Koordinierungsstelle
für IT-Standards

IT-Planungsrat

Datenübertragung mit XTA (Version 2.0)

Stand vom 23. August 2013 /
enthält die XTA-WS-Spezifikation

Inhaltsverzeichnis

Überblick über das Dokument

Das vorliegende Dokument enthält die Ergebnisse des Projektes XTA, das vom Mai 2012 bis Juni 2013 im Auftrag des IT-Planungsrates durchgeführt wurde. Dem Hauptdokument sind diverse Anlagen beigefügt.

Im ersten Kapitel (*"Einleitung: Projektauftratg und Ergebnisse"*) werden Hintergrund und Auslöser für das Projekt XTA und die Projektergebnisse (auf unterschiedlichen Detailierungsebenen) beschrieben. Diese Einleitung und Zusammenfassung der Ergebnisse sollen einen Gesamtüberblick herstellen.

Im zweiten Kapitel (*"Kooperation beim Datenaustausch: Anwendungsfälle"*) wird das Gesamtbild des Transports dargestellt: Die Aufgaben und Verantwortlichkeiten der beteiligten Akteure werden beschrieben und durch Anwendungsfälle (Use Cases) visualisiert. In diesen Anwendungsfällen sind organisatorische und technische Aspekte berücksichtigt. Diese Beschreibungen bilden die inhaltlichen Grundlagen für die weiteren Kapitel. Eine alternative Darstellung des "Modells der Rollen und Verantwortlichkeiten" in Form von Sätzen ist als Anhang A beigefügt.

In dem dritten Kapitel (*"XTA-Profilkonzept"*) werden die unterschiedlichen Profilarten beschrieben, mit deren Hilfe die Anforderungen an Transport- und Fachverfahren für die unterscheidlichen Kommunikationsszenarien sinnvoll gebündelt werden können. Prototypische Ausarbeitungen für die einzelnen Profilarten sind als Anhang B beigefügt.

Im vierten Kapitel (*"Spezifikation des XTA-Webservice"*) ist die Spezifikation der XTA-WS 2 und ihre Dokumentation enthalten. Um die inhaltliche Verbindung zu Kapitel 2 herzustellen und die Herleitung der Methoden und Funktionen deutlich zu machen, wurden Beispielszenarien beschrieben. Technik-affine Personen, die insbesondere mit der Umsetzung des XTA-WS betraut sind, werden zusätzlich auf die Schema– und WSDL-Dateien der XTA-WS verwiesen.

Mitwirkung

Die im vorliegenden Dokument enthaltenen Ergebnisse wurden von den Expertengruppen "Organisation und Recht", "Technik" und der "Qualitätssicherung" erarbeitet. Die Projektleitung erfolgte gemäß Beschluss des IT-Planungsrates durch die KoSIT. Folgende Institutionen, vertreten durch die genannten Personen, haben mitgewirkt:

Name	Institution	eMail
Albus, Hagen	procilon	hagen.albus@procilon.de
Apitzsch, Jörg	bremen online services	ja@bos-bremen.de
Buchmann-Cattau, Astrid	Landesbetreib für Statistik und Kommunikationstechnologie Niedersachsen (LSKN)	astrid.buchmann-cattau@lskn.niedersachsen.de
Behrens, Marc	Kommunale Datenverarbeitung Oldenburg	marc.behrens@kdo.de
Biere, Thomas	BSI	thomas.biere@bsi.bund.de
Collatz, Jürgen	ARD ZDF Deutschlandradio Beitragsservice (vormals GEZ)	juergen.collatz@gez.de
Fröhlich, Peter	Landesamt für Bürger- und Ordnungsangelegenheiten (LABO), Berlin	peter.froehlich@labo.berlin.de
Fuhrmann, Martin	Ministerium des Inneres, für Sport und Infrastruktur, Rheinland-Pfalz	martin.fuhrmann@isim.rlp.de
Ganzer, Marco	bremen online services	mg@bos-bremen.de
Genski, Carina	Deutsche Rentenversicherung Bund	carina.genski@drv-bund.de
Hack, Peter	AKDB	hack.peter@akdb.de
Helmer, Frank	citeq	helmer@citeq.de
Jancar, Stephan	Verlag für Standesamtswesen	stephan.jancar@vfst.de
Kampmann, Michael	Citkomm	kampmann.m@citkomm.de
Kannewischer, Sven	Deutsche Rentenversicherung Bund	sven.kannewischer@drv-bund.de
Landvogt,Walter	Bundesdruckerei	walter.landvogt@bdr.de
Laude, Uwe	BSI	uwe.laude@bsi.bund.de
Lindemann, Ralf	bremen online services	rl@bos-bremen.de
Meckelein, Werner	Deutsche Rentenversicherung Bund	werner.meckelein@drv-bund.de
Martin, Matthias	ekom21	matthias.martin@ekom21.de
Mütze, Mario	HSH	mario.muetze@hsh-berlin.com
Neumann, Andreas	Kommunale Informationsverarbeitung Reutlingen-Ulm	andreas.neumann@rz-kiru.de

Name	Institution	eMail
Nitzsche, Lars	procilon	lars.nitzsche@procilon.de
Popp, Ronald	Staatsministerium der Justiz und für Europa, Sachsen	ronald.popp@smj.justiz.sachsen.de
Poppinga, Jann	bremen online services	jpo@bos-bremen.de
Prauser, Ulrike	BMI	ulrike.prauser@bmi.bund.de
Rabenstein,Yorck	init	yorck.rabenstein@init.de
Rammenzweig, Martin	ekom21	martin.rammenzweig@ekom21.de
Rauser, Rainer	Zweckverband Kommunale Datenverarbeitung Region Stuttgart (KDRS)	r.rauser@kdrs.de
Röhl, Mathias	Datenverarbeitungszentrum Mecklenburg-Vorpommern	m.roehl@dvz-mv.de
Riedel, Martin	Datenzentrale Baden-Württemberg	m.riedel@dzbw.de
Rost, Martin	Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein (vertritt auch: AK Technik der Konferenz der Datenschutzbeauftragten des Bundes und der Länder)	uld32@datenschutzzentrum.de
Schlüter, Dieter	dataport	dieter.schlueter@dataport.de
Schulte, Beate	KoSIT	beate.schulte@finanzen.bremen.de
Schwarz, Stefan	Thüringer Landesrechenzentrum	stefan.schwarz@tlrz.thueringen.de
Söhnel , Andreas	Staatsbetrieb Sächsische Informatik Dienste	andreas.soehnel@sid.sachsen.de
Sokoll, Thorsten	IT-Innovationszentrum des Saarlandes	t.sokoll@it-i.saarland.de
Sorgenfrei, Sören	dataport	soeren.sorgenfrei@dataport.de
Steimke, Frank	KoSIT	frank.steimke@finanzen.bremen.de
Steinbeck, Andrea	HSH	andrea.steinbeck@hsh-berlin.com
Steinbeck, Volker	ekom21	volker.steinbeck@ekom21.de
Thede, Heiko	Innenministerium Mecklenburg-Vorpommern	heiko.thede@im.mv-regierung.de
Weck, Andreas	Landesbetrieb Daten und Information Rheinland Pfalz	andreas.weck@ldi.rlp.de

1 Einleitung: Projektauftrag und Ergebnisse

Der IT-Planungsrat hat in seiner 7. Sitzung im März 2012 die Durchführung des Projektes XTA durch folgende Beschlüsse (2012/15) beauftragt:

- Der IT-Planungsrat stimmt dem von der KoSIT vorgeschlagenen Projekt für die Entwicklung des IT-Interoperabilitätsstandards "XTA" für Transportverfahren auf der Basis der Anlagen 1 und 2 zu.
- Der IT-Planungsrat beauftragt die KoSIT mit der Projektleitung.
- Der IT-Planungsrat stellt die erforderlichen Mittel in Höhe von 90.000 EUR aus dem Budget für das NEGS-Projekt "Standardisierungsagenda" zur Verfügung.

(Anlagen 1 und 2 verweisen dabei auf den, in den einschlägigen Gremien abgestimmten Projektauftrag.)

Eine wesentliche Motivation für die Initiierung des Projektes war der Beschluss des AK I der Innenministerkonferenz vom 24./25.10.2011:

- Der AK I begrüßt die Bestrebungen zur Entwicklung eines fachunabhängigen IT-Interoperabilitätsstandards XTA/WS mit dem Ziel einer Vereinheitlichung des Zugangs von Fachverfahren zu der vom KoopA-ADV etablierten Transportinfrastruktur.
- Er hält eine zeitnahe Standardisierung in den Strukturen des IT-Planungsrates für erforderlich und befürwortet die Anwendung einer funktionsfähigen allgemeinen Schnittstelle.

Zusätzlich wurde die Relevanz der Entwicklung eines solchen Standards betont, als der IT-Planungsrat in seiner 8. Sitzung im Juni 2012 den Bedarf an einen einheitlichen Zugang zu Transportverfahren auf die Standardisierungsagenda (2012 – 2015) setzte (Beschluss 2012/23).

Ziele des Projektes XTA:

1. Es sollen Mindeststandards für fachunabhängige Transportverfahren definiert werden.
2. Es sollen Schnittstellen spezifiziert werden, durch deren Nutzung die sichere Übertragung der Daten zwischen Transport- und Fachverfahren (auch innerhalb eines Landes und Rechenzentrums) für die öffentliche Verwaltung kontrollierbar gemacht werden kann. Als eine Teilaufgabe soll die Spezifikation dieser Schnittstelle zwischen Fach- und Transportverfahren als OSCI 2-Profil umgesetzt werden. (Hiervon unberührt ist der Einsatz von OSCI-Transport.)
3. Es soll geklärt werden, wie die Konformität von Transport- und Fachverfahren unter Berücksichtigung der jeweiligen Anwendungsszenarien überprüft werden kann.

Dem IT-Planungsrat sollten Beschlussvorschläge zu folgenden Themen unterbreitet werden:

- zum verbindlichen Einsatz von Transportverfahren, die konform zu den definierten Anforderungen sind („XTA-konform“),
- zum verbindlichen Einsatz XTA-konformer Schnittstellen zwischen Fach- und Transportverfahren,

- zum Verfahren der Konformitätsüberprüfung von Transportverfahren und der Schnittstellen zwischen Transportverfahren und Fachverfahren,
- für ggf. notwendige Folgeaktivitäten.

Von Mai 2012 bis Mai 2013 wurden folgende Ergebnisse durch drei Expertengruppen, in denen ca. 30 Institutionen vertreten waren, erarbeitet:

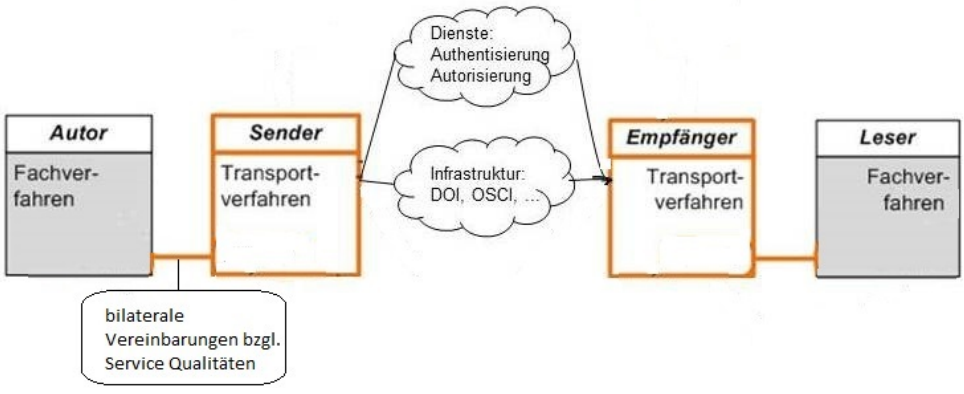
- zu 1 und 2: XTA-Profilkonzept: Da die Anforderungen an die am Transport beteiligten IT-Komponenten stark von den jeweiligen Kommunikationsszenarien abhängen, wurde zur Entwicklung des Standards *nicht* eine fest definierte Menge von Anforderungen erstellt, sondern mit dem XTA-Profilkonzept ein Instrument entwickelt, durch das flexibel auf die unterschiedlichen Kommunikationsszenarien reagiert werden kann. Durch dieses Instrument werden auf unterschiedlichen Abstraktionsebenen Service Qualitäten gebündelt, die von Transport- und Fachverfahren mit ihren Schnittstellen in Abhängigkeit vom Einsatzbereich erfüllt werden müssen.
- ergänzend zu 2: Die Spezifikation der Webservice-Schnittstelle zwischen Fach- und Transportverfahren wurde als OSCI 2-Profilierung erstellt („XTA-WS 2“).
- zu 3: Es wurde ein Entwurf für eine mehrstufige XTA-Konformität für Transportverfahren und Fachverfahren mit ihren Schnittstellen definiert, die in der Praxis erprobt werden soll. Im Rahmen dieser Erprobung soll ein angemessenes Überprüfungsverfahren und ein Betriebskonzept für die einzelnen Komponenten erstellt werden. Die Erprobungsphase wird durch den Aktionsplan des IT-Planungsrates finanziert, wie dieser im Juni 2013 in seiner 11.Sitzung durch Beschluss 2013/21 festgelegt hat.

1.1 Hintergrund

Der Bedarf für die Entwicklung eines Standards für Transportverfahren wurde vor folgendem Hintergrund identifiziert:

In der öffentlichen Verwaltung wird eine große Anzahl unterschiedlicher IT-Fachverfahren eingesetzt. In vielen Arbeitsprozessen ist zwischen den Behörden eine Übermittlung von Fachdaten notwendig. Dies erfolgt i.d.R. auf elektronischem Weg.

Für diese Übermittlung der Daten wird aus dem IT-Fachverfahren heraus meist der Adressat nicht direkt kontaktiert: Die zu transportierenden Daten werden an eine Vermittlungsstelle (Clearingstelle / Nachrichtenbroker) gegeben. Diese Stelle bereitet die Daten für den Transport auf und übermittelt sie über die jeweilige Transportinfrastruktur an die entsprechende Stelle des Adressaten. Dieser stellt sie schließlich dem Adressaten zur Verfügung, so dass im Anschluss die eigentliche Verarbeitung der Fachdaten und ggf. eine unmittelbare Antwort erfolgen kann.



Transportverfahren organisieren den Versand und Empfang von Nachrichten. Dies geschieht entsprechend der jeweils im fachlichen Kontext gültigen Vorgaben und Rahmenbedingungen. Die Umsetzung

des für den Transport geforderten Sicherheitsniveaus, das z.B. für unterschiedliche XÖV-Vorhaben unterschiedlich sein kann, ist Aufgabe des Transportverfahrens.

In der Praxis sind unterschiedliche Ausprägungen von Transportverfahren zu beobachten. Die in vielen Bundesländern eingerichteten Clearing- oder Vermittlungsstellen betreiben für viele angeschlossene Kommunen Transportverfahren für unterschiedliche XÖV-Standards. Diese Transportverfahren stellen eine mögliche Ausprägung dar. Daneben gibt es auf kommunaler Ebene Softwareprodukte, die jeweils einem Fachverfahren vorgeschaltet oder in dieses integriert sind.

Der Leistungsumfang der Transportverfahren geht in den meisten Fällen über den Einsatz der OSCI-Transportbibliothek als auch über die OSCI-Intermediäre hinaus, da sie häufig zusätzlich Eskalationsmechanismen und fachunabhängige Services bieten. Mit Transportverfahren können IT-Fachverfahren von den fachunabhängigen und fachübergreifenden Aspekten des Versands und Empfangs sowie von spezifischen technischen und organisatorischen Gegebenheiten von Betreibern und einzelnen Bundesländern entkoppelt werden.

Während Informationsverbünde für XÖV-Vorhaben (z.B. Meldewesen, Personenstandswesen oder elektronischer Rechtsverkehr) im länderübergreifenden Verkehr abgestimmte Vorgaben zur Funktionalität und zum Teil auch zur Service Qualität definiert haben, werden Datenübermittlungen zwischen Fachverfahren und Transportverfahren insbesondere für den landesinternen Einsatz sehr unterschiedlich und individuell umgesetzt.

Hierfür gibt es folgende Ursachen:

- Für eine Bundesland-interne Datenübermittlung gibt es bisher keine übergreifenden, einheitlichen Vorgaben. Die Transportverfahren sind mit ihren Schnittstellen für die einzelnen Einsatzbereiche von unterschiedlichen Softwareherstellern, meist historisch gewachsen, entwickelt worden.
- Transportverfahren beziehen sich bisher auf die einzelnen XÖV-Standards und bilden damit jeweils die spezifischen Anforderungen ab. Unterschiede in den Standards, insbesondere in den Datenelementen, sind zum Teil den unterschiedlichen rechtlichen bzw. organisatorischen Vorgaben geschuldet.

Dies hat zum einen zur Folge, dass die Kosten zur Unterhaltung insbesondere der Schnittstellen zwischen Transport- und Fachverfahren unverhältnismäßig hoch sind: Dies betrifft sowohl Rechenzentren als zentrale Betreiber von Transportverfahren, die eine große Anzahl von Fachverfahren betreuen, als auch Hersteller, deren Fachverfahren überregional eingesetzt werden und die damit verschiedene Schnittstellen zu unterschiedlichen XÖV-spezifischen Transportverfahren implementieren müssen.

Zum anderen haben die sehr unterschiedlichen Ausprägungen der Transportverfahren also insbesondere zur Folge, dass die öffentliche Verwaltung keine verlässlichen Aussagen über die Transportinfrastruktur in ihrer Gesamtheit, und damit auch zu ihrer Leistungsfähigkeit, der Datensicherheit und des Datenschutzes machen kann. Es können keine Aussagen zur Service Qualität für die *gesamte* Strecke zwischen den Fachverfahren gemacht werden. Dies gilt auch in Fällen, in denen es abgestimmte Vorgaben für den Bereich der länderübergreifenden Datenübermittlung gibt.

1.2 Ziel

Ziel ist, einheitliche Vorgaben für fach- und XÖV-unabhängige Funktionen und Qualität von Transportverfahren mit ihren Schnittstellen abzustimmen, so dass diese von dem IT-Planungsrat verbindlich vorgegeben werden können. Im Fokus der Arbeit steht hierbei die Übertragung von XÖV-Nachrichten.

Durch eine verbindliche Nutzung des XTA-Standards soll erreicht werden, dass die Anforderungen zwischen zwei Fachverfahren beim Datenaustausch zwischen Bund und Ländern und beim Datenaustausch zwischen Ländern oder landesintern vereinheitlicht werden.

Durch das Projekt XTA sollen die Voraussetzungen dafür geschaffen werden, dass auf der gesamten Strecke des Datenaustausches zwischen Fachverfahren, die länderübergreifend, landesintern oder zwi-

schen Bund und Ländern miteinander kommunizieren, Anforderungen bezüglich der Leistungsfähigkeit, der Datensicherheit und des Datenschutzes durch die Verwaltung definiert, verbindlich vorgegeben und überprüft werden können. Hierbei ist es *nicht* das Ziel, Vorgaben für die Verfügbarkeit der Transportinfrastruktur innerhalb der Länder festzuschreiben oder übergreifende Service Level Agreements zu definieren.

Durch die Vereinheitlichung der Schnittstellen sollen deren Entwicklungs- und Pflegeaufwände reduziert werden.

Diese Ziele werden in zwei Schritten erreicht:

Im ersten Schritt werden die Voraussetzungen zur Vereinheitlichung geschaffen, indem auf der Grundlage des XTA-Profilkonzeptes ein Konzept für eine XTA-Konformität erstellt und die Spezifikation für die einheitliche Webservice-Schnittstelle (XTA-WS 2) definiert werden.

Im zweiten Schritt wird ein Überprüfungs- / Zertifizierungsverfahren als Grundlage für einen verbindlichen Einsatz der XTA-Konformität erarbeitet.

Die Zieldefinition erfolgte in Abstimmung mit den einschlägigen Gremien.

1.3 Projektergebnisse

1.3.1 Zusammenfassung

Die definierten Ziele werden in zwei Schritten erreicht:

Im ersten Schritt wurden entsprechend den Vorgaben des Projektauftrages Ergebnisse erarbeitet, die im vorliegenden Dokument „Datenaustausch mit XTA“ und seinen Anhängen enthalten sind:

1. **XTA-Profilkonzept als Basis des XTA-Standards:** Im Zuge der Entwicklung der Anforderungen insbesondere an Transportverfahren wurde deutlich, dass die Vielfalt der Kommunikations- und Einsatzszenarien berücksichtigt werden muss: Das XTA-Profilkonzept wird als flexible Lösung angeboten, durch das ermöglicht wird, die Anforderungen an Transportverfahren und Fachverfahren bezüglich IT-Sicherheit (auf der Grundlage des BSI-Grundschutzes) und Datenschutz zu standardisieren; gleichzeitig wird hierbei eine ausreichende Flexibilität für unterschiedliche Kommunikationsszenarien mit den jeweils betroffenen Komponenten ermöglicht. Das XTA-Profilkonzept stellt das Grundgerüst des XTA-Standards dar.
2. **Definition einer dreistufigen XTA-Konformität:** Es wird eine mehrstufige XTA-Konformität für die unterschiedlichen am Transport beteiligten Komponenten vorgeschlagen.
3. **Vereinheitlichung der Webserviceschnittstellen zwischen Fachverfahren und Transportverfahren (XTA-WS 2):** Die Webserviceschnittstelle zwischen Fachverfahren und Transportverfahren wurde als OSCI 2- Profilierung umgesetzt. („XTA-WS 2“). Die Definition der XTA-WS 2 setzte hierbei auf die Arbeiten der Arbeitsgruppe "AG XTA", die ein wesentlicher Impulsgeber für das Projekt XTA gewesen war, auf. In die qualitätssichernde Überarbeitung und Erstellung einer XÖV-konformen Dokumentation sind neue Funktionen wie die Option einer synchronen Kommunikation eingeflossen.

Im zweiten Schritt soll eine Erprobungsphase durchgeführt werden, in der insbesondere die Prozesse des XTA-Profilkonzeptes, die die Grundlage für eine zukünftige Überprüfung einer XTA-Konformität darstellen, zu verstetigen. Daneben sollen die praktischen Erfahrungen in Betriebskonzepte einfließen.

1.3.2 Ergebnisse im Einzelnen

Die Umsetzung der durch den Projektauftrag vorgegebenen Aufgaben führte zu folgenden Ergebnissen:

Modell der Rollen und Verantwortlichkeiten (Beschreibung des Gesamtszenarios):

Als Fundament und gemeinsame Ausgangssituation für die weitere Projektarbeit wurden in der ersten Projektphase die Aufgaben und Verantwortlichkeiten aller Akteure, die vom Transport von Fachdaten berührt sind, beschrieben.

Dies betrifft:

1. die Aufgaben und Verantwortlichkeiten der Behörden, die in den IT-Fachverfahren die Fachdaten erstellen und sie für den Transport zur Verfügung stellen. („Autoren");
2. die Aufgaben und Verantwortlichkeiten der Vermittlungsstellen (auch Clearingstellen oder Nachrichtenbroker genannt), die die Daten von den Behörden entgegen nehmen und sie entsprechend der rechtlichen und fachlichen Vorgaben aufbereiten und versenden („Sender");
3. die Aufgaben und Verantwortlichkeiten der Vermittlungsstellen auf der Gegenseite, die die Nachrichten vom Sender entgegen nehmen („Empfänger");
4. und schließlich die Aufgaben und Verantwortlichkeiten der Behörden, an die die Fachdaten adressiert wurden („Leser") und die diese verarbeiten.

