werden. Dazu werden dem Validator das Datenobjekt und Format übergeben, gegen dessen Spezifikation geprüft werden soll. Der Validator muss die vom LAKD – Landesarchiv gespeicherte Konfiguration für das Dateiformat laden und anschließend mit der Konfiguration die Validierung durchführen. Ergebnis der Validierung ist ein Validierungsprotokoll. Wenn das Datenobjekt der Spezifikation entspricht, muss der Validator PREMIS-Metadaten nach PREMIS Data Dictionary Version 2.0 (significantProperties, fixity, size) für das Datenobjekt ermitteln und das Validierungsergebnis, das Validierungsprotokoll sowie die PREMIS-Metadaten zurückgeben. Wenn das Datenobjekt nicht der Format-Spezifikation entspricht, muss als Ausgabe das negative Validierungsergebnis und das Protokoll der Validierung zurückgegeben werden. Das Protokoll der negativen Validierung kann im Nachgang der Validierung zur Fehleranalyse verwendet werden.
4. Die Prozesssteuerung muss die Ergebnisse der aufgerufenen Validatoren in einem Protokoll zusammenfassen. Wenn alle Validatoren ein positives Ergebnis zurückgegeben haben, muss die Prozesssteuerung die PREMIS-Metadaten zusammenführen und das positive Validierungsergebnis, das konsolidierte Validierungsprotokoll sowie die zusammengeführten PREMIS-Metadaten als Rückgabe des Validierungsprozesses ausgeben. Die gesamte Validierung ist fehlgeschlagen, wenn einer der Validatoren ein negatives Prüfergebnis zurückgibt. In diesem Fall wird ausschließlich das konsolidierte Validierungsprotokoll zurückgegeben. Die PREMIS-Metadaten werden nicht geliefert.