Die Betrachtung dieses Gesamtszenarios führte zu einer Zusammenstellung von ca. 40 Sätzen, die in Kapitel 2 durch die Verwendung von UML-Anwendungsdiagrammen visualisiert und in linearer Form in Anhang A beigefügt sind.

Erarbeitung des XTA-Profilkonzeptes:

Die Definition des Gesamtszenarios führte zur Einsicht, dass die Anforderungen mit ihren Ausprägungen an Transportverfahren einerseits sehr stark von den jeweiligen Einsatzszenarien abhängen, sie aber andererseits in vergleichbaren Einsatzszenarien sehr ähnlich sind. Daher wurde ein „XTA-Profilkonzept" entwickelt, dessen Anwendung zum einen eine Standardisierung der Transportverfahren ermöglichen soll, und zum anderen durch ein Baukastenprinzip die benötigte Flexibilität beinhaltet:

- Die Anforderungen selbst werden in „**Schutzprofilen**" standardisiert formuliert. Die Anforderungen betreffen insbesondere Vorgaben zum Niveau der IT-Sicherheit und zum Datenschutz, können aber auch weitere rechtliche oder fachliche Vorgaben enthalten. Sie werden auf hoher abstrakter Ebene formuliert. Hierfür werden typische Konstellationen gebündelt in einzelnen Schutzprofilen zusammengestellt. Für die einzelnen Kommunikationsszenarien soll einmalig entschieden werden, welches Schutzprofil notwendig ist und eingesetzt werden soll. Im Projekt XTA wurden 4 prototypische Schutzprofile definiert, die dem Anhang B beigefügt sind.
- Für die *Umsetzung* der Schutzziele werden Vorgaben bzgl. der zu verwendenden IT-Infrastrukturkomponenten gemacht. Hierdurch soll zum einen eine Nachhaltigkeit sichergestellt werden, die die Investitionen der öffentlichen Verwaltung schützt. Zum anderen sollen durch die Verwendung der insbesondere vom IT-Planungsrat zur Verfügung gestellten standardisierten IT-Komponenten die Qualitätsanforderungen einfach überprüfbar sein. Innerhalb des Projektes XTA wurden beispielhaft zwei **Infrastrukturprofile** erarbeitet, die dem Anhang B beigefügt sind und die als Vorlage, ggf. entsprechend angepasst, für eine große Anzahl an Kommunikationsszenarien der Innenverwaltung und der Justiz bedienen können. Diese Infrastrukturprofile sind abgeleitet von bereits heute typischerweise eingesetzten IT-Komponenten.
- Die konkrete Konfiguration der durch die Infrastrukturprofile vorgegebenen IT-Komponenten in den jeweils durch die Schutzprofile benötigten Ausprägungen wird in **Wesensprofilen** vorgenommen. Prototypische und beispielhafte Ausarbeitungen der Profile sind dem Anhang B beigefügt.

Das XTA-Profilkonzept unterscheidet damit drei Arten von Profilen, für die jeweils ein Set der typischen Ausprägungen von den jeweils fachlich Verantwortlichen erarbeitet werden muss. Inhaltlich ersteht hierdurch keine neue Aufgabe. Durch die Entwicklung und Nutzung der zentral abgelegten Profile reduziert sich zukünftig der Aufwand bei der Definition eines Kommunikationsszenarios und treibt damit die Standardisierung des elektronischen Datenaustausches weiter voran.

Definition einer drei-stufigen XTA-Konformität:

Eine Teilaufgabe des Projektes bestand in der Erarbeitung eines Vorschlags zur Überprüfung einer XTA-Konformität, durch die die Einhaltung der an ein Transportverfahren mit seinen Schnittstellen gestellten Anforderungen für die öffentliche Verwaltung leicht überprüfbar sein soll.

Das XTA-Profilkonzept bildet die Basis für diese Überprüfbarkeit. Für Transportverfahren und Fachverfahren mit den jeweils betroffenen Schnittstellen wird jeweils festgelegt, welche Profile erfüllt sein müssen, um eine XTA-Konformität zu erreichen.

Um die Schwelle zur Etablierung der XTA-Konformität niedrig zu halten, wird eine stufenweise Einführung angeboten: Sie reicht damit von einer technischen Konformität bzgl. der XTA-WS bis zur Berücksichtigung der Betriebsumgebung der beteiligten IT-Komponenten.

Die XTA-Konformität sowohl für Fachverfahren als auch für Transportverfahren wird damit in drei Stufen aufgesplittet:

- XTA-WS-Spezifikationskonformität: Fachverfahren und Transportverfahren sind XTA-WS-spezifikationskonform, wenn sie die jeweils für sie relevanten Teile der XTA-Webservice Schnittstelle umgesetzt haben.
- XTA-Wesenskonformität: Fach- und Transportverfahren sind wesenskonform, wenn sie die jeweils beauftragten Profile unterstützen.
- XTA-Betriebskonformität. Fach und Transportverfahren sind XTA-betriebskonform, wenn sie in einer den Wesensprofilen entsprechenden Betriebsumgebung betrieben werden.

Erstellung einer Spezifikation für eine einheitliche Schnittstelle zwischen Fachverfahren und Transportverfahren:

Es wurde eine Spezifikation für eine Webserviceschnittstelle erarbeitet. Ausgehend von der XTA-WS 1.1.1., die in der "AG XTA" von Fachverfahrensherstellern und Rechenzentren erarbeitet worden war, wurden folgende Ergebnisse erzielt:

- XTA-WS 2 als OSCI 2 Profilierung
- XÖV-konforme Dokumentation
- Abgleich der XTA-WS 2 mit den Aufgaben, die aus der Betrachtung des Gesamtszenarios resultieren, und eine Verallgemeinerung und Ergänzung von Funktionen

Überprüfung der Projektergebnisse in einer Pilotphase:

Durch das Projekt XTA wurden komplexe Konzepte erarbeitet, deren praktische Erprobung noch aussteht. Die Erprobung und Verstetigung insbesondere des XTA-Profilkonzeptes wird begleitet durch die Erstellung eines Betriebskonzeptes, in das die praktischen Erfahrungen einfließen sollen.

Weiterhin soll geprüft werden, in welcher Art eine Konformitätsüberprüfung für die unterschiedlichen Komponenten erfolgen soll.

2 Kooperation beim Datenaustausch: Anwendungsfälle

2.1 Einleitung

In den Prozesse der Datenübermittlung kooperieren die Infrastrukturkomponenten Fachverfahren und Transportverfahren zusammen. Fachverfahren sind dabei die IT-Verfahren, die in Behörden für die Vorgangsbearbeitung der jeweiligen Fachdomäne (z.B. Personenstandswesen, Pass- und Ausweisbehörde) eingesetzt werden. Transportverfahren haben die Funktion, Nachrichten zu senden, zu empfangen und an weiteren Aspekten der Übermittlung mitzuwirken. Dies geschieht unabhängig von der jeweiligen Fachdomäne.

Fach- und Transportverfahren werden häufig von getrennten Organisationen (z.B. Behörde und Rechenzentrum) betrieben, die für die Zwecke der Nachrichtenübermittlung vereinbarte Dienstleistungsbeziehungen eingehen.

In diesem Kapitel werden die Anwendungsfälle, die beim Datenaustausch notwendig sind, beschrieben. Hierbei wird von den IT-Verfahren abstrahiert. Stattdessen werden die Aufgaben und Prozesse auf der Basis der Rollen analysiert, in denen die Akteure der Prozesse kooperieren. Daher werden nicht die Fachverfahren, sondern die Rollen "Autor" und "Leser" benannt und statt der Transportverfahren die Rollen "Sender" und "Empfänger".

Der Darstellung der Anwendungsfälle liegen die Fragen zugrunde, welche Aufgabe und Zuständigkeitsbereiche an Erstellung, Transport und Verarbeitung von Nachrichten geknüpft sind. Hiermit verbundene Kompetenzen, Rechten und Pflichten werden berücksichtigt.

Die Anwendungsfälle werden als Use Case Modell dargestellt. Für die Visualisierung wird die UML-Notation gewählt, die es gestattet, die Beteiligung von Akteuren an Anwendungsfällen, die Beziehung eines Anwendungsfalls zu anderen Anwendungsfällen und auch zu Informationsobjekten (Klassen oder Objekten) darzustellen.

Ziel ist es hierbei, eine Sicht zu entwickeln, die sowohl fachlich-organisatorische als auch technische Aspekte berücksichtigt, sie aber voneinander abgrenzt und in einen Zusammenhang stellt.

So können technische Komponenten wie der XTA-Webservice (XTA-WS) in ihrer Funktionalität klarer bestimmt werden. Der XTA-WS muss aus dieser Sichtweise alles anbieten, was die Interaktion der Rollen "Autor" und "Leser" mit den Rollen "Sender" bzw. "Empfänger" unterstützt. Wenn also der Autor einen Transportauftrag erteilen können soll, muss beispielsweise eine entsprechende Operation bzw. Methode im XTA-WS angeboten werden.

Ergänzend zum hier dargestellten Use Case Modell wird auf das Modell der Rollen und Verantwortlichkeiten verwiesen (siehe Anhang A).

Grundlage dieser Darstellung ist der rechtliche Rahmen, der durch den Gesetz- oder Verordnungsgeber vorgegeben wird und der von den Akteuren zu beachten ist. Die rechtlichen Rahmenbedingungen können Vorgaben für die Qualität der Datenübermittlung enthalten.

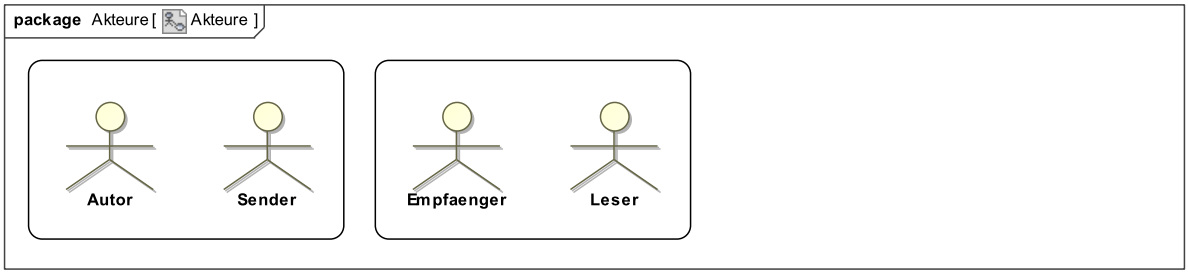
2.2 Anwendungsfälle beim Datenaustausch

Das Kapitel beginnt mit der Vorstellung der an den Anwendungsfällen beteiligten Akteure und mit einer Überblicksdarstellung, in der die Handlungsfelder skizziert sind. Jeder hier genannte Anwendungsfall wird weiter unten detailliert dargestellt.

2.2.1 Akteure

Alle beteiligten Akteure setzen in den Prozessen die jeweils anzuwendenden rechtlichen Regelungen um.

Abbildung 2.1. Anwendungsfalldiagramm "Akteure"



2.2.1.1 Autor

Der Autor ist als Fachverantwortlicher für die fachliche Erstellung der zu transportierenden Nachricht zuständig. Er ist außerdem Auftraggeber des Transports, für den er die rechtlich-organisatorischen Rahmenbedingungen vorgibt, und er überwacht die Erfüllung seiner Transportanforderungen.

Eine detaillierte Darstellung der Aufgaben und Verantwortlichkeiten des Autors wird in Abschnitt A.2.1 ab Seite 89 gegeben.

2.2.1.2 Sender

Der Sender ist gemäß des Transportauftrags des Autors zuständig für die Abwicklung des Transports und aller damit zusammenhängenden Leistungen wie Adressierung, Transport-Verschlüsselung und -Signatur sowie Protokollierung.

Detaillierter werden seine Aufgaben und Verantwortlichkeiten in Abschnitt A.2.2 ab Seite 91 dargestellt.

2.2.1.3 Empfaenger

Der Empfänger ist in seiner Aufgabe als Transporteur vom Leser beauftragt, Nachrichten entgegenzunehmen, sie vorzuhalten und sie dem Leser ggf. direkt zuzustellen. Der Empfänger ist außerdem beauftragt, hiermit verbundene Aufgaben auszuführen. Dies sind insbesondere die Prüfungen zur Identität und von Zertifikaten und Aufgaben der Protokollierung.

Im Detail werden die Aufgaben und Verantwortlichkeiten des Empfängers in Abschnitt A.2.3 ab Seite 92 dargestellt.

2.2.1.4 Leser

Als Fachverantwortlicher ist der Leser zuständig für die Entgegennahme und fachliche Auswertung der transportierten Nachricht. Zu den Auswertungen gehören insbesondere die Prüfungen in Bezug auf die Autorenschaft.

Eine detaillierte Darstellung seiner Aufgaben und Verantwortlichkeiten wird in Abschnitt A.2.4 ab Seite 94 gegeben.

2.2.2 Anwendungsfälle im Überblick

Diese Abbildung gibt einen Überblick über die Handlungsfelder, die vom Standard XTA berührt sind.

Jedem Anwendungsfall aus diesem Diagramm entspricht weiter unten eine detailliertere Darstellung in einem eigenen Diagramm.

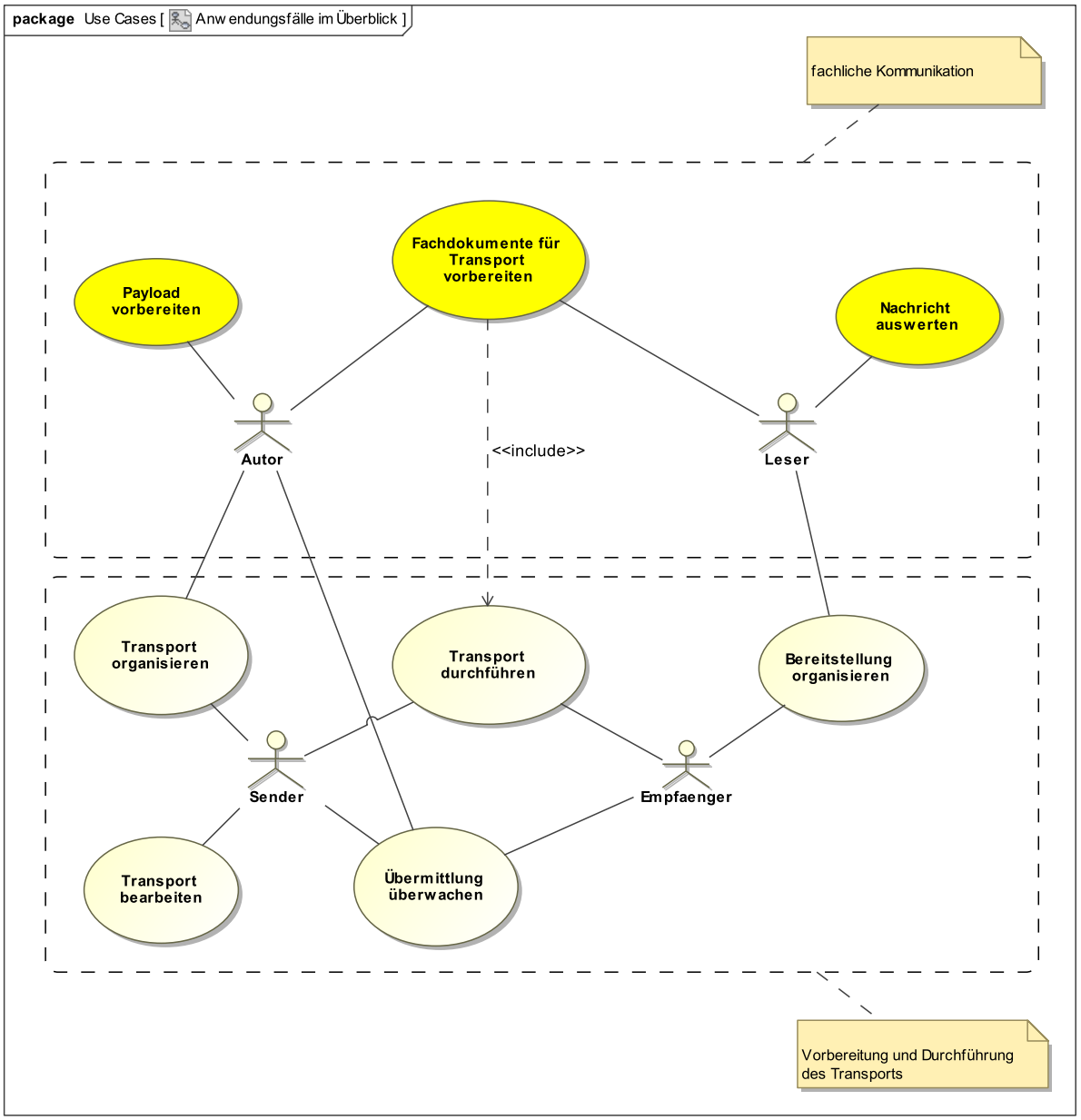
Diese Anwendungsfall bezogene Darstellung ergänzt die lineare Sicht, die der Architektur-Abbildung und dem Rollenmodell zugrunde liegt (siehe Anhang A, *Modell der Rollen und Verantwortlichkeiten*). Die im Rollenmodell benannten Kernprozesse für die Akteure Autor, Sender, Empfänger und Leser werden in den Anwendungsfalldiagrammen um detailliertere organisatorische und technische Aktivitäten ergänzt.

Zur Verwendung der Anwendungsfalldiagramme: Es werden sowohl die aktiven Akteure als auch die indirekt betroffenen oder eher passiven Akteure dargestellt und benannt. Beispiel: Beteiligte Akteure im Anwendungsfall "Transport durchführen" sind der Sender und auch der Empfänger.

In der Dokumentation werden Anwendungsfalldiagramme auch als "Use Cases" (UC) bezeichnet.

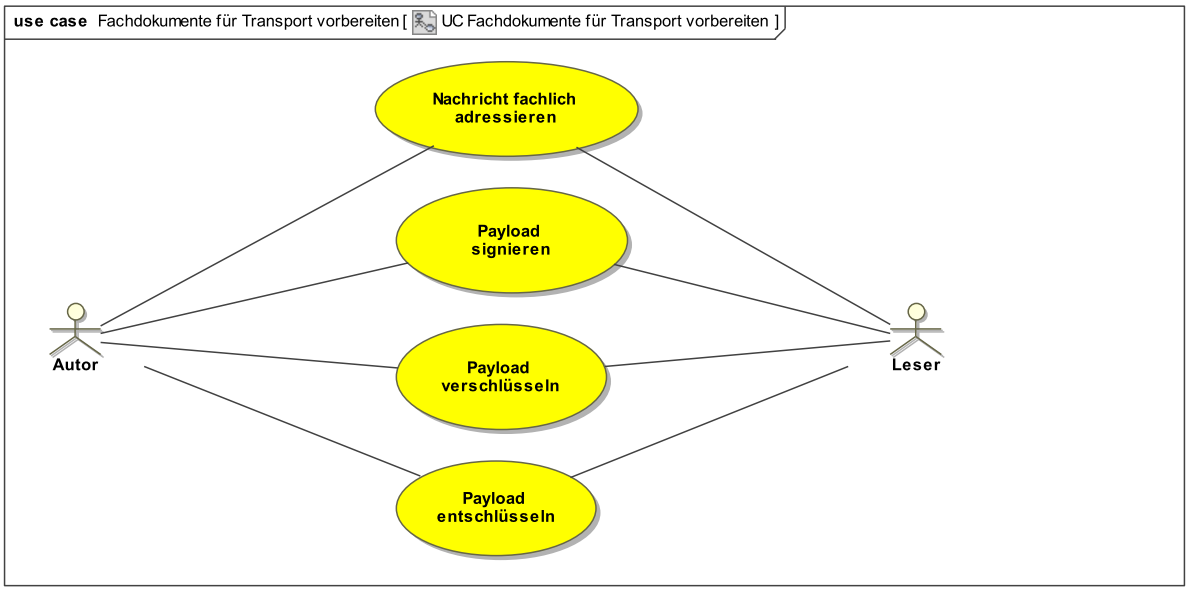
Die in der Dokumentation der Anwendungsfälle verwendeten Fachbegriffe werden im Glossar erläutert.

Abbildung 2.2. Anwendungsfalldiagramm "Anwendungsfälle im Überblick"



2.2.3 UC Fachdokumente für Transport vorbereiten

Abbildung 2.3. Anwendungsfalldiagramm "UC Fachdokumente für Transport vorbereiten"



Der Nachrichtenaustausch ist in fachliche Prozesse eingebettet, an denen Beteiligte organisationsübergreifend mitgewirken können. Diese werden an definierten Stellen in den Prozess einbezogen und ggf. zu Folgeprozessen innerhalb ihrer Zuständigkeit veranlasst.

Der Austausch von Nachrichten wird realisiert auf der Basis technischer Integration von IT-Systemen.

2.2.3.1 Enthaltene Anwendungsfälle

2.2.3.1.1 Nachricht fachlich adressieren

Beschreibung
Der Autor ist für die fachliche Adressierung des Lesers zuständig. Der Autor kann prüfen, ob der Leser in einem bestimmten fachlichen Kontext grundsätzlich elektronisch erreichbar ist. (Hiermit ist nicht die Prüfung gemeint, ob der Leser aktuell erreichbar / verfügbar ist.) Der Sender stellt hierfür eine entsprechende Funktionalität zur Verfügung. Die Prüfung erfolgt durch qualitätsgesicherte Verzeichnisse der öffentlichen Verwaltung (z.B. DVDV, S.A.F.E.). Der Autor muss benötigte Attribute für die elektronische Kommunkation mit dem Leser abrufen können, sofern dies im fachlichen Kontext notwendig ist. Hierbei stellt der Sender eine entsprechende Funktionalität zur Verfügung.

2.2.3.1.2 Payload signieren

Beschreibung
Der Autor kann die zu transportierende Nachricht oder Teile der Nachricht signieren. Der Autor ist zuständig für die Signatur der Nachricht, nicht der Sender. Ein Sender kann im Auftrag des Autors diese Aufgabe wahrnehmen. Der Autor muss immer über die Signatur identifizierbar bleiben.

2.2.3.1.3 Payload verschlüsseln

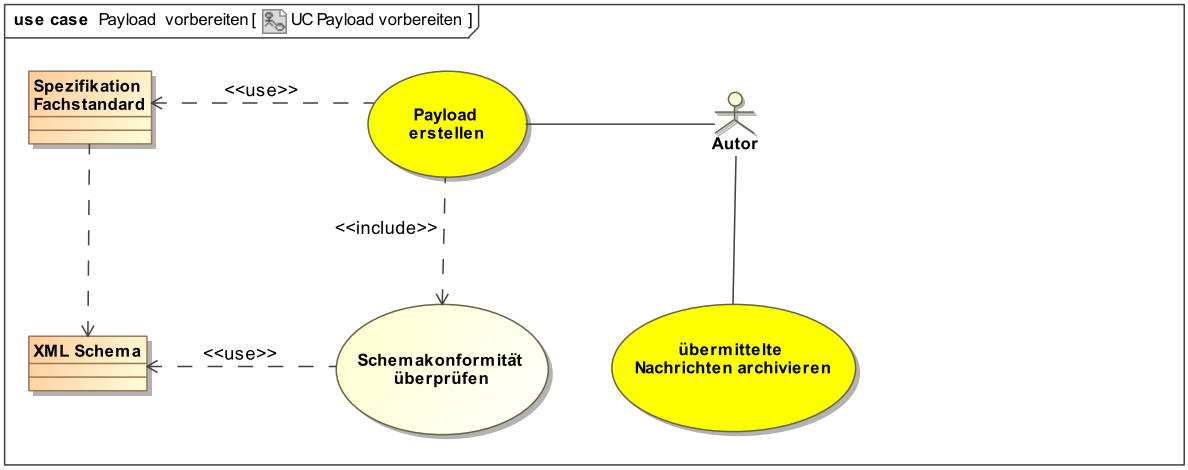
Beschreibung
Der Autor kann die zu transportierende Nachricht oder Teile der Nachricht verschlüsseln. Der Autor ist zuständig für die Verschlüsselung der Nachricht, nicht der Sender. Ein Sender kann im Auftrag des Autors diese Aufgabe wahrnehmen.

2.2.3.1.4 Payload entschlüsseln

Beschreibung
Der Leser kann die zu transportierende Nachricht oder Teile der Nachricht entschlüsseln. Der Leser ist zuständig für die Entschlüsselung der Nachricht, nicht der Empfänger. Ein Empfänger kann im Auftrag des Lesers diese Aufgabe wahrnehmen.

2.2.4 UC Payload vorbereiten

Abbildung 2.4. Anwendungsfalldiagramm "UC Payload vorbereiten"



Der Nachrichtenaustausch ist in organisatorisch-fachliche Prozesse eingebunden, die unterschiedliche Akteure betreffen. Diese Partner werden an definierten Stellen in den Prozess einbezogen und ggf. zu Folgeprozessen innerhalb ihrer Zuständigkeit veranlasst.

Der Use Case fasst Aktivitäten des Autors, der fachliche Prozesse im Rahmen seiner Zuständigkeit verfolgt und an definierten Schnittstellen Nachrichten- und Daten für den Austausch mit Kooperationspartnern (Leser) erstellt, zusammen.

Bei entsprechenden vertraglichen Regelungen zwischen Autor und Sender kann die Durchführung bestimmter Aufgaben an den Sender delegiert werden.

2.2.4.1 Enthaltene Anwendungsfälle

2.2.4.1.1 Payload erstellen

Eingebundene Use Cases		
Use Case	**Ref.**	**Seite**

Eingebundene Use Cases		
Schemakonformität überprüfen	2.2.4.1.2	17
Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
Spezifikation Fachstandard	2.3.5	28

Beschreibung

Aktivitäten des Autors, der im Rahmen seiner fachlichen Zuständigkeit die zu übermittelnde Nachricht vorbereitet:

Der Autor ist fachlich zuständig, d.h. er ist für den Inhalt der zu transportierenden Nachricht verantwortlich.

Der Autor erstellt den Inhalt der zu transportierenden Nachricht. Er erstellt die zu transportierende Nachricht gemäß den Regeln des zu Grunde liegenden Standards (z.B. OSCI-XMeld) in einer bestimmten Version.

Amerkung:

- *Der vollständige Inhalt der vom Autor erstellten Nachricht ist für den Leser relevant. Und alles, was für den Leser relevant ist, sollte in der Nachricht enthalten sein. Dies betrifft auch die Informationen, die im Nachrichtenkopf einer XÖV-Nachricht (etwa vergleichbar dem Inhalt eines Briefkopfes), enthalten sind, wie z.B. der AGS von Absender und Empfänger sowie die Nachrichten-Identifizierung.*
- *Es ist nicht ausgeschlossen, dass einzelne Informationen aus dem Briefkopf auch für den Sender relevant sind. Dies kann der Fall sein, wenn der Sender die Informationen benötigt, um die technischen Adressdaten des Lesers / Empfängers zu ermitteln*

Der Autor ist verantwortlich dafür, dass die Nachricht spezifikationskonform ist. Das schließt ein, dass die Nachricht valide bezüglich des für den Standard (in der entsprechenden Version) gültigen Schemas ist.

2.2.4.1.2 Schemakonformität überprüfen

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
XML Schema	2.3.9	29

Beschreibung

Der Autor ist verantwortlich für die Schemakonformität der Nachricht. Er stellt sicher, dass die Nachricht valide bezüglich der XML-Schema-Definition, die zur Spezifikation gehört, ist. Der Autor kann, bei entsprechender vertraglicher Regelung, diese Aufgabe an den Sender delegieren.

2.2.4.1.3 übermittelte Nachrichten archivieren

Beschreibung

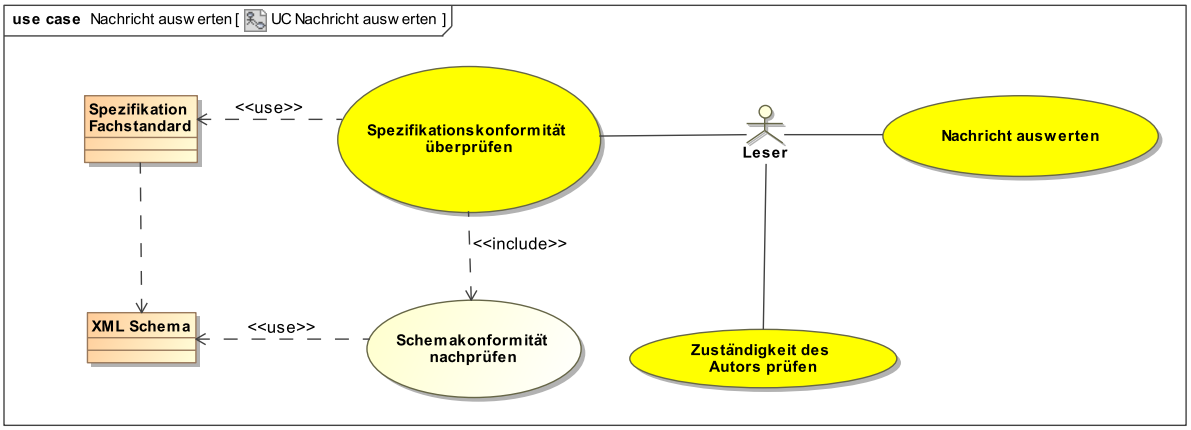
Entsprechend der rechtlichen Vorgaben werden die Nachrichten archiviert:

Der Autor ist für die Aufbewahrung der versandten Nachrichten und der relevanten Transportinformation zuständig und dafür, dass fristgerecht gelöscht wird. Für die Aufbewahrung kann er sich eines Dienstleisters bedienen.

Der Autor legt fest, wie lange beim Sender die Nachrichten und die Protokolle der Nutzungsdaten gespeichert werden. Die Löschfristen werden vertraglich vereinbart.

2.2.5 UC Nachricht auswerten

Abbildung 2.5. Anwendungsfalldiagramm "UC Nachricht auswerten"



Der Leser nimmt die übermittelte Nachricht an den definierten fachlichen Schnittstellen entgegen und führt eine inhaltliche Analyse mit den daraus folgenden Aktivitäten durch.

2.2.5.1 Enthaltene Anwendungsfälle

2.2.5.1.1 Spezifikationskonformität überprüfen

Eingebundene Use Cases		
Use Case	**Ref.**	**Seite**
Schemakonformität nachprüfen	2.2.5.1.2	18
Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
Spezifikation Fachstandard	2.3.5	28
Beschreibung		
Der Leser prüft die Nachricht gegen die Regeln des zugehörigen Fachstandards.		

2.2.5.1.2 Schemakonformität nachprüfen

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
XML Schema	2.3.9	29
Beschreibung		
Der Leser überprüft, ob die Nachricht valide bzgl. der XML-Schema-Definition, die zur Spezifikation gehört, ist. Der Leser kann diese Aufgabe durch entsprechende vertragliche Regelungen an den Empfänger delegieren.		

2.2.5.1.3 Zuständigkeit des Autors prüfen

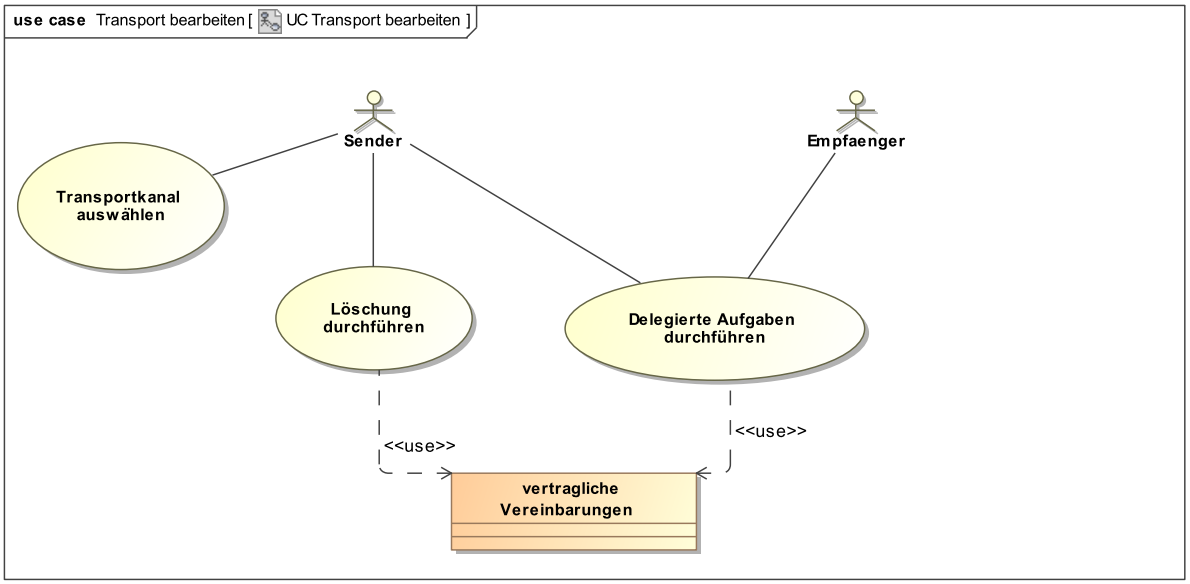
Beschreibung
Der Leser prüft Zuständigkeit und Berechtigung des Autors. Die Prüfung erfolgt, weil der Leser aus dem Ergebnis ableiten kann, wie er die erhaltenen Informationen verarbeitet, ob z.B. ein Register fortgeschrieben werden muss. Diese Prüfung kann an den Empfänger delegiert werden.

2.2.5.1.4 Nachricht auswerten

Beschreibung
Aktivitäten des Lesers, der im Rahmen seiner fachlichen Zuständigkeit die übermittelte Nachricht inhaltlich analysiert und entsprechend weiterbearbeitet. Die Regeln des entsprechenden Fachstandards unterstützen den Leser, den fachlichen Inhalt der Nachricht zu identifizieren und zu interpretieren.

2.2.6 UC Transport bearbeiten

Abbildung 2.6. Anwendungsfalldiagramm "UC Transport bearbeiten"



Vor und nach dem eigentlichen Transport der Daten sind Sender und Empfänger für weitere, hiermit eng verbundene Aufgaben zuständig. Dazu zählen Aufgaben, die als Delegation vom Autor bzw. Leser durch entsprechende vertragliche Vereinbarung übernommen wurden, wie das Anbringen oder Prüfen von Signaturen, Virenprüfungen oder die Auswahl alternativer Transportinfrastrukturen.

2.2.6.1 Enthaltene Anwendungsfälle

2.2.6.1.1 Delegierte Aufgaben durchführen

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
vertragliche Vereinbarungen	2.3.7	29
Beschreibung		
Sender und Empfänger führen Aufgaben durch, die sie durch vertraglich geregelte Delegation vom Autor bzw. Leser erhalten haben. Dies kann bspw. das Anbringen einer Signatur oder das Verschlüsseln auf Inhaltsebene sein oder auch die Durchführung von Schemaprüfungen.		

2.2.6.1.2 Transportkanal auswählen

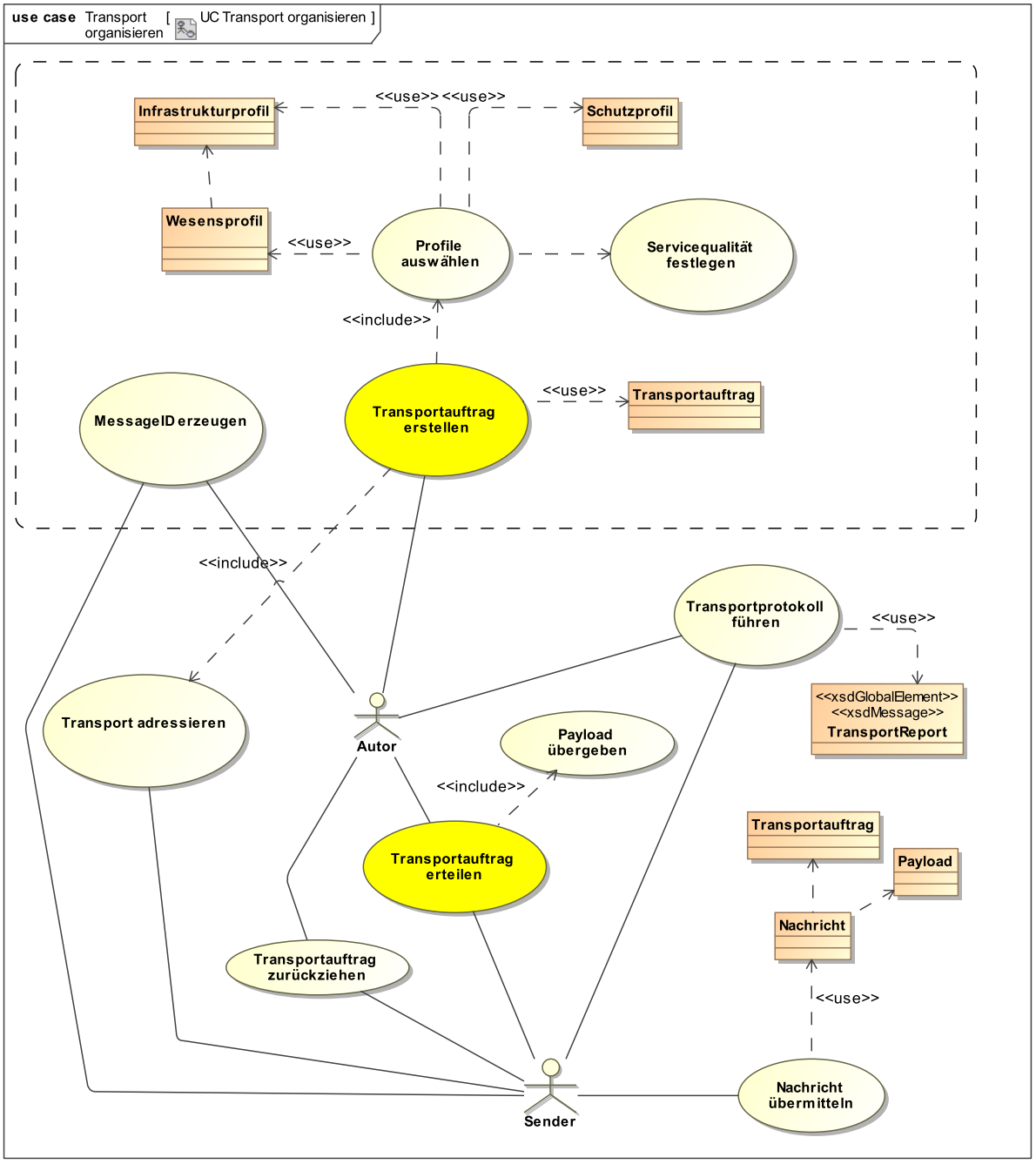
Beschreibung
Für die Übermittlung an den Empfänger wählt der Sender den passenden Transportkanal aus. Diese Wahl trifft er auf der Basis der Angaben im Transportauftrag sowie weiterer Kontextinformationen, die z.B. für den länderinternen Datentransport gelten.

2.2.6.1.3 Löschung durchführen

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
vertragliche Vereinbarungen	2.3.7	29
Beschreibung		
Im Rahmen der Vereinbarungen und Bestimmungen löscht der Sender aufbewahrte Dokumente und Protokolle innerhalb bestimmter Fristen.		

2.2.7 UC Transport organisieren

Abbildung 2.7. Anwendungsfalldiagramm "UC Transport organisieren"



Zur Vorbereitung des Transports gehört die Zusammenstellung und Erteilung des Transportauftrags. Den hierfür notwendigen rechtlichen Rahmen gibt der Gesetz- oder Verordnungsgeber vor. Die rechtlichen Rahmenbedingungen können Einfluss auf die Qualität der Datenübermittlung haben. Bei der Erteilung des Transportauftrags wird der Payload übergeben und die Übermittlung der Nachricht initiiert.

2.2.7.1 Enthaltene Anwendungsfälle

2.2.7.1.1 Servicequalität festlegen

Beschreibung
Das Festlegen der Service Qualität bedeutet, dass die Parameter des Transports aus Sicht der Anforderungen der Fachverantwortlichen definiert werden. Sofern rechtliche Regelungen zur Qualität der Datenübermittlung existieren, sind die Parameter des Transports durch die Fachverantwortlichen entsprechend zu formulieren. Die Festlegung der benötigten Service Qualität erfolgt durch die Auswahl von Profilen. Wichtige Parameter der Service Qualität sind Integrität, Vertraulichkeit, Verfügbarkeit, Transparenz und Intervenierbarkeit.

2.2.7.1.2 Profile auswählen

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
Schutzprofil	2.3.4	28
Infrastrukturprofil	2.3.1	28
Wesensprofil	2.3.8	29
Beschreibung		
Die Profile, die die Anforderungen an die Transportdurchführung erfüllen, werden ausgewählt und im Transportauftrag referenziert. (Wenn kein geeignetes Profil zur Verfügung steht, muss es in einem zu definierenden Prozess erstellt werden.) Die Beschreibungen zu den Profilarten sind den Abschnitten Abschnitt 3.2 auf Seite 32 (Schutzprofile), Abschnitt 3.3 auf Seite 33 (Infrastrukturprofile) und Abschnitt 3.4 auf Seite 34 (Wesensprofile) zu entnehmen. Prototypische bzw. beispielhafte Profile sind im Anhang beigefügt.		

2.2.7.1.3 Transportauftrag erstellen

Eingebundene Use Cases		
Use Case	**Ref.**	**Seite**
Transport adressieren	2.2.7.1.5	23
Profile auswählen	2.2.7.1.2	22
Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
Transportauftrag	2.3.6	29
Beschreibung		
Der Transport einer Nachricht wird vorbereitet durch das Zusammenstellen der Daten des Transportauftrags vorbereitet. Hierzu gehören insbesondere die Adressierung von Autor und Leser, die Festlegung des Wesensprofils und die ID des Transportauftrags (MessageID). Die Parameter des Transportauftrags werden vorgehalten und bei Erteilung des Auftrags übergeben.		

2.2.7.1.4 MessageID erzeugen

Beschreibung
Jeder Transportauftrag ist über seine MessageID (siehe XML-Schemata zu WS-Addressing) identifzierbar. Diese wird durch Autor oder Sender erzeugt und in die Transportauftragsdaten eingetragen.

2.2.7.1.5 Transport adressieren

Beschreibung
Für die Erteilung des Transportauftrags muss der Leser der Nachricht adressiert werden können. Außerdem sollte verifiziert sein, dass er zum Empfang des vorliegenden Nachrichtentyps einen entsprechenden technischen Dienst eingerichtet hat. Die hierfür notwendigen technischen Parameter für die Adressierung des Lesers werden vom Sender ermittelt und in die Daten des Transportauftrags eingetragen.

2.2.7.1.6 Transportauftrag erteilen

Eingebundene Use Cases		
Use Case	**Ref.**	**Seite**
Payload übergeben	2.2.7.1.7	23
Beschreibung		
Der Auftrag zum Transport einer Nachricht wird vom Autor durch den Aufruf der entsprechenden Operation der Sende-Schnittstelle des XTA-Webservice dem Sender erteilt. Diesem Aufruf werden die Daten des Transportauftrags und der zu transportierende Payload mitgegeben. Damit beginnt die Ausführung durch den Sender innerhalb der Transportinfrastruktur. Der Sender initiiert den Transport zum Empfänger gemäß der Parameter des Transportauftrags.		

2.2.7.1.7 Payload übergeben

Beschreibung
Die zu transportierende Nachricht, die ggf. signiert und / verschlüsselt ist, wird durch den Autor mit der Erteilung des Transportauftrags und / oder übergeben und ist damit der Payload des Transports. Die Übergabe geschieht als Paramter beim Aufruf einer XTA-WS-Methode.

2.2.7.1.8 Nachricht übermitteln

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
Nachricht	2.3.2	28
Beschreibung		
Der Sender wertet die Parameter des Transportauftrags aus, um die Nachricht auf dem vorgesehenen Weg dem Empfänger zuzuleiten.		

2.2.7.1.9 Transportprotokoll führen

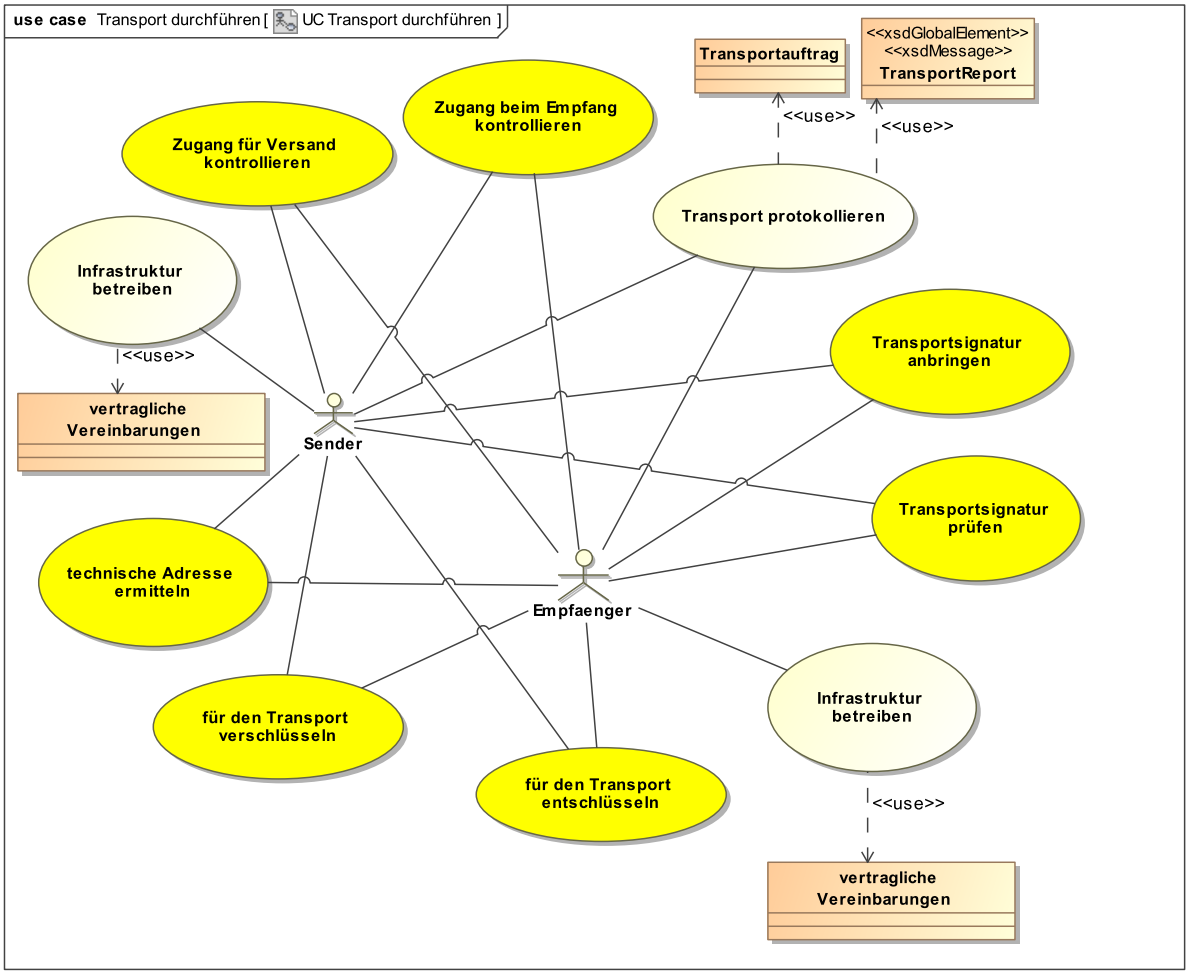
Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
TransportReport	4.5.2.4	78
Beschreibung		
Das Transportprotokoll wird vom Sender geführt, der Ereignisse, Warnungen und Fehler einträgt. Das Transportprotokoll ist jederzeit vom Autor einsehbar. Das Transportprotokoll wird durch das Objekt TransportReport realisiert.		