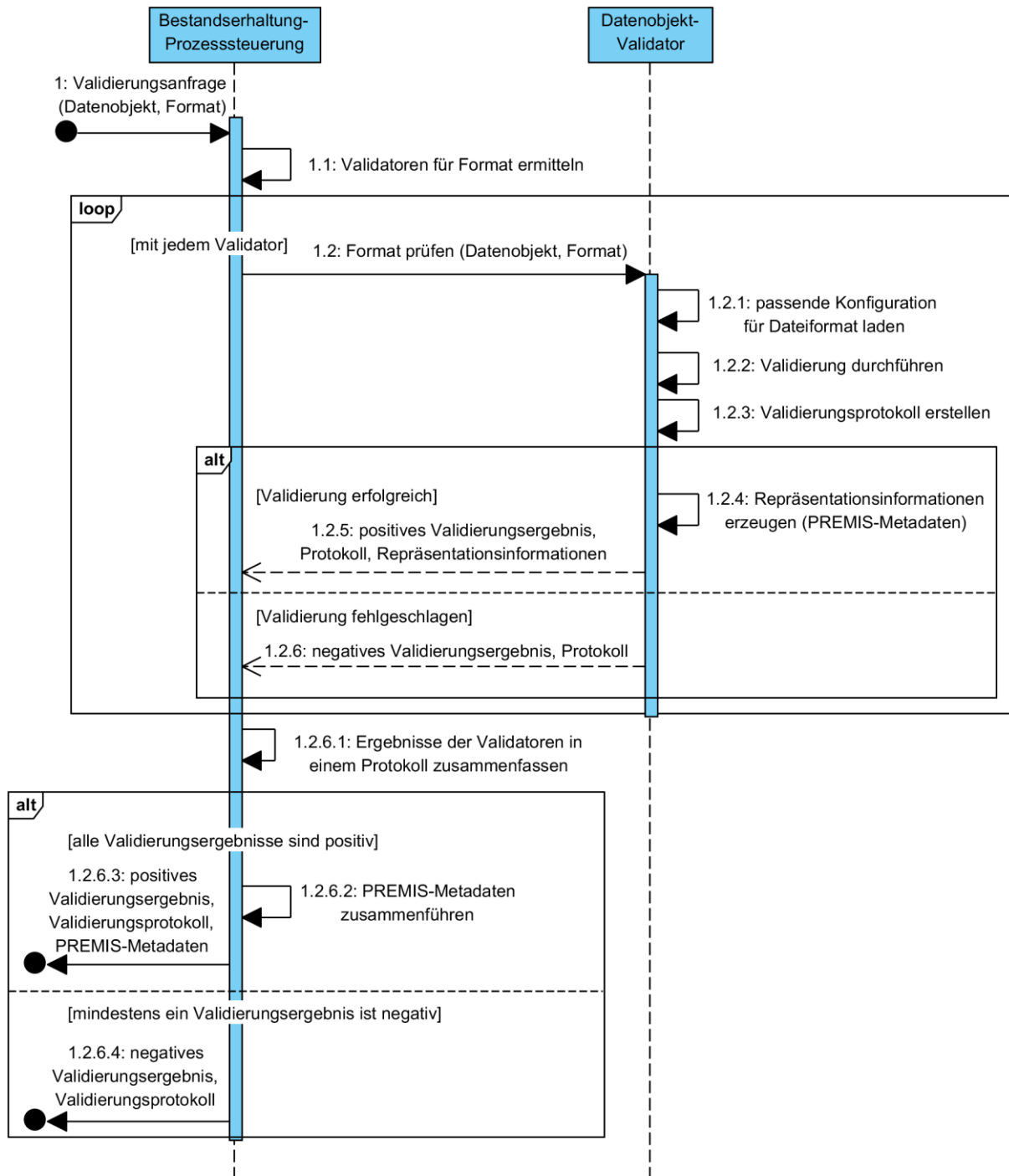


Abbildung 9 Sequenzdiagramm für den Use-Case „Datenobjekte validieren“

4.4.3 Datenobjekte transformieren

Auch der Ablauf zur Transformation von Datenobjekten kann durch die Akteure Archivar, VS-Beauftragter, DV-Beauftragter und Fachadministrator manuell gestartet werden. Alternativ

wird der Prozess im Hintergrund über die Schnittstelle „Datenobjekt transformieren“ durch eine andere Funktionseinheit des eLA gestartet. Es ist folgender Ablauf des Transformationsprozesses vorgesehen:

1. Die eingehende Transformationsanforderung muss das zu transformierende Datenobjekt, das über den Use-Case „Datenformat erkennen“ ermittelte Datenformat für das Datenobjekt sowie das Zieldatenformat enthalten. Das Zieldatenformat für das Quelldatenformat kann im Vorfeld des Aufrufs des Transformationsprozesses aus der Funktionseinheit Datenverwaltung ermittelt werden.
2. Das Datenobjekt, das transformiert werden soll, wird über die Aktivität „Datenobjekte validieren“ geprüft.
3. Anschließend wird der passende Transformator anhand des Quell- und Zieldatenformates ermittelt. Der Transformator wird aufgerufen. Ihm sind das Datenobjekt, das Quell- und das Zieldatenformat zu übergeben. Der Transformator muss die vom LAKD – Landesarchiv getestete Konfiguration für die Transformation laden und die Transformation durchführen.
4. Sofern die Transformation erfolgreich war, erstellt der Transformator ein Protokoll der Formatmigration und gibt das transformierte Datenobjekt sowie das Protokoll als Ergebnis zurück. Es werden die PREMIS-Metadaten (creatingApplication) für den Transformator generiert. Das transformierte Datenobjekt wird mittels der Aktivität „Datenobjekte validieren“ geprüft. Wenn die Validierung positiv ausfällt, wird als Ergebnis des Transformationsprozesses das transformierte Datenobjekt, das Transformationsprotokoll, das Validierungsergebnis, das Validierungsprotokoll sowie die PREMIS-Metadaten zurückgegeben. Schlägt die Validierung fehl, wird ausschließlich das transformierte Datenobjekt, das Transformationsprotokoll, das negative Validierungsergebnis und das Validierungsprotokoll geliefert.
5. Wenn die Transformation nicht erfolgreich war, muss eine durch die aufrufende Funktionseinheit automatisiert-auswertbare Fehlermeldung erzeugt werden.