2.2.7.1.10 Transportauftrag zurückziehen

Beschreibung
Transportaufträge, die noch nicht an den Empfänger weitergegeben wurden, also noch offen sind, können bei Bedarf zurückgezogen werden. Dies ist Aufgabe des Autors. Für das Zurückziehen stellt der Sender dem Autor eine Funktionalität zur Verfügung.

2.2.8 UC Transport durchführen

Abbildung 2.8. Anwendungsfalldiagramm "UC Transport durchführen"



Der Austausch von Nachrichten wird durch die Ausführung von Request/Response-Protokollen wie z.B. SOAP über HTTP realisiert. Transport-Signaturen und -Verschlüsselung werden entsprechend angewendet.

2.2.8.1 Enthaltene Anwendungsfälle

2.2.8.1.1 Infrastruktur betreiben

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**

Verwendete Artefakte		
vertragliche Vereinbarungen	2.3.7	29

Beschreibung

Der Sender ist gemäß Transportauftrag für die Abwicklung des Transports zuständig. Der Sender unterhält dafür die Infrastruktur und gibt dem Autor einen entsprechenden Zugang.

Der Empfänger ist vom Leser mit der Entgegennahme von Nachrichten beauftragt. Anmerkung: Ein Intermediär ist Bestandteil der Empfänger-Infrastruktur.

Der Empfänger unterhält für die Entgegennahme von Nachrichten die Infrastruktur.

2.2.8.1.2 Transportsignatur anbringen

Beschreibung

Der Sender bringt je nach geltender Policy, die im Wesensprofil abgebildet ist, für den jeweiligen Transportkanal ggf. die Transportsignatur an.

2.2.8.1.3 für den Transport verschlüsseln

Beschreibung

Der Sender verschlüsselt die zu transportierende Nachricht je nach Policy für den jeweiligen Transportkanal ggf. für den Empfänger.

2.2.8.1.4 technische Adresse ermitteln

Beschreibung

Der Sender prüft, ob für den Leser ein Zugang eröffnet ist. Der Sender ermittelt die technische Adresse des Lesers anhand dessen fachlicher Adresse. Er verwendet hierfür ein Verzeichnis wie DVDV oder S.A.F.E.

Der Sender stellt, falls nötig, dem Autor die technischen Attribute des Lesers zur Verfügung.

2.2.8.1.5 Zugang für Versand kontrollieren

Beschreibung

Der Zugang zur Transportinfrastruktur und der Zugang zu den angeforderten Diensten wird von den Transporteuren kontrolliert:

Prüfung der Identität des Autors durch den Sender:

- Der Sender ist für die Authentifizierung des Autors zuständig, d. h. er prüft die Identität des Autors. Anmerkung: Der Sender überprüft, ob ihm die Authentisierungsinformationen des Autors bekannt sind.
- Der Sender ist verpflichtet, die Angaben der Authentifizierung auf Konsistenz mit den Absenderangaben des Transportauftrages zu prüfen. Anmerkung: Durch diese Prüfung wird geklärt, ob die authentisierte Behörde in diesem Fachkontext mit dieser Behördenidentität (z.B. AGS:12343123 für Oldenburg im Meldewesen) auftreten darf.
- Abgrenzung: Der Sender ist nicht dafür zuständig, die fachliche Zuständigkeit des Autors für den Inhalt der Nachricht zu prüfen. Dies geschieht durch den Leser.

2.2.8.1.6 Transport protokollieren

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
Transportauftrag	2.3.6	29
TransportReport	4.5.2.4	78

Beschreibung
Sender und Empfänger protokollieren die Ereignisse entsprechend der Vorgaben aus dem Transportauftrag.

2.2.8.1.7 Zugang beim Empfang kontrollieren

Beschreibung
Der Zugang zur Transportinfrastruktur und der Zugang zu den angeforderten Diensten wird von den Transporteuren kontrolliert: Prüfung der Identität des Lesers durch den Empfänger: • Der Empfänger ist für die Authentifizierung des Lesers zuständig, d. h. er hat die Identität des Lesers zu prüfen. Anmerkung: Die Prüfung dient der Zugriffskontrolle im Kontext der Autorisierung nach Absprache mit dem Leser. • Der Empfänger ist für die Prüfung zuständig, ob die Identität des Lesers (Authentisierung gegenüber dem Empfänger) konsistent ist mit der Identität des Lesers für die Fachkommunikation im Rahmen desTransportauftrags. • Abgrenzung: Der Empfänger ist nicht dafür zuständig, die fachliche Zuständigkeit des Lesers für den Inhalt der Nachricht zu prüfen. Dies geschieht durch den Leser.

2.2.8.1.8 Transportsignatur prüfen

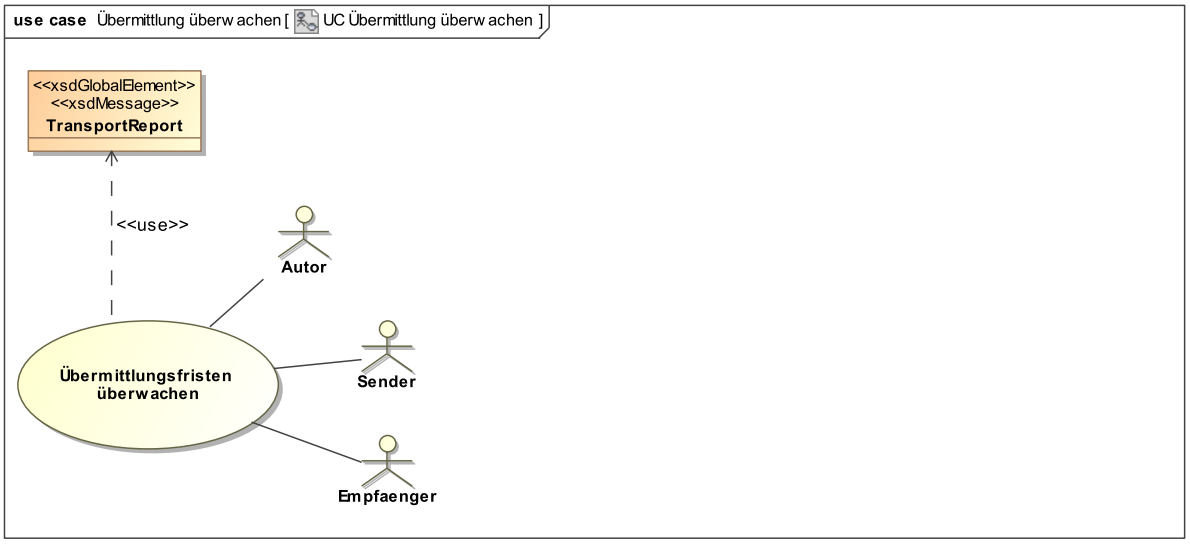
Beschreibung
Der Empfänger prüft je nach geltender Policy, die im Wesensprofil abgebildet ist, für den jeweiligen Transportkanal ggf. die Transportsignatur.

2.2.8.1.9 für den Transport entschlüsseln

Beschreibung
Der Empfänger entschlüsselt je nach Policy für den jeweiligen Transportkanal ggf. die Transportverschlüsselung für den Leser.

2.2.9 UC Übermittlung überwachen

Abbildung 2.9. Anwendungsfalldiagramm "UC Übermittlung überwachen"



Der Autor überwacht Erfolg und Fristen der Zustellung seiner Transportaufträge an Empfänger und Leser.

Dies geschieht durch Auswertung der Protokolleinträge.

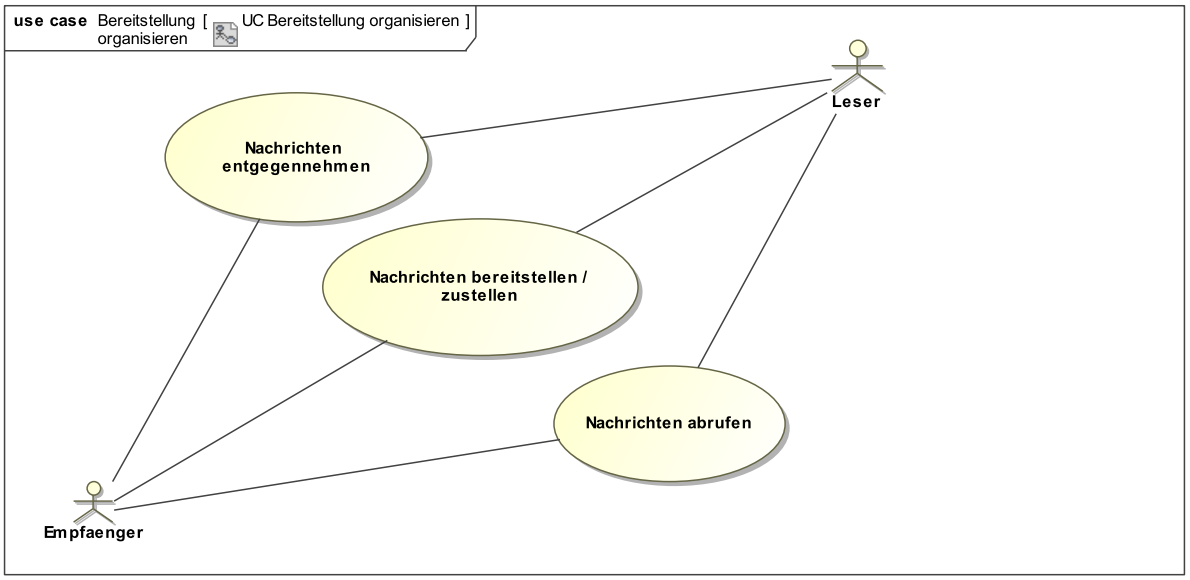
2.2.9.1 Enthaltene Anwendungsfälle

2.2.9.1.1 Übermittlungsfristen überwachen

Verwendete Artefakte		
Artefakt	**Ref.**	**Seite**
TransportReport	4.5.2.4	78
Beschreibung		
Der Autor ist für die Überwachung der Übermittlung und für die Einhaltung der (rechtlich- organisatorischen Vorgaben für) Übermittlungfristen der Nachricht an den Empfänger bzw. Leser zuständig. Für diese Aufgabe nimmt er Einsicht in die vom Sender zu führenden Protokolle.		

2.2.10 UC Bereitstellung organisieren

Abbildung 2.10. Anwendungsfalldiagramm "UC Bereitstellung organisieren"



Der Empfänger stellt dem Leser die nötigen Dienste nach Bedarf bereit. Dazu zählt die Nachrichtenkommunikation im engeren Sinne (Nachrichten bereitstellen oder zustellen), aber auch Kontroll- und Prüfdienste, die die Kommunikation absichern.

2.2.10.1 Enthaltene Anwendungsfälle

2.2.10.1.1 Nachrichten entgegennehmen

Beschreibung
Der Empfänger ist vom Leser mit der Entgegennahme von Nachrichten beauftragt. Der Empfänger nimmt die Nachrichten vom Sender an der Schnittstelle zur Transportinfrastruktur entgegen und verfährt mit ihr entsprechend der Vorgaben des Transportauftrags.

2.2.10.1.2 Nachrichten bereitstellen / zustellen

Beschreibung
Je nach Vorgaben im Transportauftrag verfährt der Empfänger mit den entgegengenommenen Nachrichten: Er stellt sie für den Abruf durch den Leser bereit bzw. stellt sie direkt zu (in synchronen Kommunikationsszenarien).

2.2.10.1.3 Nachrichten abrufen

Beschreibung
Der Leser ruft die bereitgehaltenen Nachrichten vom Empfänger ab. Asynchrones Kommunikationszenario: Der Leser ist verpflichtet, Nachrichten und Transportinformationvom Empfänger abzurufen oder entgegenzunehmen. Synchrones Kommunikationsszenario: Der Leser bedient die Anfrage des Autors unmittelbar.

2.3 Zentrale Artefakte beim Nachrichtenaustausch

2.3.1 Infrastrukturprofil

Artefakt: *Infrastrukturprofil*

Ein Infrastrukturprofil ist eine Zusamenstellung vom IT-Planungsrat betriebenen IT-Komponenten (siehe Abschnitt 3.3 auf Seite 33).

2.3.2 Nachricht

Artefakt: *Nachricht*

Die Nachricht besteht aus dem Transportauftrag mit dem zugehörigen Payload.

2.3.3 Payload

Artefakt: *Payload*

Der Payload ist der fachliche Inhalt der Nachricht, der vom Autor für den Leser erstellt wird. Er umfasst die Gesamtheit der zu übermittelnden Informationen einschließlich Nachrichtenkopf mit den Angaben zu Absender, Adressat, Thema und Datum.

Der Payload kann vom Autor für den Leser verschlüsselt werden. Deshalb muss der Sender seine Aufgaben mit ausschließlicher Kenntnis des Transportauftrags erfüllen können. Wenn XÖV-Nachrichten zu übermitteln sind, ist der Payload eine (komplette) XÖV-Nachricht.

2.3.4 Schutzprofil

Artefakt: *Schutzprofil*

Ein Schutzprofil wird auf der Basis einer Schutzbedarfsfeststellung ausgewählt. Es bündelt technische und organisatorische Anforderungen, die erfüllt sein müssen, um den Schutzbedarf abzudecken, siehe auch Abschnitt 3.2 auf Seite 32.

2.3.5 Spezifikation Fachstandard

Artefakt: *Spezifikation Fachstandard*

Die Spezifikation des Fachstandards ist ein Regelwerk, das die kollaborativen Prozesse im Kontext des Nachrichtenaustauschs definiert.

Die Spezifikation liegt immer in einer anzuwendenden Version vor, durch die die Regeln des Datenaustausches für Syntax und Semantik definiert sind.

2.3.6 Transportauftrag

Artefakt: *Transportauftrag*

Der Transportauftrag enthält alle erforderlichen Angaben, um die fachliche Nachricht gemäß der Intention des Autors zum Empfänger zu transportieren. Über den Transportauftrag wird die Qualität der Protokollierung der beteiligten Systeme unter Angabe des anzuwendenden Schutzprofils gesteuert. Jeder Transportauftrag ist eindeutig identifizierbar. Der Transportauftrag wird durch das Objekt MessageMetaData im XTA-WS repräsentiert (vgl. Kapitel 4 auf Seite 37 und zur beispielhaften Darstellung der Inhalte eines Transportauftrags die Schutzprofile siehe Abschnitt B.1 auf Seite 97 und Wesensprofile siehe Abschnitt B.3 auf Seite 102).

2.3.7 vertragliche Vereinbarungen

Artefakt: *vertragliche Vereinbarungen*

Sender und Empfänger führen ihre Dienste im Auftrag der Fachverantwortlichen (Autor und Leser) aus. Dienstleistungsverträge mit entsprechenden Vereinbarungen zu den Auftragskonditionen regeln den Ablauf der Kooperation.

2.3.8 Wesensprofil

Artefakt: *Wesensprofil*

Ein Wesensprofil ist für die einzelnen fachlichen Domänen zugeschnitten. Es beschreibt für jedes Schutzprofil die benötigte Konfiguration der Infrastukturkomponenten des vorgegebenen Infrastrukturprofils (siehe Abschnitt 3.4 auf Seite 34).

2.3.9 XML Schema

Artefakt: *XML Schema*

Das in der Sprache XML Schema definierte Artefakt zur Version des Fachstandards enthält einen Teil der Regeln der Spezifikation. Es lässt sich verwenden, um mechanisch die Konformität der zu übermittelnden Nachricht mit diesen Regeln zu prüfen

3 XTA-Profilkonzept

Das XTA-Profilkonzept ist die Basis der XTA-Konformität, deren Ziel es ist, kontrollierbare Bedingungen zwischen Kommunikations-Endpunkten zu schaffen, so dass diese von der öffentlichen Verwaltung leicht nachvollziehbar und überprüfbar sind:

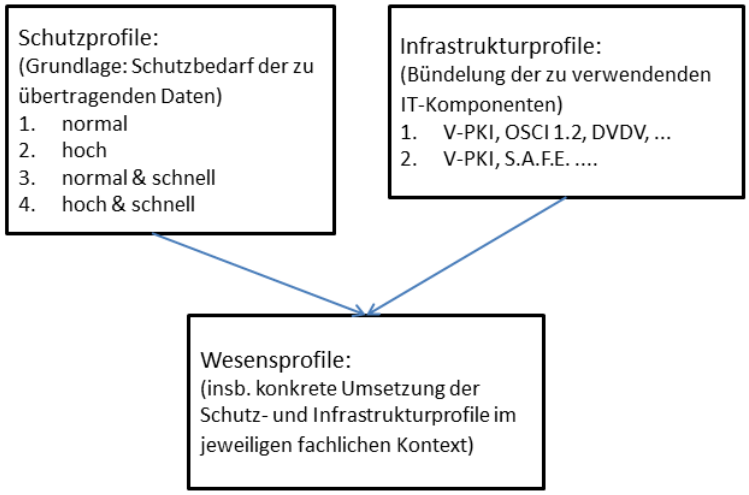
Die XTA-Konformität bezeichnet eine Menge von prüfbaren Anforderungen bzgl. IT-Sicherheit und Datenschutz an die am Transport beteiligten Komponenten. Der rechtliche Rahmen hierfür wird vom Gesetz- oder Verordnungsgeber vorgegeben. Der Gesetz- oder Verordnungsgeber legt fest, unter welchen Umständen eine Datenübermittlung zulässig ist. Er beschreibt Zweck und Umfang der Datenübermittlung. Der Gesetz- oder Verordnungsgeber gibt auch allgemeine Rahmenbedingungen vor, die für das gesamte Verfahren gelten. Hierbei können Anforderungen enthalten sein, die Einfluss auf die Qualität der Datenübermittlung haben. Sofern rechtliche Regelungen zur Qualität der Datenübermittlung existieren, sind diese zu beachten.

Die Gesamtheit der Anforderungen wird bereits heute von unterschiedlichen Verantwortlichen formuliert und ist dabei für vergleichbare Szenarien meist ähnlich: Dies gilt sowohl für die generischen Anforderungen, die beispielsweise an die IT-Sicherheit und an die Verfügbarkeit gestellt werden, wie auch für die konkrete Umsetzung dieser Anforderungen bzgl. der Wahl der IT-Komponenten und deren Konfiguration.

Diese Beobachtung führt zur Definition von XTA-Profilarten, durch die eine Vereinheitlichung der Anforderungen – und damit eine Standardisierung - erreicht werden soll. Diese Standardisierung stellt die Grundlage für die XTA-Konformität dar und soll so die Nachvollziehbarkeit der öffentlichen Verwaltung bzgl. der Einhaltung der Anforderungen an IT- Sicherheit und Datenschutz erleichtern.

3.1 Übersicht der Profilarten

Abbildung 3.1. Übersicht der Profilarten



- **Schutzprofile**: In den Schutzprofilen wird der Schutzbedarf einer Anwendung auf der Grundlage der zu verarbeitenden Daten festgelegt. Die Festlegung kann entweder durch rechtliche Regelungen vorgegeben sein oder im Nachgang durch die Verfahrensverantwortlichen (z.B. über ein Fachgremium) erfolgen. Die Festlegung wird dabei anhand einer Schutzbedarfsfeststellung in Anlehnung an die Systematik des BSI –Grundschutzes vorgenommen und mündet in der Wahl eines der (vor-)definierten Schutzprofile. Im Rahmen des Projektes XTA werden prototypische Schutzprofile erarbeitet, siehe Abschnitt 3.2 auf Seite 32.
- **Infrastrukturprofile**: In den Infrastrukturprofilen werden die IT-Infrastrukturkomponenten, für die der IT-Planungsrat verantwortlich ist, derart zusammengefasst / gebündelt, wie sie typischerweise in unterschiedlichen Bereichen der öffentlichen Verwaltung eingesetzt werden. Im Rahmen des Projektes XTA werden prototypische Infrastrukturprofile erarbeitet, siehe Abschnitt 3.3 auf Seite 33.
- **Wesensprofile**: Die Wesensprofile nehmen die Vorgaben aus Schutzprofilen und Infrastrukturprofilen auf. Sie enthalten für den jeweiligen fachlichen Kontext die Konfiguration der durch die Infrastrukturprofile vorgegebenen IT-Komponenten und die Umsetzung der Schutzprofile.

  Im Wesensprofil können weitere Attribute abgelegt werden, die nicht unmittelbar aus den Vorgaben der Schutz- und Infrastrukturprofile abgeleitet werden können, aber für die Durchführung des Transports benötigt werden.

  Die Fachgremien sind für die Erstellung und Pflege der Wesensprofile verantwortlich. Die Definition der Wesensprofile soll mit einer Standardisierung der durch den rechtlichen Rahmen vorgegebenen Attribute einhergehen, so dass die Anzahl der Wesensprofile gering, und damit der Aufwand der Pflege der Profile überschaubar bleibt. Im Rahmen des Projektes XTA werden prototypische Wesensprofile beispielhaft erstellt, siehe Abschnitt 3.4 auf Seite 34.

Die abgestimmten Profile werden jeweils im XRepository abgelegt und von der KoSIT betrieben.

3.2 Schutzprofile

Der Schutzbedarf wird durch die rechtlichen Rahmenbedingungen vorgegeben. Diese gelten für das gesamte Verfahren, nicht nur für den Daten-Transport selbst. Die Festlegung wird anhand einer Schutz-

bedarfsfeststellung in Anlehnung an die Systematik des BSI – Grundschutzes für die Schutzziele der IT–Sicherheit und für die Schutzziele des Datenschutzes vorgenommen.

Basierend auf dieser Schutzbedarfsanalyse kann für den Daten-Transport eine abweichende Einstufung erfolgen, wenn das Schutzniveau der zu übertragenden Daten entsprechend eingestuft wurde.

Die Schutzbedarfsfeststellung mündet in der Auswahl eines Schutzprofils. (Wenn kein geeignetes Schutzprofil vorhanden ist, muss über einen noch zu definierenden Prozess ein neues Schutzprofil erstellt werden.)

Die durch das Schutzprofil resultierenden Maßnahme-Empfehlungen umfassen technische und organisatorische Maßnahmen. Zuständig für diese Operationalisierung der Schutzziele ist im Rahmen eines gemeinsamen Verfahrens jede beteiligte Stelle für den von ihr zu verantwortenden Teil: Die Fachverfahrensverantwortlichen erarbeiten, ggf. mit Unterstützung Dritter, die die Datenverarbeitung im Auftrag übernommen haben, einheitliche und überprüfbare Vorgaben.

3.2.1 Schutzprofile I bis IV

Aufgrund der besonderen Anforderungen an eine Risikoanalyse und der daraus abzuleitenden speziellen Maßnahmen sind “sehr hohe” Schutzbedarfe nicht Gegenstand einer allgemeinen fach- und ebenen-übergreifenden Standardisierung und werden deshalb im Projektkontext nicht betrachtet.

Es wird angenommen, dass vier Schutzprofile ausreichen, um die praxisrelevanten Anforderungen abzudecken. Diese prototypischen Schutzprofile sind hier im Überblick und im Anhang ausführlich dargestellt:

- **Schutzprofil I „normal“**: Das Schutzprofil bietet einen "normal" abgesicherten Nachrichtenaustausch auf der Basis des IT-Grundschutzes. Es dürfen personenbezogene Daten im geringen Umfang und mit geringem Schutzwert (z.B. öffentliche Adressen) transportiert werden.
- **Schutzprofil II „hoch“**: Das Schutzprofil verschärft das Schutzprofil I, so dass ein "hoher" Schutzbedarf bzgl. Vertraulichkeit und Integrität gewährleistet werden kann. Dieses Schutzprofil ist für hoch schutzwürdige Daten und allgemein für persönliche Daten geeignet.
- **Schutzprofil III „normal und schnell“**: Das Schutzprofil ist für den “normalen” Datenaustausch mit hohen Anforderungen an die Verfügbarkeit geeignet.
- **Schutzprofil IV „hoch und schnell“**: Das Schutzprofil vereint die Eigenschaften der Schutzprofile “hoch” und “normal und schnell”, um Daten mit hohem Schutzbedarf schnell und / oder mit hoher Verfügbarkeit übermitteln zu können

Die Schutzprofile enthalten jeweils Aussagen zu den Schutzzielen der IT-Sicherheit (Vertraulichkeit, Integrität, Verfügbarkeit), sowie zu den Schutzzielen des Datenschutzes (Nichtverkettbarkeit, Intervenierbarkeit, Transparenz). Zusätzlich können weitere Merkmale, die nicht unmittelbar den Schutzzielen zuzuordnen sind, enthalten sein, wenn sie für die Übertragung benötigt werden. Insbesondere können rechtliche Vorgaben existieren, die Vorgaben für die Qualität der Datenübermittlung enthalten. Dies betrifft beispielsweise die maximale Durchlaufzeit, die für einen Datentransport akzeptiert wird, sowie die Anforderung der Eignung für ein Dialogverfahren.

3.3 Infrastrukturprofile

Der IT-Planungsrat hat die Aufgabe, die vom Gesetz- oder Verordnungsgeber definierten Schutzbedarfe insbesondere hinsichtlich des Nachweises der Interoperabilität, Rechtskonformität, Sicherheit und Wirtschaftlichkeit einer Datenübermittlung standardisiert umsetzbar zu machen. Daher betreibt der IT-Planungsrat IT-Infrastrukturkomponenten (Verzeichnisdienste, Public Key Infrastrukturen, …) und bietet Standards für die sichere Datenübermittlung.

In der Praxis ergänzen sich die IT-Komponenten zu sinnvollen Konstellationen, die in Infrastrukturprofilen zusammengefasst / gebündelt werden sollen: Sie sollen in diesen Kombinationen geeignet sein, um die fachlichen Anforderungen an den Datenaustausch angemessen befriedigen zu können.

Rechtliche Regelungen, wie beispielsweise das E-Government-Gesetz, können dazu führen, dass weitere Infrastrukturprofile erstellt oder bestehende Infrastrukturprofile angepasst werden müssen.

Da für die landesinterne Datenübermittlung grundsätzlich auch IT-Komponenten genutzt werden können, die nur innerhalb des Landes verfügbar sind, können landeseigene, funktional äquivalente Infrastrukturprofile erstellt werden. Diese Option wird im Projekt XTA nicht weiter betrachtet, da sie außerhalb des Fokus des IT-Planungsrates liegt.

Im Rahmen des Projektes XTA wurden zwei beispielhafte Infrastrukturprofile erarbeitet:

Eines erscheint für die Innenverwaltung geeignet. In ihm sind insbesondere die Komponenten OSCI-Transport 1.2, OSCI-Intermediär, DVDV und die PKI des Bundes gebündelt.

Das zweite, für die Justiz geeignet erscheinende Infrastrukturprofil enthält zum ersten starke Überschneidungen, verwendet aber insbesondere einen anderen Verzeichnisdienst.

Zukünftige Einsatzbereiche der in den Infrastrukturprofilen genannten IT-Komponenten hängen von deren Weiterentwicklung ab.

3.4 Wesensprofile

Ein Wesensprofil ist für die einzelnen fachlichen Domänen des E-Government (z.B. Ausländerwesen, Finanzwesen, Justizwesen, Meldewesen, Personenstandswesen) zugeschnitten. Es beschreibt für jedes Schutzprofil die jeweils benötigte Konfiguration der Infrastukturkomponenten des vorgegebenen Infrastrukturprofils. Das Wesensprofil enthält damit die notwendigen Parameter, um die technische Interoperabilität zwischen den an der Datenübermittlung Beteiligten herzustellen.

Im Wesensprofil können außerdem Attribute definiert werden, die nicht unmittelbar aus den Vorgaben aus Schutz- und Infrastrukturprofil abgeleitet werden können, aber ergänzend für eine standardisierte Datenübermittlung benötigt werden.

Die Definition der Wesensprofile erfolgt durch die zuständigen Fachgremien auf der Grundlage der Vorgaben aus den Schutzprofilen und den Infrastrukturprofilen.

Die erstellten Wesensprofile werden im XRepository abgelegt. Eine Hinterlegung als standardisiertes XML erscheint sinnvoll, um eine maschinenlesbare Auswertung einer lokalen Kopie zu ermöglichen. Für einen direkten Zugriff auf das im XRepositpry abgelegte Wesensprofil ist das XRepository derzeit nicht ausgelegt und scheint nicht notwendig bzw. hilfreich.

Die Angabe der zu nutzenden Profile erhält der Sender vom Autor durch den Transportauftrag.

Die Updatezyklen der Wesensprofile sind nicht an denen der XÖV-Standards gekoppelt.

Für die landesinterne Datenübermittlung können weitere, funktional äquivalente Wesensprofile definiert werden, durch die insbesondere abweichende Infrastrukturkomponenten genutzt werden können. Diese landesspezifischen Wesensprofile liegen nicht in der Zuständigkeit des IT-Planungsrates und werden daher nicht im Projekt XTA betrachtet.

Wenn funktional äquivalente Wesensprofile in einem Land angeboten werden, liegt die Wahl des zu nutzenden Wesensprofils beim Sender, siehe auch Rollenmodell B1.1. Form und Umfang des Nachweises der inhaltlichen Äquivalenz der landesspezifischen Wesensprofile sind zu klären.

3.4.1 Attribute eines Wesensprofils

Zu einem Wesensprofil gehören insbesondere folgende Attribute:

1. Allgemeine Angaben:
   a. Name (z.B. “XMeld”)
   b. Version (z.B. “1.0”)

   c. Angabe des Infrastrukturprofils
   d. Max. Durchlaufdauer (z.B. “3 Werktage”)
   e. Dialogverfahren (“nein” (meint asynchrone Übermittlung), "ja")
   f. zu verwendene Netze (“Internet/DOI”; “DOI”)
2. Angaben zum zu verwendendem Verzeichnisdienst:
   a. Adressierungsverzeichnisdienst (z.B. “DVDV” oder "S.A.F.E.")
   b. (bei DVDV:) Bezeichnung des Intermediärs (z.B. “Testa-Intermediär/ Internet-Intermediär”)
   c. (bei DVDV:) Bezeichnung Empfänger (“Testa-Adressee / Internet-Adressee”)
   d. (bei DVDV:) Bezeichnung Leser (“Reader”)
   e. Zertifikatsprüfung (z.B. “V-PKI”)
3. Für die im Wesensprofil enthaltenen Schutzprofile:
   a. Angaben zur Verschlüsselung
      i. für Transportebene (z.B. “AES 256”)
      ii. für Payload (z.B. “AES 256”)
   b. Angaben zur Signatur
      i. für Transportebene (z.B.: für fortgeschrittene Signatur: SHA 256, RSA)
      ii. für Payload (z.B. “optional” (d.h. “Schriftformerfordernis = nein”)

4 Spezifikation des XTA-Webservice

4.1 Überblick

Die Kommunikation zwischen Fachverfahren und Transportverfahren – also aus der Sicht des Rollenmodells die Kommunikation zwischen Autor und Sender - wird über eine Webservice-Schnittstelle realisiert, die XTA-Webservice („XTA-WS“) genannt wird. Diese Schnittstelle ist der einheitliche Zugang eines Fachverfahrens zu einer XTA-konformen Transport-Infrastruktur. Gleichzeitig entkoppelt sie das Fachverfahren von der technischen Komplexität der Transportprozesse. Der XTA-Webservice ist damit unabhängig von den beim Transport verwendeten Kommunikationsprotokollen (z.B. HTTPS oder OSCI-Transport).

Um diesen einheitlichen Zugang zu nutzen, wird in die Fachverfahren des Autors und des Lesers jeweils ein XTA-Webservice-Client eingebunden. Dieser ist in der Lage, die Funktionalitäten des XTA-WS, der von einem XTA-konformen Transportverfahren angeboten wird, aufzurufen.

Der XTA-Webservice wird durch eine WSDL technisch beschrieben. Sie ist als Anlage der Spezifikation beigefügt. Der XTA-WS ist separat vom Fachverfahren implementiert und wird vom Sender zur Verfügung gestellt und betrieben. Der Betrieb kann zentral durch eine Clearingstelle (Vermittlungsstelle, Nachrichtenbroker) oder lokal durch eine Kommunikationssoftware (z.B. OSCI-Client) erfolgen.

4.2 Rahmenbedingungen für die XTA-WS-Schnittstelle

4.2.1 XTA-WS als OSCI 2 Profilierung

Der XTA-Webservice ist als OSCI 2 Profilierung realisiert. OSCI 2 wurde im Auftrag der öffentlichen Verwaltung auf der Basis internationaler Web Services Standards entwickelt.

OSCI 2 ist selbst eine Profilierung: Es basiert auf dem WS-Stack, der eine Menge von internationalen, auf Webservices basierenden Protokollstandards darstellt. Der WS-Stack bietet konfigurierbare Bausteine. Durch OSCI 2 wird die Konfiguration vorgenommen.

So wird durch die OSCI 2-Profilierung zum einen die Interoperabilität mit auf Webservice basierenden Lösungen sichergestellt. Zum anderen bewirkt die Profilierung eine Einengung auf die Anforderungen der deutschen Verwaltung an eine vertrauliche, verlässliche und rechtsverbindliche Kommunikation, wie sie u.a. durch das BSI gefordert werden.

In dieser XTA-Dokumentation werden die Methoden, die aus OSCI 2 stammen, dokumentiert, so dass ihre Funktion und Aufgabe innerhalb des XTA-WS deutlich werden. Die mit der Implementierung des XTA-WS betrauten Personen werden (zusätzlich) auf die OSCI 2- Spezifikation (in der Version 2.01) verwiesen. (Die im Projekt betrachteten Szenarien waren Anlass für Änderungen an der OSCI 2-Spezifikation, die in der Version OSCI 2.01 mündeten.)

4.2.2 Authentifizierung und Autorisierung

Die gegenseitige Authentifizierung und Autorisierung der Kommunikationspartner sind wesentliche Aufgaben, die für die Erreichung des geforderten Sicherheitsniveaus geleistet werden müssen. Im Modell der Rollen und Verantwortlichkeiten werden sie im Rahmen der wechselseitigen Prüfung der Identitäten der Akteure benannt.

So müssen sich der Autor gegenüber seinem Sender und der Leser gegenüber seinem Empfänger authentisieren, wobei die hierfür notwendigen Daten verschlüsselt übertragen werden müssen. Diese beiden Ziele werden durch die Verwendung von TLS erreicht, wobei der Autor bzw. Leser sich durch die Verwendung eines ihm zugeordneten Client-Zertifikats ausweist.

Der Sender bzw. Empfänger überprüft die Berechtigung des Clients zum Zugriff (Authentifizierung). Reicht die Angabe des Zertifikats nicht zur eindeutigen Identifikation aus, muss der Sender bzw. der Empfänger weitere Angaben beim Verbindungsaufbau (im Header der Soap Nachricht) mitgeben.

Für die Gegenrichtung, also für die Authentisierung des Senders durch den Autor benötigt dieser eine lokale Konfiguration für den Zugriff. Hierfür nutzt der Sender in der TLS Verbindung das ihm zugeordnete Zertifikat. Der Autor vergleicht dieses mit den, in der lokalen Konfiguration gespeicherten Angaben.

Entsprechend prüft der Leser die Authentisierung des Empfängers.

Zur Authentifizierung werden von WS-Interoperability grundsätzlich nur zertifizierte Webservicestandards zugelassen. Die vorliegende XTA-Schnittstellenversion erlaubt ausschließlich die SSL-Client-Authentifizierung.

Ob dem authentifizierten Benutzer der Zugang zum Webservice gewährt werden darf, entscheidet der XTA-WS im Rahmen der Autorisierung.

Im Regelfall wird der Zugriff auf einen Account beim XTA-Betreiber über das Zertifikat autorisiert. Wenn dem Zertifikat mehrere Accounts zugeordnet sind, also eine mandantenorientierte Organisation erfolgt, werden optionale SOAP-Attribute befüllt, um gemeinsam mit dem Zertifikat einen bestimmten Account beim XTA-Betreiber zu identifizieren. Die dafür erforderlichen Parameter werden im XTA-Header in den Elementen (<AbsenderPraefix> und <AbsenderKennung>) mitgegeben, siehe Abschnitt 4.4.1.6.1 auf Seite 47.

4.3 Umsetzung der fachlichen Anforderungen in der XTA-WS-Schnittstelle (beispielhafte Darstellung)

Die Funktionen des XTA-Webservice (XTA-WS) sind von den Anforderungen der Rollen und Anwendungsfälle (vgl. Kapitel 2 und Anhang A) abgeleitet. Um dies zu veranschaulichen, werden hier die Aufgaben und Abläufe zwischen Autor und Sender bzw. Empfänger und Leser (mit Verweisen auf die Sätze des Rollenmodells in Anhang A) beispielhaft beschrieben und der Bezug zu den Methoden des XTA-WS hergestellt.

Ergänzt wird diese Dokumentation durch Beispielcode im Anhang D, *Beispielcode* für den asynchronen und synchronen Versand und Empfang und für den Rückruf von Nachrichten.

Die gegenseitige Authentifizierung der Kommunikationspartner ist eine sich wiederholende Aufgabe, die über Übermittlung und Überprüfung von Client-Zertifikaten erfolgt. Dieser Arbeitsschritt wird in den beispielhaften Darstellungen jeweils vorausgesetzt und nicht explizit erwähnt.

4.3.1 Aufgaben des Autors

Die Aufgaben eines Autors bestehen im Versenden von Nachrichten und der Überwachung des Nachrichtentransports (Anhang A Rollenmodell A 1.1, A 5.2, A 8.1, A 8.2):

- Asynchroner Versand einer Nachricht (siehe Abschnitt 4.3.1.1 auf Seite 39)
- Synchroner Versand einer Nachricht (siehe Abschnitt 4.3.1.2 auf Seite 39)
- Rückruf einer Nachricht (siehe Abschnitt 4.3.1.3 auf Seite 40)

Vor der ersten Übertragung sind technische und organisatorische Rahmenbedingungen zu schaffen (Anhang A Rollenmodell A 10.1, A 9.2, A 8.1), die hier vorausgesetzt werden.

4.3.1.1 Asynchroner Versand einer Nachricht

Bei einem asynchronen Versand beauftragt der Autor den Sender mit der Übertragung einer Nachricht. Zu einem späteren Zeitpunkt prüft der Autor den Status der Übertragung, bis diese eindeutig (durch Erfolg oder Misserfolg) beendet wurde.

1. Sendebereitschaft und -berechtigung

   Der erfolgreiche Versand setzt voraus, dass der Autor in der Lage ist, zu senden, und auch, dass der Leser in der Lage und berechtigt ist, zu empfangen (Anhang A Rollenmodell A 2.2).

   XTA Funktionalitäten:
   - Verbindung Autor - Sender (siehe Abschnitt 4.4.1.1 auf Seite 42)
   - Erreichbarkeit Autor - Leser (siehe Abschnitt 4.4.1.2 auf Seite 43)
2. Erstellung Nachricht

   Der Autor erstellt die zu übertragende Nachricht (Anhang A Rollenmodell A 1.3, A 2.1, A 2.3, A 3.1, A 4.1).
3. Erstellung Transportauftrag

   Der Autor legt die Metadaten fest, die den Transportauftrag beschreiben, z. B. fachliche Adressierung, Service Qualitäten und den eindeutigen Identifikator (MessageId) des Transportauftrags (Anhang A Rollenmodell A 5.2, A 6.1, A 7.1, A7.2)

   XTA Funktionalitäten:
   - Erzeugung eines eindeutigen Identifikators (siehe Abschnitt 4.4.1.5 auf Seite 46)
4. Asynchroner Versand

   Der Autor übergibt die Nachricht zusammen mit dem Transportauftrag für den Versand an den Sender (Anhang A Rollenmodell A 1.2, A 5.1).

   XTA Funktionalitäten:
   - Asynchroner Versand (siehe Abschnitt 4.4.1.2 auf Seite 43)
5. Überwachung des Versands

   Der Autor überprüft, ob der Versand der Nachricht erfolgreich durchgeführt werden konnte (Anhang A Rollenmodell A 8.1), z. B. ob die verwendeten Zertifikate gültig waren. Im Falle eines Misserfolgs muss er ggf. Eskalationsmaßnahmen ergreifen.

   XTA Funktionalitäten:
   - Abruf eines Transportprotokolls (siehe Abschnitt 4.4.1.3 auf Seite 45)

4.3.1.2 Synchroner Versand einer Nachricht

1. Sendebereitschaft und -berechtigung

   Für den Versand einer Nachricht ist es wichtig, dass nicht nur der Autor in der Lage ist, zu senden, sondern auch der Leser in der Lage und berechtigt ist, zu empfangen (Anhang A Rollenmodell A 2.2).

   XTA Funktionalitäten:
   - Verbindung Autor → Sender (siehe Abschnitt 4.4.1.1 auf Seite 42)

- Erreichbarkeit Autor → Leser (siehe Abschnitt 4.4.1.2 auf Seite 43)

2. Erstellung Nachricht

   Der Autor erstellt die Nachricht, die übertragen werden soll (Anhang A Rollenmodell A 1.3, A 2.1, A2.3, A 3.1, A 4.1).

3. Erstellung Transportauftrag

   Der Autor legt die Metadaten fest, die den Transportauftrag beschreiben, z. B. fachliche Adressierung, Service Qualitäten und den eindeutigen Identifikator des Transportauftrags (Anhang A Rollenmodell A 5.2, A 6.1, A 7.1, A7.2)

   XTA Funktionalitäten:

   - Synchroner Versand (siehe Abschnitt 4.4.1.5 auf Seite 46)

4. Synchroner Versand

   Der Autor übergibt die Nachricht mit dem Transportauftrag für die Kommunikation mit dem Leser an den Sender (Anhang A Rollenmodell A 1.2, A 5.1).

   - Synchroner Versand (siehe Abschnitt 4.4.2.2 auf Seite 52)

5. Überprüfung der Kommunikation

   Der Autor überprüft, ob der Versand der Nachricht erfolgreich durchgeführt werden konnte (Anhang A Rollenmodell A 8.1), z. B. ob die verwendeten Zertifikate gültig waren. Bei positivem Ergebnis kann er die Rückantwort des Lesers verarbeiten. Im Falle eines Misserfolgs muss er ggf. Eskalationsmaßnahmen ergreifen.

   XTA Funktionalitäten:

   - Abruf eines Transportprotokolls (siehe Abschnitt 4.4.1.3 auf Seite 45)

4.3.1.3 Rückruf einer Nachricht

Wenn der Autor eine Nachricht für den asynchronen Versand an den Sender übermittelt hat, kann er den Versandauftrag zurückziehen. Dies ist nur möglich, wenn die Nachricht vom Sender noch nicht an den Empfänger übergeben wurde.

1. Rückruf eines Versandauftrags

   Der Autor übergibt dem Sender den eindeutigen Identifikator (MessageID) der Nachricht, die nach Erteilung eines asynchronen Versandauftrags doch nicht verschickt werden soll (Anhang A Rollenmodell A 8.2).

   - Rückruf einer Nachricht (siehe Abschnitt 4.4.1.4 auf Seite 46)

4.3.2 Aufgaben des Lesers

Die Aufgaben eines Lesers bestehen im Empfangen von Nachrichten und in der Überwachung des Nachrichtentransports (Anhang A Rollenmodell D1.1, D 5.1, D 5.2):

- Asynchroner Empfang einer Nachricht (siehe Abschnitt D.2.1 auf Seite 114)
- Asynchroner Empfang von Metadaten (siehe Abschnitt D.2.3 auf Seite 116)
- Synchroner Empfang einer Nachricht (siehe Abschnitt D.2.4 auf Seite 117)

Vor der ersten Übertragung sind technische und organisatorische Rahmenbedingungen zu schaffen (Anhang A Rollenmodell D10.1, D 9.2, D 9.1, D 8.3).

Bei jedem Empfang einer Nachricht hat der Leser eine Reihe von Aufgaben zu erfüllen:

- Der Leser muss seinen Empfänger authentifizieren (Anhang A Rollenmodell D8.2).

- Der Leser muss eingehende Nachrichten syntaktisch und semantisch prüfen (Anhang A Rollenmodell D1.2, D1.3, D2.1, D2.2, D2.3, D3.1, D3.2).
- Der Leser muss, wenn notwendig, Teile der Nachricht fachlich entschlüsseln (Anhang A Rollenmodell D4.1).
- Der Leser muss gemäß den geforderten Service Qualitäten reagieren (Anhang A Rollenmodell D6.1).
- Der Leser muss Informationen zum Nachrichtentransport bewerten (Anhang A Rollenmodell D7.1, D8.1).

4.3.2.1 Asynchroner Empfang von Nachrichten

Bei einem asynchronen Empfang nimmt der Empfänger die Nachrichten entgegen und hält diese für den Leser für eine Abholung bereit. Der Leser kann dann die Nachrichten zu einem von ihm bestimmten Zeitpunkt abholen.

Der Leser holt die Liste der MessageIDs der abzuholenden Nachrichten. Er entscheidet, ob und wann er die zugehörigen Nachrichten abholt.

1. Liste der Nachrichten ermitteln

   Zuerst holt der Leser die Liste der MessageIDs. Er kann Selektionskriterien angeben: Möchte er nur ungelesene Nachrichten? In welchem Zeitraum sollen diese Nachrichten eingegangen sein (Anhang A Rollenmodell D5.1)?