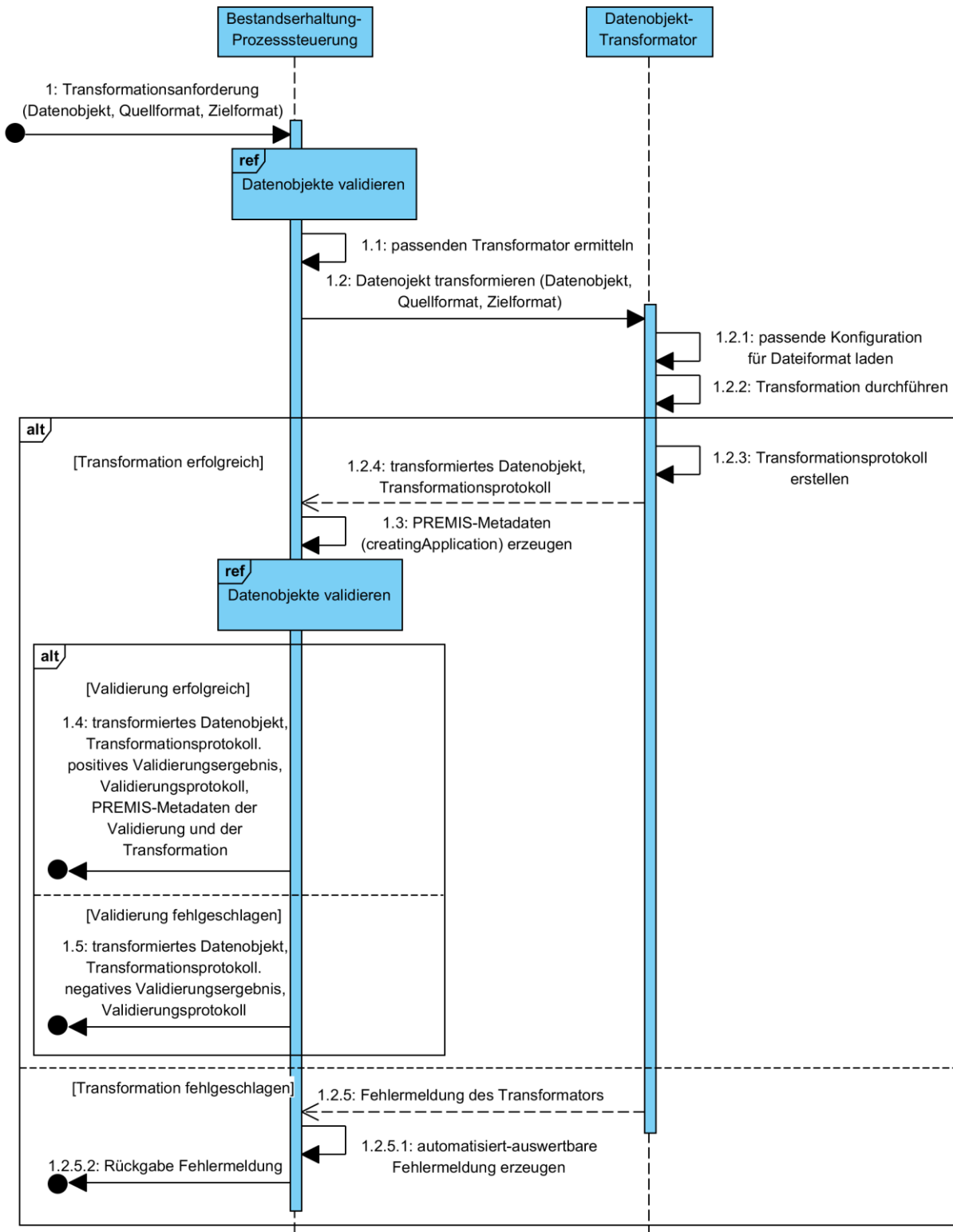


Abbildung 10 Sequenzdiagramm für den Use-Case „Datenobjekte transformieren“

Abbildungsverzeichnis

Abbildung 1 Abgrenzung des Systemkontextes für das vorliegende Konzept	8
Abbildung 2 Logische Sicht auf die Funktionseinheit Bestandserhaltung.....	14
Abbildung 3 Logische Sicht auf den Use-Case "Dateiformat erkennen"	14
Abbildung 4 Logische Sicht auf den Use-Case "Datenobjekte validieren"	15
Abbildung 5 Logische Sicht auf den Use-Case "Datenobjekte transformieren"	16
Abbildung 6 Blackbox-Darstellung für die Funktionseinheit Bestandserhaltung.....	17
Abbildung 7 Whitebox-Darstellung der Funktionseinheit Bestandserhaltung	17
Abbildung 8 Sequenzdiagramm für den Use-Case „Dateiformat erkennen“	21
Abbildung 9 Sequenzdiagramm für den Use-Case „Datenobjekte validieren“	23
Abbildung 10 Sequenzdiagramm für den Use-Case „Datenobjekte transformieren“	25