   XTA Funktionalitäten:

   a. Liste der MessageIDs und Metadaten holen (siehe Abschnitt 4.4.3.1 auf Seite 62)
2. Nachrichten an Hand einer MessageID abholen

   Der Leser verarbeitet die vom Empfänger abgerufenen MessageIDs. Hierfür erfolgen für jede MessageID drei Arbeitsschritte:
   - Der Leser kann sich zu jeder abgeholten Nachricht den Report vom Empfänger mit allen Transportinformationen holen. Hierfür verwendet er die MessageID der abgeholten Nachricht (Anhang A Rollenmodell D7.1). An Hand der Informationen aus dem Report bewertet der Leser, ob die Nachricht fachlich verarbeitet werden darf (Anhang A Rollenmodell D8.1).
   - Ist alles regelgemäß gelaufen, holt er die Nachricht für die aktuell zu verarbeitende MessageID.
   - Wenn er diese korrekt empfangen konnte, muss er noch den Empfang quittieren. Das kann er sofort tun, oder erst nach einer angemessenen Bearbeitung der Nachricht, z. B. wenn er sie lokal persistieren konnte (Anhang A Rollenmodell D5.1).

   XTA Funktionalitäten:
   - Abruf eines Transportprotokolls (siehe Abschnitt 4.4.1.3 auf Seite 45)
   - Abholen einer Nachricht für eine MessageID (siehe Abschnitt 4.4.3.2 auf Seite 63)
   - Quittieren der Abholung (siehe Abschnitt 4.4.3.3 auf Seite 65)

Hinweis: Der hier beschriebene Empfang von Nachrichten setzt voraus, dass nur ein Leser auf ein Postfach zugreift. Für den parallelen Zugriff durch mehrere Leser auf ein Postfach wird in Anhang C eine zweite Variante beschrieben, die in der aktuellen XTA-WS Version noch nicht unterstützt wird. Die Erfüllung der Anforderungen für eine XTA-Konformität ist nicht an die Umsetzung der im Anhang beschriebenen Variante gekoppelt.

4.3.2.2 Synchroner Empfang von Nachrichten

Bei einem synchronen Empfang leitet der Empfänger die eingehenden Informationen sofort an den Leser weiter. Dies ist nur möglich, wenn der Leser die vorgegebene Schnittstelle implementiert. Wird der Leser über die Schnittstelle vom Empfänger aufgerufen, dann kann er die Nachricht verarbeiten und das Ergebnis an den Empfänger zurückgeben. Der Empfänger leitet es an den Sender und der an den Autor weiter.

1. Empfangsbereitschaft des Leser

   Der Leser muss die WS-Methode sendMessageSync gemäß der Spezifikation implementieren (Anhang A Rollenmodell D5.2, D 8.3). Den Zugriff auf diese Methode muss der Leser dem Empfänger gewähren. Hierzu hat er diesem die notwendigen Informationen mitzuteilen, z. B. die URI.
2. Empfang von Nachrichten

   Der Leser wartet auf eingehende Nachrichten und reagiert hierauf unverzüglich (Anhang A Rollenmodell D5.2). Nach einer Prüfung der Korrektheit des Transports liefert er als Rückgabe an den Empfänger das Ergebnis seiner Verarbeitung.

   XTA-Funktionalität:

   Synchroner Versand einer Nachricht (siehe Abschnitt 4.4.2.2 auf Seite 52)

4.4 Methoden

Der XTA-WS wird durch die nachfolgend dokumentierten und spezifizierten Methoden beschrieben, die in drei Schnittstellentypen zusammengefasst sind (vgl. die Dateien xta.wsdl und xta-synchron.wsdl):

- Im Schnittstellentyp managementPort werden Methoden zusammengefasst, die Serviceleistungen im Umgang mit Nachrichten anbieten.
- Der Schnittstellentyp sendPort bündelt Methoden, die direkt die Erteilung eines Transportauftrages betreffen.
- Im Schnittstellentyp msgBoxPort sind Methoden zusammengefasst, die „Postkasten-Funktionen“ wahrnehmen: es handelt sich hier um Aufgaben, die beim Entgegennehmen und Abholen einer oder mehrerer Nachricht erledigt werden müssen.

Die Beispiele, die der Dokumentation der Methoden beigefügt sind, nutzen folgende Werte:

- Das Meldewesen benutzt als Prefix das Kürzel „ags“.
- Der Autor ist die Meldebehörde Stadt Testhausen mit dem amtlichen Gemeindeschlüssel 87654321.
- Der Leser ist die Meldebehörde Dorf Testdorf mit dem amtlichen Gemeindeschlüssel 76859403.
- Der fachliche Dienst hat die Bezeichnung

  http://www.osci.de/xmeld18/xmeld18Fortschreibung.wsdl.

4.4.1 Schnittstellentyp managementPort

In diesem Schnittstellentyp sind alle Management-Methoden des XTA-WS zusammengefasst, die als Service-Leistungen zur Verfügung gestellt werden. Dies sind:

- checkAccountActive
- lookupService
- createMessageId
- cancelMessage
- getTransportReport

4.4.1.1 Methode checkAccountActive (Verbindung Autor –> Sender bzw. Verbindung Leser -> Empfänger)

Mit der Methode checkAccountActive kann der Autor prüfen, ob seine Verbindung zum Sender funktioniert. Der Autor fragt beim Sender an. Damit der Sender „weiß“, wer ihn fragt, muss der Autor eine

Angabe über seine Identität mitgeben und diese nachweisen. Die Angabe der Identität erfolgt nach den Vorgaben des jeweiligen fachlichen Kontextes.

Die Methode checkAccoutActive prüft also, ob der Webservice verfügbar ist und ein Account beim XTA-Betreiber eingerichtet ist.

Typischerweise erfolgt der Aufruf nach einer Änderung der Konfiguration oder für den Fall, dass technische Probleme auftreten.

4.4.1.1.1 Ergebnisse

- Kehrt die Methode ohne Fehler zurück, ist der XTA-WS erreichbar und der Account aktiv.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.

4.4.1.1.2 Operation checkAccountActive

Input.

Soap Part	Name	Type
Header	AuthorIdentifier (fachliche Identität des Autors)	osci21:PartyType

Wesentliche Parameter:

- „osci21: Author“: die fachliche Identität des Autors

Output. Keine Rückgabewerte.

4.4.1.1.3 Beispielcode (Aufruf der Methode)

```
checkAccountActive(osci21:Author)
```

4.4.1.2 Methode lookupService (Erreichbarkeit Autor – Leser)

Der Autor nutzt die Methode lookupService um festzustellen, ob ein Leser prinzipiell für einen Dienst elektronisch erreichbar ist: Über die Methode werden Verzeichnisdienste wie z.B. das DVDV angefragt, um die Informationen abzurufen.

In der Antwort erhält der Autor Daten, die er z. B. für eine Verschlüsselung der Nachricht benötigt: Dies kann das Inhaltsdatenverschlüsselungszertifikat des Lesers für die Ende-zu-Ende-Verschlüsselung sein.

Um die Prüfung durchführen zu können, benötigt der Autor die fachliche Identität des Lesers und die Bezeichnung des fachlichen Dienstes.

Die Methode kann zur fachlichen Steuerung der Prozesse im Fachverfahren verwendet werden.

4.4.1.2.1 Ergebnisse

- Folgende Ergebnisse sind möglich:
  - <ServiceIsAvailable> = "true" : Der Leser bietet den Dienst an

  - <ServiceIsAvailable > = "false": Der Leser bietet den Dienst nicht an
  - <ServiceIsAvailableUnknown > = "true": Der benötigte Verzeichnisdienst ist nicht erreichbar.
- Wenn vorhanden, wird das Inhaltsdatenverschlüsselungszertifikat des Lesers geliefert.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist,
- Der technische Fehler (SoapFault) <ParameterIsNotValidException> wird zurückgegeben, wenn ein Pflicht-Übergabeparameter fehlt oder ein Übergabeparameter fehlerhaft ist. Der Übergabeparameter ist fehlerhaft,
  - wenn der Parameter <ServiceType> des LookupServiceType keine gültige Dienstebezeichnung (z.B. aus dem DVDV) repräsentiert oder
  - wenn der Parameter <ReaderIdentifier> des LookupServiceType keinen gültigen Wert enthält.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist. Insbesondere entsteht er auch dann, wenn der Verzeichnisdienst für den Sender nicht erreichbar bzw. innerhalb eines vom Sender festgelegten Timeouts nicht erreichbar ist.

4.4.1.2.2 Operation lookupService

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	LookupServiceRequest	xta:LookupServiceRequest

Wesentliche Parameter:

- „osci21:Author“: Angabe der fachlichen Identität des Autors
- „xta:lookupServiceRequest“: Liste der zu prüfenden Empfänger
  - „osci21:Reader“: die fachliche Identität des Empfängers z.B. osci21:Identifier = „ags:76859403”
  - „xta:ServiceType“: Dienstebezeichnungen des fachlichen Dienstes, z.B. „http://www.osci.de/xmeld18/xmeld18Fortschreibung.wsdl"

Output.

Soap Part	Name	Type
Body	LookupServiceResponse	xta:LookupServiceResponse

Rückgabewerte:

- „xta:LookupServiceResponse“: xta:LookupServiceRequest mit den Ergebnissen:
  - „osci21:Reader“: die fachliche Identität des Empfängers
  - „xta:ServiceType“: Dienstebezeichnungen des fachlichen Dienstes
  - „xta:IsServiceAvailableValue“: Ergebnis der Erreichbarkeitsprüfung
  - „xta:Property“: Rückgabe der vom Autor im Fachszenario benötigten optionalen Informationen über den Leser, z. B. dessen öffentlicher Verschlüsselungsschlüssel.

4.4.1.2.3 Beispielcode (Aufruf der Methode)

```
lookupService(osci21:Author, LookupServiceRequest)
```

4.4.1.3 Methode getTransportReport (Abruf eines Transportprotokolls)

Autor und Leser haben die Verantwortung für den korrekten Nachrichtentransport. Für die notwendige Überwachung stellen Sender und Empfänger eine Funktion zum Abruf des Transportprotokolls zur Verfügung. Dieses Protokoll (<TransportReport>), enthält Angaben zum konkreten Transportauftrag und zu den Ereignissen, die während des Transports protokolliert worden sind.

Mit der Methode getTransportReport können also der Autor vom Sender - und der Leser vom Empfänger - das Transportprotokoll abholen. Es kann also erfragt werden, ob zu einer bestimmten Nachricht ein Transportprotokoll vorliegt, z.B., um zu ermitteln, mit welchem Ergebnis die Zertifikatsprüfungen durchgeführt wurden und wie die Nachricht weitergeleitet wurde.

4.4.1.3.1 Ergebnisse

- Es wird das Transportprotokoll in einem <TransportReport> Objekt zurückgegeben.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist,
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist,
- Der technische Fehler (SoapFault) <InvalidMessageIdException> entsteht, wenn zu der übergebenen MessageID kein Transportprotokoll für den Account bekannt ist.

4.4.1.3.2 Operation getTransportReport

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MessageID	wsa:MessageID

Wesentliche Parameter:

- "osci21:Author": die fachliche Identität des Autoren.
- „wsa:MessageID“: eindeutiger Identifikator des Transportauftrags, z.B. "DE_NDS_KDO:743ab5e0-8277-11e2-9e96-0800200c9a66"

Output.

Soap Part	Name	Type
Body	GetTransportReportResponse	xta:TransportReport

Rückgabewerte:

- „xta:TransportReport“: Es wird der Report mit den Informationen über den Transport geliefert.

4.4.1.3.3 Beispielcode (Aufruf der Methode)

```
getTransportReport(osci21:Author, wsa:MessageID)
```

4.4.1.4 Methode cancelMessage (Rückruf einer Nachricht)

Mit der Methode cancelMessage kann der Autor einen vom Sender noch nicht bearbeiteten Versandauftrag zurückziehen / zurückrufen. Dies ist möglich, wenn folgende Bedingungen erfüllt sind:

- Die zugehörige Nachricht muss dem Autor gehören.
- Die Nachricht muss zuvor vom Autor zum asynchronen Versand übergeben und darf noch nicht verschickt worden sein.
- Die entsprechende Nachricht wird über ihre MessageID eindeutig bezeichnet.

Der Aufruf ist erfolgreich, wenn kein Fehler (Exception) zurückgegeben wird.

4.4.1.4.1 Ergebnisse

- Kehrt die Methode ohne Fehler zurück, so ist der XTA-WS erreichbar und der Transportauftrag ist zurückgezogen.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <ParameterIsNotValidException> wird zurückgegeben, wenn ein Pflicht-Übergabeparameter fehlt oder ein Übergabeparameter fehlerhaft ist.
- Der technische Fehler (SoapFault) <InvalidMessageIdException> entsteht, wenn die MessageID, also der angeforderte Transportauftrag, dem Account nicht bekannt ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.

4.4.1.4.2 Operation cancelMessage

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MessageID (Angabe der ID der zurückzuziehenden Nachricht)	wsa: MessageID

Wesentliche Parameter:

- „osci21: Author“: die fachliche Identität des Autors
- „wsa:MessageID“: Übergabe der ID der zurückzuziehenden Nachricht, z.B. “DE_NDS_KDO: 743ab5e0-8277-11e2-9e96-0800200c9a66

Output. Keine Rückgabewerte.

4.4.1.4.3 Beispielcode (Aufruf der Methode)

```
cancelMessage(osci21:Author, wsa:MessageID)
```

4.4.1.5 Methode createMessageId (Erzeugung eines eindeutigen Identifikators)

Jeder Transportauftrag benötigt einen (räumlich und zeitlich) eindeutigen Identifikator. Diese MessageID kann über die gesamte Transportkette hinweg zur eindeutigen Identifikation des Transportauftrages oder zur Abfrage von Protokollinformationen verwendet werden.

Die MessageID besteht aus zwei Teilen: einen den Aussteller eindeutig identifizierenden Prefix und eine UUID. Beispiel: „DE_NDS_KDO: 743ab5e0-8277-11e2-9e96-0800200c9a66“.

Mit der Methode createMessageId kann der Autor den Sender veranlassen, eine solche MessageID zu generieren und liefern zu lassen. Der Aufruf der Methode gehört zur Vorbereitung eines Transportauftrages, der über diese MessageID identifiziert werden soll.

Die Methode createMessageId wird nicht benötigt, wenn das Fachverfahren selbst eine eindeutige Identifikation erzeugt, die den genannten Anforderungen genügt.

4.4.1.5.1 Ergebnisse

- Es wird eine neu erzeugte MessageID zurückgeliefert.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.

4.4.1.5.2 Operation createMessageID

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType

Wesentliche Parameter:

- „osci21: Author“: die fachliche Identität des Autors

Output.

Soap Part	Name	Type
Body	MessageID	wsa:MessageID

Rückgabewerte:

- „wsa:MessageID“: Die vom Sender berechnete eindeutige MessageID für den Transportauftrag.

4.4.1.5.3 Beispielcode (Aufruf der Methode)

```
createMessageId(osci21:Author)
```

4.4.1.6 Wichtige Objekte der managementPort-Schnittstelle

4.4.1.6.1 PartyIdentifierType

Die (generische) Adressierung erfolgt über die Kommunikationsendpunkte. Diese werden durch einen Typ „ **PartyIdentifierType** “ modelliert: **PartyIdentifierType** ist die Typdefinition für die Instanzen der Source- und Target-Endpunkte **Originators** (Autor, auch Sender) und **Destinations** (Empfänger).

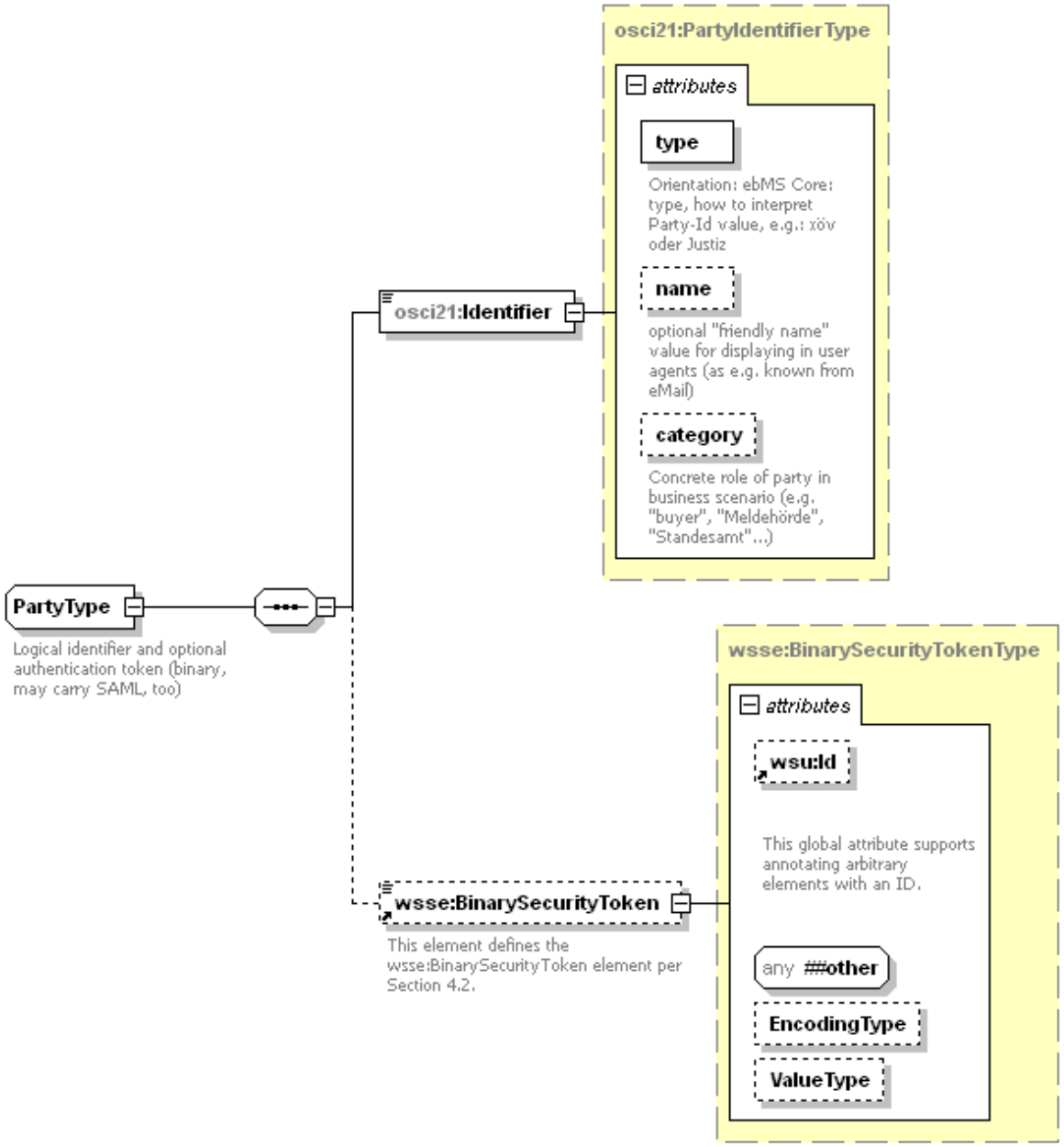
Der Identifier selbst ist vom Typ **xs:normalizedString** , attributiert durch @type , vom Type **xs:QName**, der den Typ dieses Identifiers auszeichnet – also z.B. einen Identifier aus dem DVDV oder aus S.A.F.E. Aus dem jeweils zugeordneten Verzeichnisdienst können die Verbindungsparameter entnommen werden, wie sie OSCI für WS-Addressing benötigt.

Ein PartyIdentifierType hat zusätzliche optionale informatorische Attribute:

```
<xs:complexType name="PartyIdentifierType">
 <xs:annotation>
  <xs:documentation>Value of generic party identifier, as classified by
 @type attribute,
   e.g.: Prefix:Kennung
  </xs:documentation>
 </xs:annotation>
 <xs:simpleContent>
  <xs:extension base="xs:normalizedString">
   <xs:attribute name="type" type="xs:QName" use="required">
    <xs:annotation>
     <xs:documentation>Orientation: ebMS Core: type, how to interpret
 Party-Id value, e.g.: xöv oder Justiz</xs:documentation>
    </xs:annotation>
   </xs:attribute>
   <xs:attribute name="name" type="osci21:NonEmptyStringType">
    <xs:annotation>
     <xs:documentation>optional "friendly name" value for displaying in
 user agents (as e.g. known from eMail)</xs:documentation>
    </xs:annotation>
   </xs:attribute>
   <xs:attribute name="category" type="xs:QName">
    <xs:annotation>
     <xs:documentation>Concrete role of party in business scenario
      (e.g. "buyer", "Meldehörde", "Standesamt"...)
     </xs:documentation>
    </xs:annotation>
   </xs:attribute>
  </xs:extension>
 </xs:simpleContent>
</xs:complexType>
```

Für die Auflösung der generischen Adressen werden in OSCI 2.01 die WS-Addressing Parameter beim Sender vom XTA WebService gesetzt.

Die Authentisierungsinformationen (X509- oder SAML-Token) werden zu den Kommunikationsendpunkt durch den Typ **PartyType** übermittelt, der einen PartyIdentifierType um ein optionales Element **BinarySecurityToken** gemäß WS-Security erweitert:



Die dargestellten komplexen Typen zur Aufnahme generischer Adressierungsinformationen werden in den Elementen instanziiert, die in dem Header **MessageMetaData** enthalten sind.

4.4.1.6.2 MessageID

Eine MessageID stellt eine eindeutige Identifizierung eines bestimmten Transportauftrages dar (Transportauftragsnummer). Damit entspricht sie nicht anderen MessageID's, wie z. B. der MessageID aus OSCI-Transport 1.2 oder der ID des XÖV-Dokuments.

Die MessageID ist eine zeitlich und räumlich unbegrenzt gültige eindeutige Identifikationsnummer (z.B. entsprechend WS-Adressing). Sie gewährleistet im XTA-WS-Kontext die eindeutige Identifikation einer Nachricht und wird normalerweise vom Fachverfahren (Autor/Leser) an das Transportverfahren (Sender/Empfänger) übergeben. Für jeden Transportauftrag wird eine neue MessageID erzeugt, auch wenn

dieselbe fachliche Nachricht übertragen werden soll. Muss z. B. eine XÖV Nachricht auf Grund von aufgetretenen Fehlern erneut verschickt werden, dann bleibt die ID der XÖV Nachricht gleich, während die MessageID des Transportauftrags neu berechnet wird.

Verwendung findet die MessageID bei der Protokollierung und den Statusabfragen. Sie ist aber auch bei der Kommunikation im Fehlerfall sehr wichtig, weil sie die Identifizierung der Nachricht über die gesamte Transportstrecke hinweg erleichtert. Das Transportverfahren kann die benötigten MessageID's zur Verfügung stellen, Fachverfahren können sie auch selber berechnen.

Diese MessageID gilt sowohl für die Nachricht, wie auch für das dazugehörige Transportprotokoll. Die MessageID hat das Format einer URN. Sie sollte folgende Anforderungen erfüllen:

- Die MessageID muss beim Sender eindeutig sein. Sonst kann das Fachverfahren nicht den zugehörigen TransportReport abholen.
- Die MessageID muss beim Empfänger eindeutig sein. Sonst kann das Fachverfahren nicht den zugehörigen TransportReport abholen.
- Die MessageID soll erkennen lassen, wer die MessageID erstellt hat. So kann in Problemfällen der Ersteller der Nachricht leichter ermittelt werden.
- Die MessageID soll beim Empfänger dieselbe sein wie beim Sender. Nur so kann die Nachricht über den gesamten Transwortweg eindeutig identifiziert werden.

Für die Erfüllung der ersten beiden Anforderungen reicht es aus, eine UUID als Identifikator zu verwenden. Wegen der dritten Anforderung wird ein Präfix hinzugefügt, der eindeutig die Softwareinstanz identifiziert, die die MessageID erstellt hat. (Das Verfahren zur Erstellung dieses Präfixes ist noch zu klären.) Somit ergibt sich der folgende Aufbau der MessageID:

- Präfix: Angabe über die Softwareinstanz, die die MessageID erstellt, z.B. ClearingstelleXY_Xta_01 oder ClearingstelleXY_SAP_15.
- Identifikator: Dieser muss aus einer UUID generiert sein (siehe RFC4122, z.B. 000ca2fe-f4e1-45c2-8233-3a0eb760bd16)

Allgemeiner Aufbau als URN: urn:xoevmessageid:<Präfix>:<Identifikator>

Beispiel: urn: xoevmessageid:Dataport_Xta_01:000ca2fe-f4e1-45c2-8233-3a0eb760bd16

4.4.2 Schnittstellentyp sendPort

In diesem Schnittstellentyp sind die Methoden des XTA-WS zusammengefasst, die für die Erteilung eines Transportauftrags angeboten werden. Dies sind:

- sendMessage
- sendMessageSync

4.4.2.1 Methode sendMessage (Asynchroner Versand einer Nachricht)

Für den asynchronen Versand erteilt der Autor dem Sender einen Versandauftrag. Dabei muss der Autor, der für den Transport verantwortlich ist, die folgenden Informationen mitgeben:

- die eigentliche Nachricht,
- die Beschreibung des Transportauftrags (Metadaten) mit der MessageID,
- die Liste der zu prüfenden Zertifikate.

Diese Daten werden in einem Aufruf an den XTA-WS übergeben. Mit diesem Aufruf ist die Erteilung des Transportauftrags erfolgt.

Die Mitteilungspflicht des Fachverfahrens ist mit Aufruf dieser Methode durch den Webservice-Client erfüllt, wenn kein technischer Fehler (Exception) ausgelöst wurde. Dies enthebt den Autor allerdings nicht von der Pflicht, die Zustellung zu überwachen, also z.B. Transportprotokolle auszuwerten.

4.4.2.1.1 Ergebnisse

- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <ParameterIsNotValidException> entsteht, wenn ein Pflichtübergabeparameter fehlt oder ein Übergabeparameter fehlerhaft ist. Dies tritt in folgenden Fällen auf:
  - Der Parameter <MessageID> innerhalb des XTA ist nicht eindeutig.
  - Der Parameter <ServiceType> repräsentiert keine gültige Dienstebezeichnung oder der Dienst wird vom Empfänger nicht angeboten.
  - Der Parameter <AuthorIdentifier> ist nicht gemäß der jeweiligen fachlichen Spezifikation gefüllt.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn
  - ein technischer Fehler im XTA-WS aufgetreten ist,
  - der Empfänger nicht erreichbar ist,
  - der Empfänger nicht innerhalb eines vom Sender festgelegten Time-Outs antwortet.
- Der technische Fehler (SoapFault) <MessageSchemaViolationException> entsteht, wenn die zum Versand übergebene Nachricht nicht konform zur jeweiligen XML Schema-Definition ist. Insbesondere entsteht der Fehler dann, wenn in der übergebenen Nachricht ein fehlerhaftes Encoding eingestellt ist oder wenn das standardspezifische Wesensprofil verletzt ist.
- Der technische Fehler (SoapFault) <MessageVirusDetectionException> entsteht, wenn in der zum Versand übergebenen Nachricht schadhafter Code ermittelt wurde.
- Der technische Fehler (SoapFault) <SyncAsyncException> entsteht, wenn eine Nachricht übergeben wurde, die nur für den synchronen Versand gültig ist.

4.4.2.1.2 Operation sendMessage

Input.

Soap Part	Name	Type
Header	MessageMetaData	osci21:MessageMetaData
Header	X509TokenContainer	osci:X509TokenContainer
Body	GenericContentContainer	xta:GenericContentContainer

Wesentliche Parameter:

- „osci21:MessageMetaData“: In dieser Struktur werden die Metadaten des Transportauftrags definiert. Da diese Daten als optionaler Soap-Header mitgeführt werden, muss für einen Zugriff nur dieser Header, aber nicht eine eingebettete Nachricht gelesen werden. Die Metadaten beinhalten Zeitstempel, Quittungsanforderungen, Wesensprofil, Angaben über den Autoren und den Leser, Informationen zur Identifikation der Nachricht und weitere Informationen.
- „osci:X509TokenContainer“: In diesem optionalen Soap-Header können zu prüfende Zertifikate eingestellt werden. (Die Prüfung der Zertifikate kann auf dem Transportweg durchgeführt werden und ist eine optionale Serviceleistung der Transportinfrastruktur.) Beispiel für ein eingestelltes Zertifikat ist ein Signaturzertifikat für eine fachliche Signatur der zu übertragenden Nachricht im Kontext XhD.
- „xta:GenericContentContainer“: Dieses Objekt beinhaltet die zu übertragende Nachricht und eine beliebige Anzahl von Anhängen (Attachments). Die Nachricht selber kann in einem verschlüsselten Container hinterlegt werden. Zu der Nachricht kann ein Betreff (Subject) angegeben werden.

Output. Keine Rückgabewerte.

4.4.2.1.3 Beispielcode (Aufruf der Methode)

```
sendMessage(osci21:MessageMetaData, osci21:X509TokenContainer,
 xta:GenericContentContainer)
```

4.4.2.2 Methode sendMessageSync - Sender (Synchroner Versand einer Nachricht)

Mit der Methode sendMessageSync kann der Autor über den Sender mit einem Leser kommunizieren: Der Autor schickt synchron eine Nachricht und bekommt (synchron) direkt eine Nachricht als Ergebnis zurück.

Bei einem synchronen Versand erteilt der Autor also dem Leser einen Versandauftrag. Dabei muss der Autor alle notwendigen Informationen mitgeben, denn er ist für den Transport verantwortlich. Folgende Informationsblöcke müssen mitgegeben werden:

- die eigentliche Nachricht,
- die Beschreibung des Transportauftrags (Metadaten) mit der MessageID
- die Liste der zu prüfenden Zertifikate.

Diese Informationen werden in einem Aufruf an den XTA-WS übergeben. Der Autor wartet, bis der Sender die Antwort des Lesers an ihn übergibt.

Mit der Methode sendMessageSync wird also eine Nachricht an den XTA-WS für einen synchronen Transport übergeben. Durch den Aufruf der Methode ist der Auftrag zum Transport erteilt.

Die Methode sendMessageSync ist fast identisch zur Methode sendMessage. Wesentlicher Unterschied: Als Rückgabewert (< GenericContentContainer>) wird die Antwortnachricht des adressierten Lesers entsprechend der jeweiligen Fachspezifikation geliefert.

4.4.2.2.1 Ergebnisse

- Im Erfolgsfall wird als Rückgabewert das Element <GenericContainerBody> zurückgegeben, das die Antwortnachricht des Kommunikationspartners enthält.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <ParameterIsNotValidException> entsteht, wenn ein Pflichtübergabeparameter fehlt oder ein Übergabeparameter fehlerhaft ist. Dies tritt in folgenden Fällen auf:
  - Der Parameter <MessageID> ist innerhalb des XTA nicht eindeutig.
  - Der Parameter <ServiceType> repräsentiert keine gültige Dienstebezeichnung oder der Dienst wird vom Empfänger nicht angeboten.
  - Der Parameter <AuthorIdentifier> ist nicht gemäß der jeweiligen fachlichen Spezifikation gefüllt.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn
  - ein technischer Fehler im XTA-WS aufgetreten ist,
  - der Empfänger nicht erreichbar ist,
  - der Empfänger nicht innerhalb eines vom Sender festgelegten Time-Outs antwortet.
- Der technische Fehler (SoapFault) <MessageSchemaViolationException> entsteht, wenn die zum Versand übergebene Nachricht nicht konform zur jeweiligen XML Schema-Definition ist. Insbesondere entsteht der Fehler dann, wenn in der übergebenen Nachricht ein fehlerhaftes Encoding eingestellt ist oder wenn das standardspezifische Wesensprofil verletzt ist.

- Der technische Fehler (SoapFault) <MessageVirusDetectionException> entsteht, wenn in der zum Versand übergebenen Nachricht schadhafter Code ermittelt wurde.
- Der technische Fehler (SoapFault) <SyncAsyncException> entsteht, wenn eine Nachricht übergeben wurde, die nur für den asynchronen Versand gültig ist.

4.4.2.2.2 Operation sendMessageSync

Input.

Soap Part	Name	Type
Header	MessageMetaData	osci21:MessageMetaData
Header	X509TokenContainer	osci:X509TokenContainer
Body	GenericContentContainer	xta: GenericContentContainer

Wesentliche Parameter:

- „osci21:MessageMetaData“: In dieser Struktur werden die Metadaten des Transportauftrags definiert. Da diese Daten als optionaler Soap-Header mitgeführt werden, muss für einen Zugriff nur dieser Header und nicht die ggf. eingebettete Nachricht gelesen werden. Die Metadaten beinhalten Zeitstempel, Quittungsanforderungen, das Wesensprofil, Angaben über den Autoren und den Leser, Informationen zur Identifikation der Nachricht und weitere Informationen.
- „osci:X509TokenContentCointainer“: In diesem optionalen Soap-Header können zu prüfende Zertifikate eingestellt werden. Die Prüfung kann auf dem Transportweg durchgeführt werden und ist eine optionale Serviceleistung der Transportinfrastruktur.
- „xta:GenericContentContainer“: Dieses Objekt beinhaltet die zu übertragende Nachricht und eine beliebige Anzahl von Anhängen (Attachments). Die Nachricht selber kann in einem verschlüsselten Container hinterlegt werden. Zu der Nachricht kann ein Betreff (Subject) angegeben werden.

Output.

Soap Part	Name	Type
Header	MessageMetaData	osci21:MessageMetaData
Header	X509TokenContainer	osci:X509TokenContainer
Body	GenericContentContainer	xta: GenericContentContainer

Rückgabewerte:

- „osci:X509TokenContentCointainer“: In diesem optionalen Soap-Header kann der Leser zu prüfende Zertifikate eingestellen. Die Prüfung kann auf dem Transportweg durchgeführt werden und ist eine optionale Serviceleistung der Transportinfrastruktur.
- "osci21:MessageMetaData“: In dieser Struktur werden die Metadaten des Transportauftrags zu der Antwort des Lesers definiert. Da diese Daten als optionaler Soap-Header mitgeführt werden, muss für einen Zugriff nur dieser Header und nicht ev. die eingebettete Nachricht gelesen werden. Die Metadaten beinhalten Zeitstempel, Quittungsanforderungen, das Wesensprofil, Angaben über den Autoren und den Leser, Informationen zur Identifikation der Nachricht und weitere Informationen.
- „xta:GenericContentContainer“: Dieses Objekt beinhaltet die Antwort des Leser. Sie besteht aus einer Nachricht und einer beliebigen Anzahl von Anhängen (Attachments). Die Nachricht selber kann in einem verschlüsselten Container hinterlegt werden. Zu der Nachricht kann ein Betreff (Subject) angegeben werden.

4.4.2.2.3 Beispielcode (Aufruf der Methode)

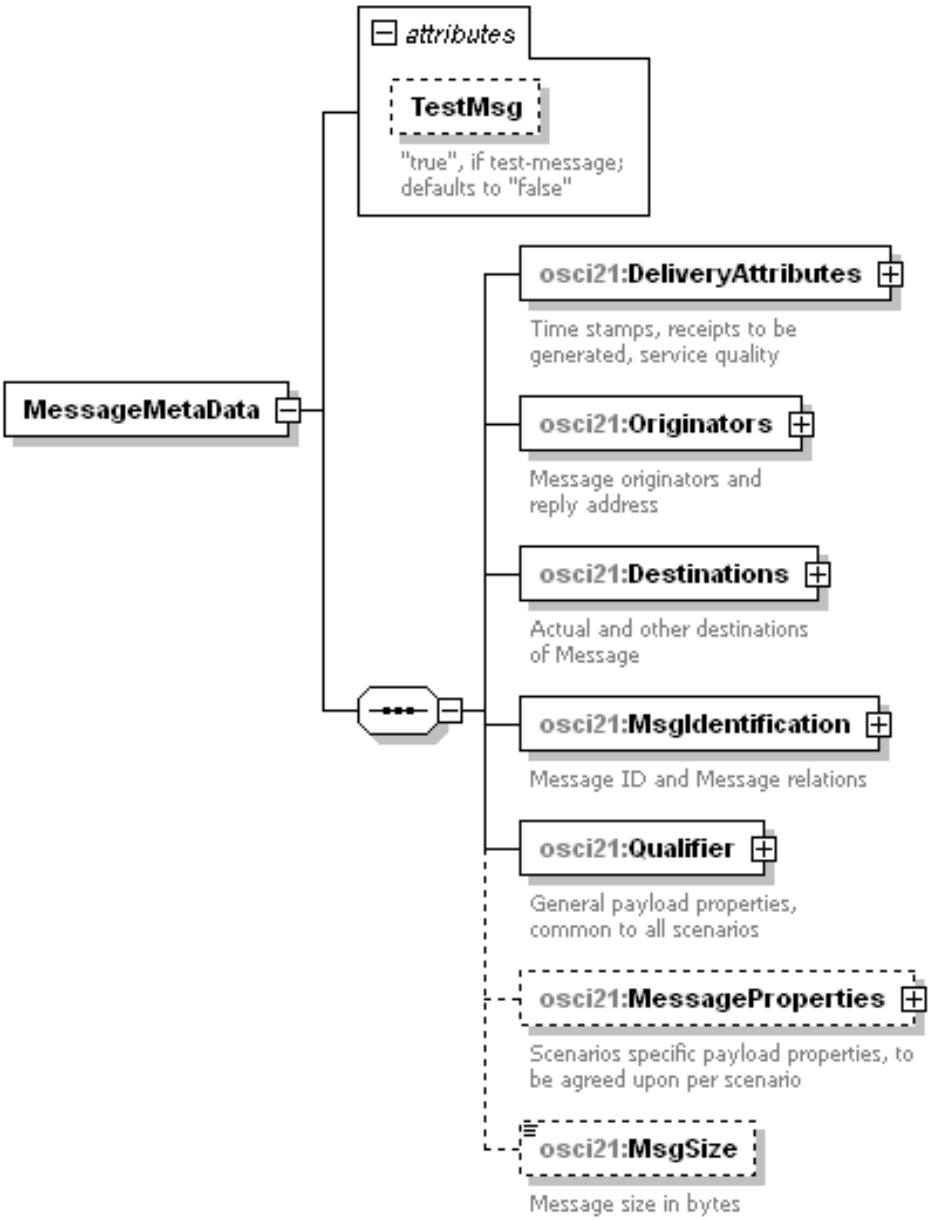
```
sendMessageSync(ref osci21:MessageMetaData, ref osci21:X509TokenContainer,
 ref xta:GenericContentContainer)
```

4.4.2.3 Wichtige Objekte der sendPort-Schnittstelle

4.4.2.3.1 Daten des Transportauftrags

4.4.2.3.1.1 MessageMetaData

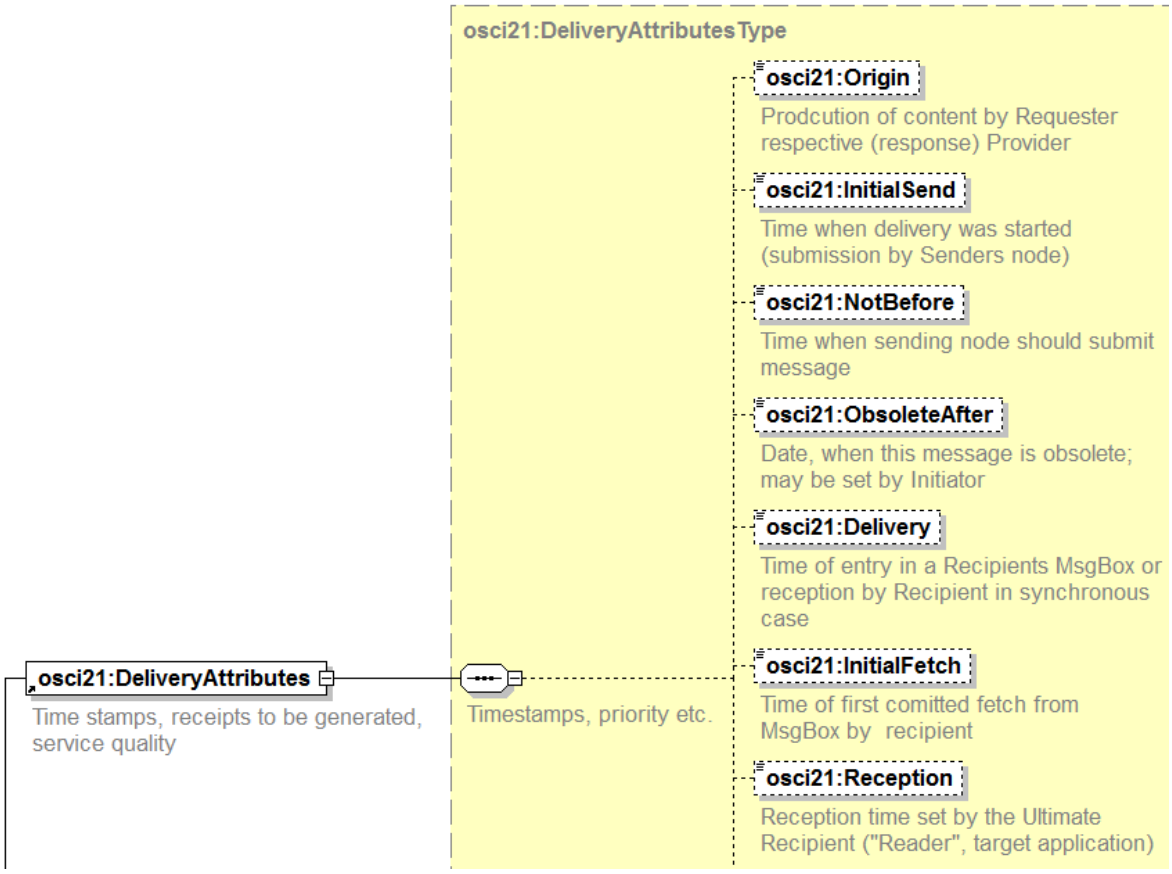
Namensraum des Headers: **"http://www.osci.eu/ws/2013/02/transport"**.



Der Header-Block **MessageMetaData** nimmt alle Transportdaten und Metainformationen zum Payload auf. Er ist attributiert durch **xs:boolean @TestMsg**, um bei Bedarf Testnachrichten auszuzeichnen.

MessageMetaData hat generelle Bestandteile, die für alle unterstützten Szenarien relevant sind. Im optionalen Element **MessageProperties** können Szenarien-spezifische Meta-Informationen zum Payload hinterlegt werden, die pro Szenario profiliert werden müssen („Property"-Benennung, -Werte, -Semantik). Das Element **MsgSize** dient der Optimierung bei Empfangsknoten (Streaming) und wird vom sendenden OSCI-Gateway mit der Größe der Nachricht in Bytes gesetzt.

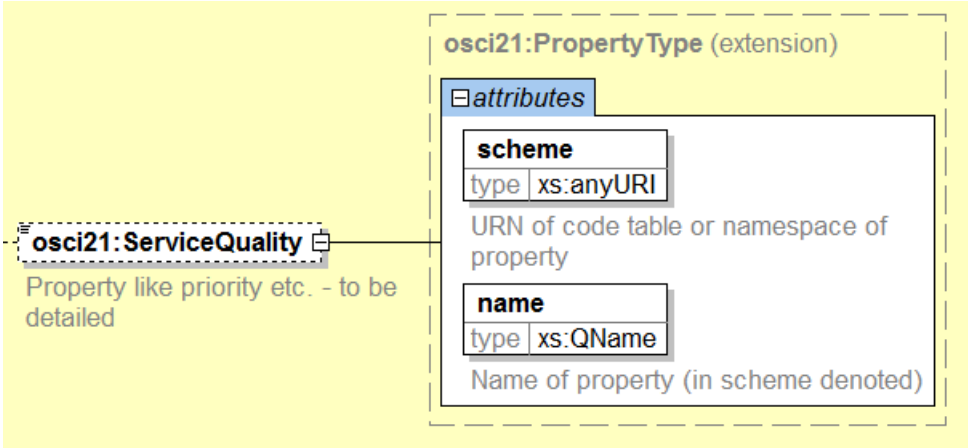
4.4.2.3.1.2 DeliveryAttributes



- Zeitstempel, die von entsprechenden Knoten während des Transports appliziert werden (siehe auch **MsgTimeStamps** aus OSCI 2, vgl. Abschnitt F.1 auf Seite 121)
  - **Origin, ObsoleteAfter** können vom Autor gesetzt werden
  - **InitialSend** muss vom Sender gesetzt werden
  - **NotBefore** kann vom Autor gesetzt werden. Der Sender kann die Nachricht erst zu diesem Zeitpunkt versenden.
  - **Delivery** muss von MsgBox (im asynchronen Szenario) bzw. Empfänger (im synchronen Szenario) gesetzt werden.
  - **InitialFetch** muss von MsgBox beim initialen Abholen einer Nachricht gesetzt werden.
  - **Reception** muss vom Leser gesetzt (bzw. durch diesen Knoten getriggert) werden.

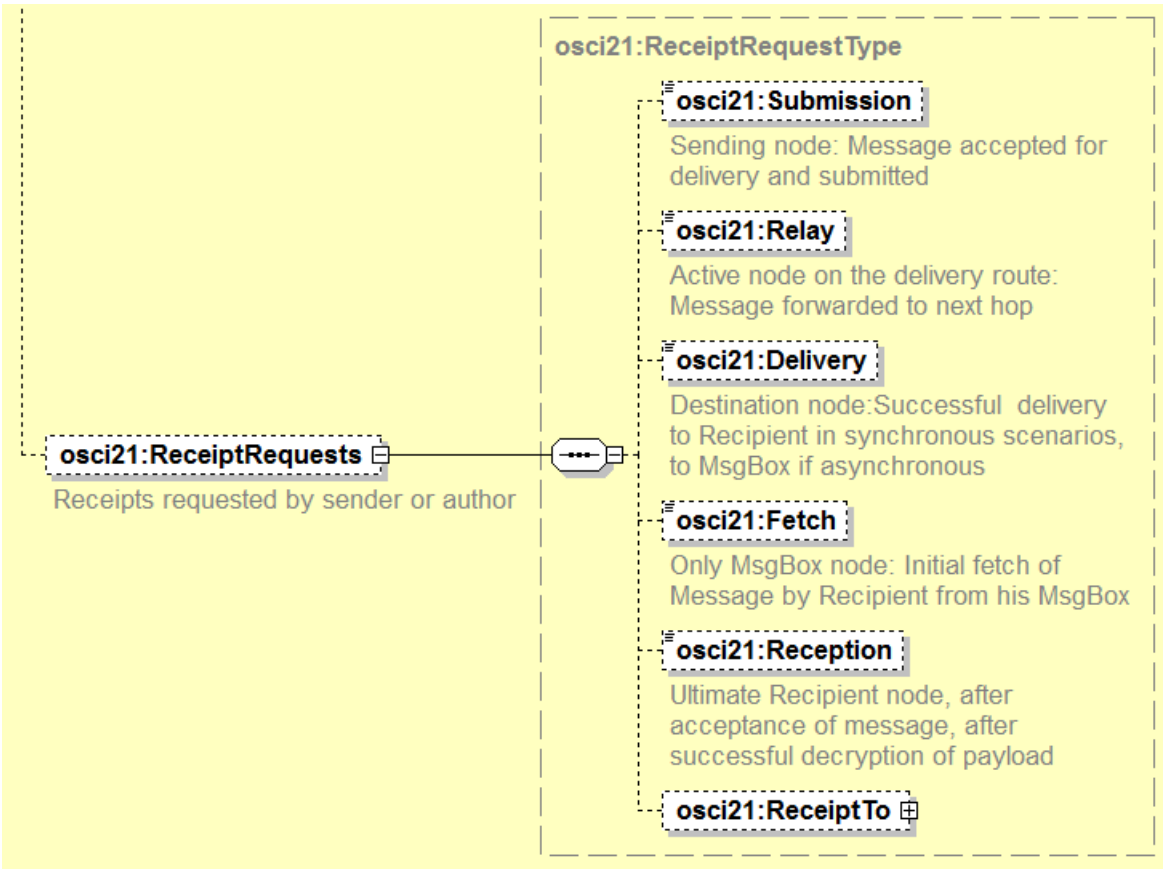
Der in OSCI2.01 definierte Header **MsgTimeStamps** wird aus Kompatibilitätsgründen weiter in OSCI-Nachrichten mitgeführt. Die Einträge in **MsgTimeStamps** müssen von OSCI2.01-Gateways bzw. OSCI2.01-MsgBox-Services entsprechend der oben aufgeführten Elemente in **MessageMetaData** transparent gesetzt werden. Der damit redundante Header **MsgTimeStamps** kann in einer zukünftigen Version von OSCI Transport entfallen.

- **ServiceQuality** : Hier wird die Referenz auf das Wesenspofil gesetzt, das aus den zu verwendenden Schutz- und Infrastrukturprofilen resultiert (vgl. Abschnitt 3.1 Übersicht der Profilarten). Der Typ „**PropertyType** “ ist eine Extension von **xs:anySimpleType** mit qualifizierende Attributen **@name** und dem Namensraum **(@scheme**) dieser Property. Die Parameter für xta:ServiceQuality werden mit Bezug auf diesen Namensraum festgelegt und referenzierbar gemacht. (Die Codeliste der referenzierbaren Profile und die Profile selbst werden im XRepository abgelegt.)




```
<xs:complexType name="PropertyType">
 <xs:simpleContent>
  <xs:extension base="xs:anySimpleType">
   <xs:attribute name="scheme" type="xs:anyURI" use="required">
    <xs:annotation>
     <xs:documentation>URN of code table or namespace of property </
xs:documentation>
    </xs:annotation>
   </xs:attribute>
   <xs:attribute name="name" type="xs:QName" use="required">
    <xs:annotation>
     <xs:documentation>Name of property (in scheme denot-ed)</
xs:documentation>
    </xs:annotation>
   </xs:attribute>
  </xs:extension>
 </xs:simpleContent>
</xs:complexType>
```

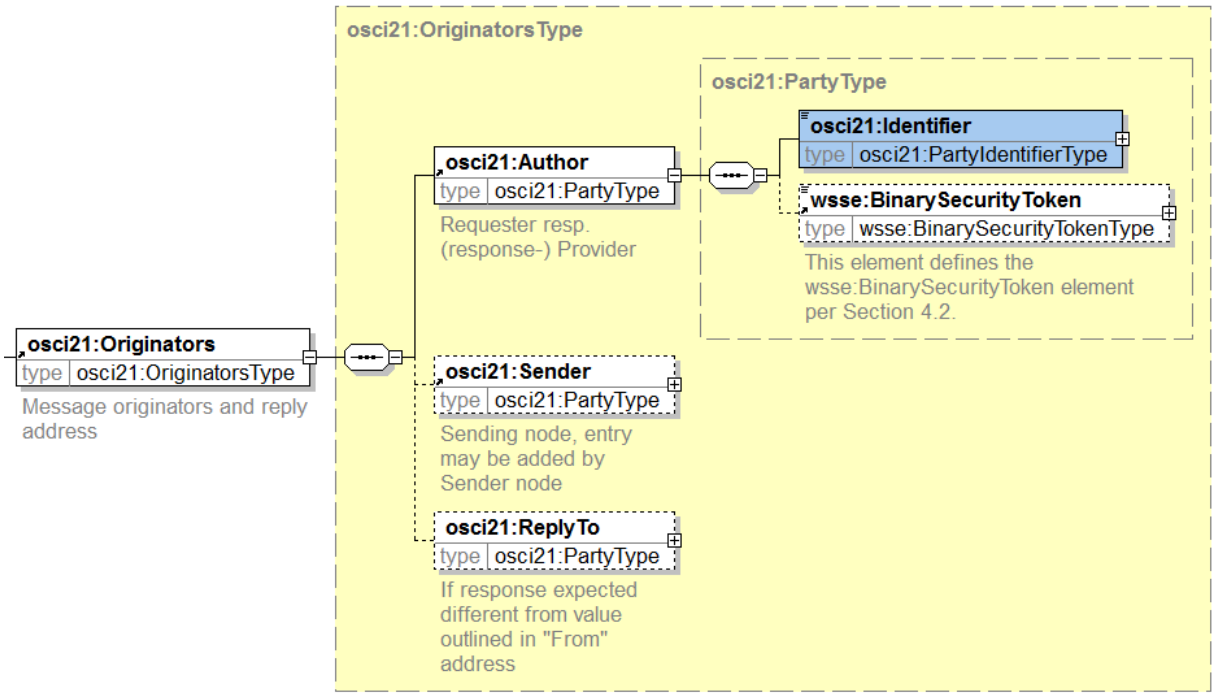
- **ReceiptRequests**: Quittungsanforderungen als Sequenz von leeren Elementen. Die Quittungsanforderungen sind beim Knoten Sender konfigurierbar gemäß Szenario oder sie werden explizit durch den Initiator gesetzt. Wenn sie durch den Initiator gesetzt sind, haben sie Vorrang vor der Konfiguration. Die Umsetzung erfolgt vom sendenden OSCI-Gateway in den Receipt-/Notification Requests.



Für die Parametrierung der Receipt-/Notification Requests gelten folgende Default-Werte: Während Quittungen standardmäßig zum Sender ausgeliefert werden, ist ein „Echo“ des übermittelten Payload häufig nicht gewollt.

Ist in **ReceiptRequests** auch „Submission“ gesetzt, erzeugt der Knoten „Sender“ eine Quittung „ **SubmissionReceipt** “. Das Format entspricht dem des „**DeliveryReceipt** “.

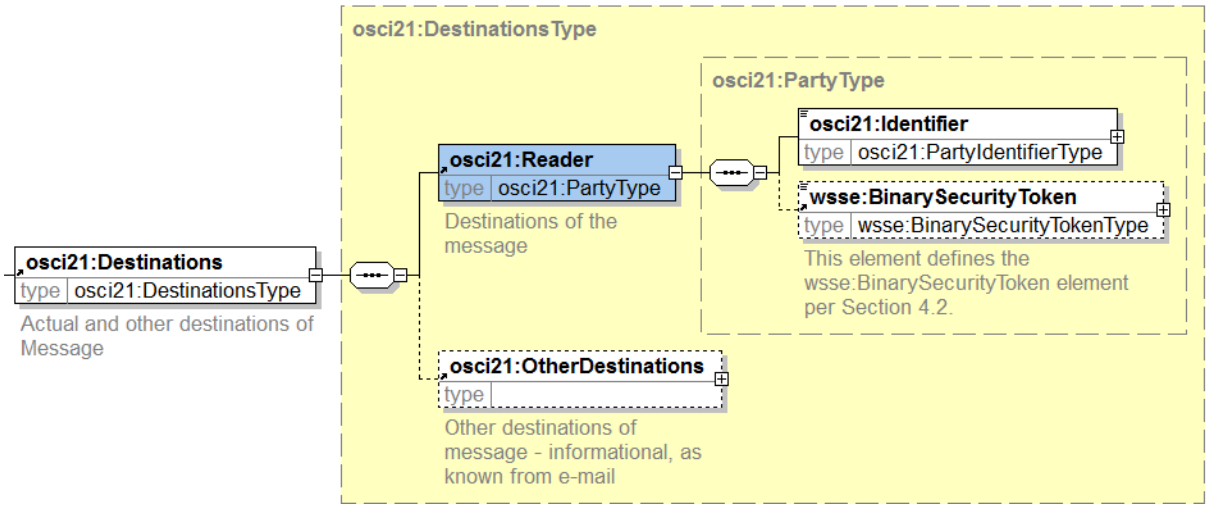
4.4.2.3.1.3 Originators



Die Elemente sind vom Typ **PartyType.** Neben dem generischen Identifier (**PartyIdentifierType)** können Authentisierungstoken (X509, auch SAML) aufgenommen werden.

- **Author:** Obligatorisch. Entspricht Initiator, Source Application, WS-Addressing **From.**
- **Sender:** Sendender Knoten / OSCI Gateway, z.B. XTA-WS.
- **ReplyTo:** Zieladresse, an die Antwort gesendet werden soll; Default = **Author,** dies entspricht der Semantik von WS-Addressing.

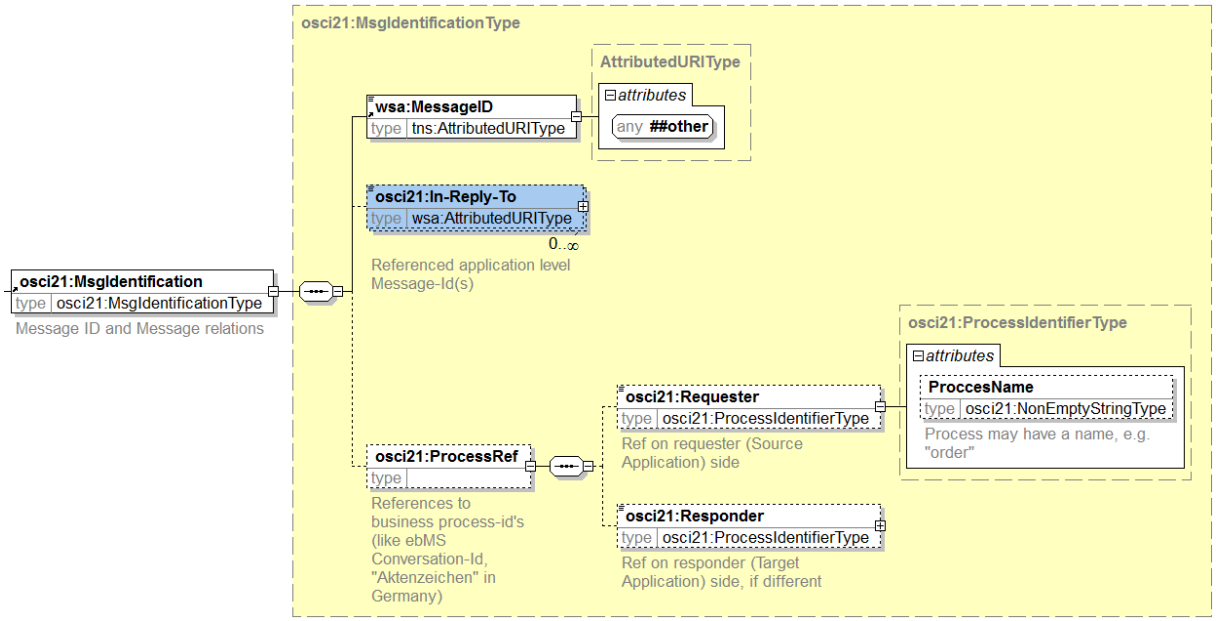
4.4.2.3.1.4 Destinatons



- **Reader:** Obligatorisch; entspricht Target Application, Service Provider. Muss vom Autor gesetzt werden.

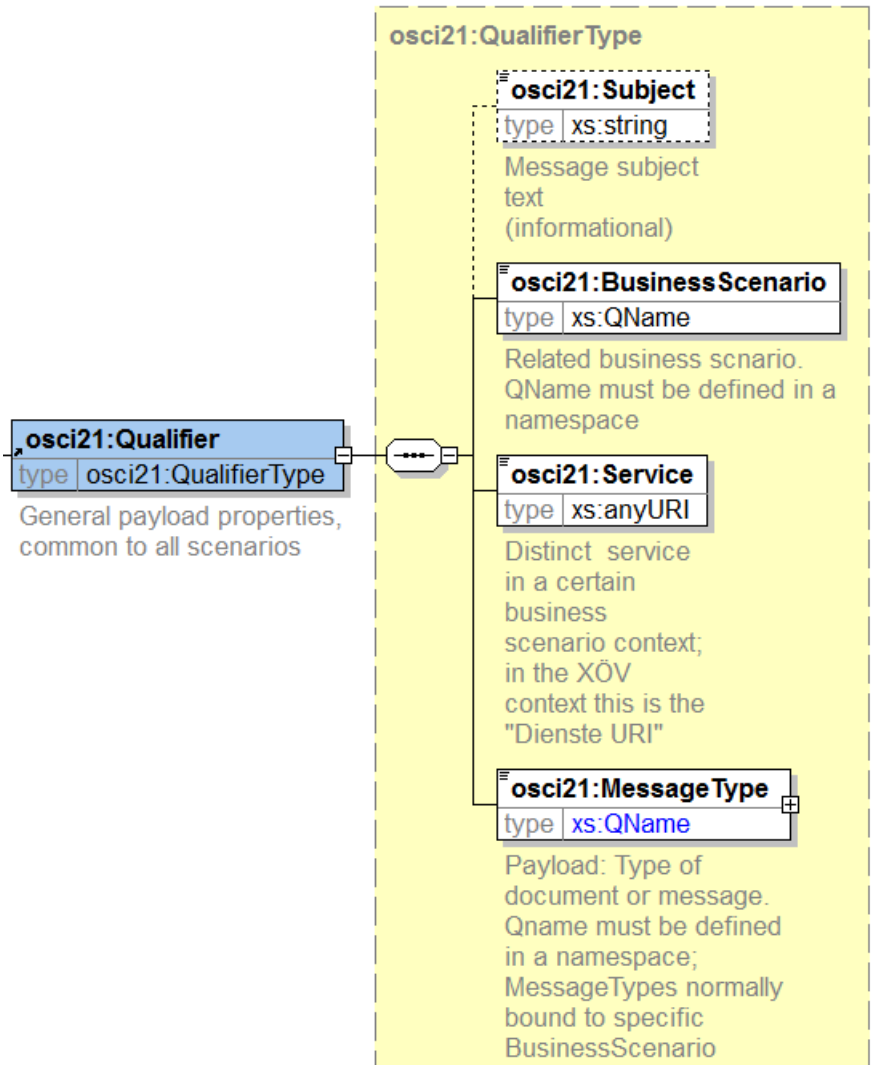
- **OtherDestinations:** Informatorisch, optionaler Eintrag: weitere Adressaten der Nachricht, unterschiedlich in **OtherReaders** und **CCReaders**. Es ist nicht vorgesehen, zum generischen „ **PartyIdentifer** “ auch optionale Authentisierungstoken aufzunehmen.

4.4.2.3.1.5 MsgIdentification



- **MessageID:** Obligatorischer eindeutiger Identifikator der Nachricht, vom Autor vergeben.
- **In-Reply-To:** bezogene Nachricht(en) (auf Applikations-/ Fachebene)
- **ProcessRef:** Bezug zu laufendem Vorgang (z.B. Aktenzeichen)
  - Es ist eine Unterscheidung zwischen Vorgangsnummern auf der Seite des Autors und Lesers möglich.
  - Die Vorgangsnummer kann mit einem Vorgangsnamen **ProcessName** attributiert werden.

4.4.2.3.1.6 Qualifier

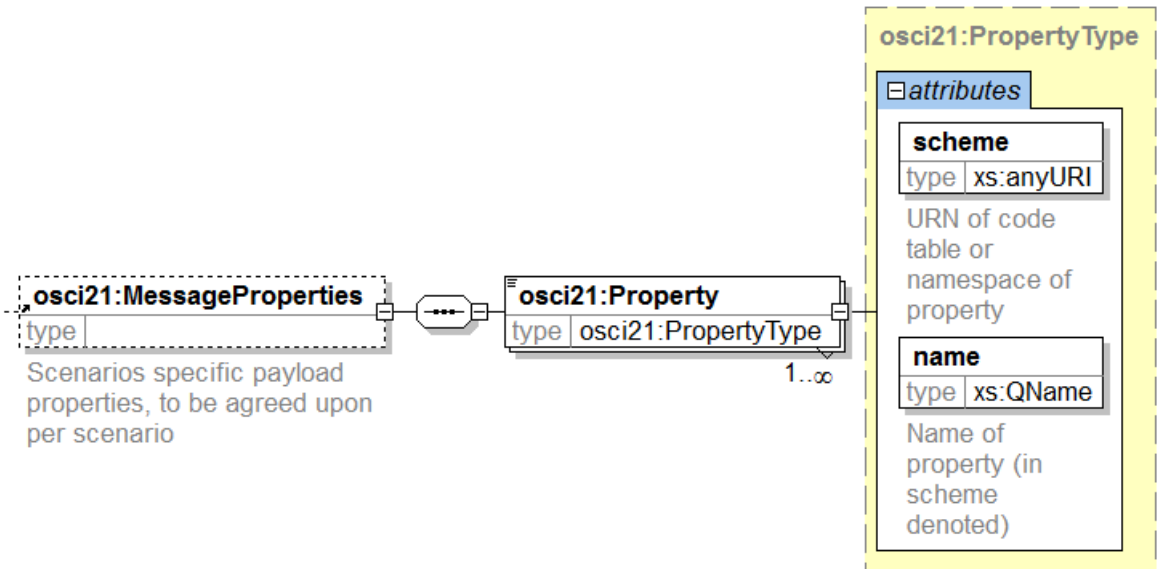


- **Subject:** Informatorischer Begleittext.
- **BusinessScenario:** Obligatorisch, Geschäftsszenario als **xs:QName** (ersetzt das OSCI2.0 „BusinessScenarioType).
- **Service:** Obligatorisch, Dienst des Lesers (Target Application) als URI.
- **MessageType** (**xs:anyURI):** Obligatorischer Nachrichtentyp innerhalb des Geschäftsszenarios als **xs:QName** ; er bindet ein definiertes Payload-Schema; wie im Meldewesen gebräuchlich ist es ggf. attributiert durch eine **@version.**

4.4.2.3.2 Szenarienspezifische Metadaten zum Payload

4.4.2.3.2.1 MessageProperties

Dieses Element nimmt bei Bedarf Metainformation auf, die bezogen auf ein spezielles Szenario benötigt wird. Semantik und Verarbeitung dieser Informationen erfolgen fachbezogen.



MessageProperties

- Dieser Container nimmt **Property** -Elemente auf, die innerhalb eines definierten Geschäftsszenarios festzulegen sind (Name, Werte, Semantik).

Property

- Eine Property ist ein Key/Value-Pärchen innerhalb eines Namensraums. Für Kommunikationsszenarien innerhalb eines definierten Geschäftsszenarios müssen entsprechende Listen festgelegt werden, um eine identische semantische Interpretation sicher zu stellen. Das Element ist vom Type **xs:anySimpleType** und kann damit z.B. auch base64-codierte Werte aufnehmen, wie z.B. vom XTA-WS adressiert.

Property, @ scheme

- Das Attribut **@scheme** definiert den Namensraum einer Property, z.B. Geschäftsszenario.
- Der XTA-WS bezieht sich für die Payload-Auszeichnung u.a. auf XÖV-Codetabellen. Der Bezug zu den „Properties“ kann auch durch eine entsprechende URN als Wert im Attribut **@scheme** von **Property** abgebildet werden.

4.4.3 Schnittstellentyp msgBoxPort

Für den asynchronen Nachrichtenaustausch holt der Leser, der in der Regel meist nicht online erreichbar ist, die an ihn adressierten Nachrichten aus einem „Postkasten“ (auch „Message-Box“ genannt) ab. Dort sind die Nachrichten zwischengespeichert. (Der Leser nutzt hierfür einen Pull-Mechanismus.)

Im Rahmen des Abholens der Nachrichten werden dem Leser mehrere Funktionen angeboten: Der Leser ruft direkt alle Nachrichten oder aber einzelne Nachrichten aus dem „Postkasten“ ab, oder er läßt sich Statuslisten geben, aus denen ablesbar ist, wieviele Nachrichten noch nicht abgeholt wurden.

Diese Angebote werden durch einen Service von OSCI 2 spezifiziert, der in XTA-WS genutzt wird: Im Schnittstellentyp msgBoxPort werden alle „Postkasten-Funktionen“ für den XTA-WS zusammengefasst:

- getStatusList
- getMessage

- close

4.4.3.1 Methode getStatusList (Abruf einer Liste bzw. Teilliste von Metadaten und MessageIDs)

Mit der Methode getStatusList kann der Leser vom Empfänger Informationen (MessageId und Metadaten) über eingegangenen Nachrichten abrufen, bzw. prüfen, ob Nachrichten bereitstehen.

Um die Ergebnisliste einzuschränken, kann der Leser dem Methodenaufruf Selektionskriterien (Stati, Zeiträume) mitgeben. Die Rückgabeliste enthält pro Nachricht, die den Auswahlkriterien entspricht, Metainformationen, z.B. Autor, Leser, Subjekt, MessageID.

Mithilfe der Ergebnisliste entscheidet der Leser, welche Nachrichten er tatsächlich abholen möchte. Für die Abholung wird die Methode getMessage (siehe Abschnitt 4.4.3.2 auf Seite 63) verwendet.

Ein mögliches Anwendungsszenario ist der Abruf einer Liste, die die MessageIDs nicht gelesener, nicht abgerufener Nachrichten enthält. Dies kann sinnvoll sein, wenn es bei der Verarbeitung von Nachrichten zu Abbrüchen kam.

Optionen zur Definition der Selektion:

- Ohne Angabe eines Zeitraums (Parameter <MsgBoxEntryTimeFrom> und <MsgBoxEntryTimeTo> leer) werden in der Ergebnisliste alle Nachrichten berücksichtigt.
- Ist ein Zeitraum angegeben, ist die Ergebnisliste entsprechend eingeschränkt. Bei der Zusammenstellung der Ergebnisliste wird die Löschfrist berücksichtigt.
- In beiden Fällen kann die Ergebnismenge durch das Setzen des Attributs "newEntry" auf neue Nachrichten, die also noch nicht vom Leser abgeholt wurden, eingeschränkt werden.

4.4.3.1.1 Ergebnisse getStatusList

- Liste der Ergebnisparameter. Liegen für die Selektionskriterien keine Nachrichten vor, ist die Liste leer. Im Ergebnis-Header (MsgBoxResponse) werden Zusatzinformationen zum Anfragevorgang und seiner Ergebnisliste geliefert.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.
- Der technische Fehler (SoapFault) <InvalidMessageIdException> entsteht, wenn die angeforderte Nachricht dem Account nicht bekannt ist.

4.4.3.1.2 Operation getStatusList

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MsgBoxStatusListRequest	osci:MsgBoxStatusListRequestType

Wesentliche Parameter:

- „osci:MsgBoxEntryTimeFrom“: Wenn ein Zeitpunkt angegeben ist, werden nur die MessageIDs und Metadaten von Nachrichten geliefert, die nach diesem Zeitpunkt empfangen wurden.
- „osci:MsgBoxEntryTimeTo“: Wenn ein Zeitpunkt angegeben ist, werden nur die MessageIDs und Metadaten von Nachrichten geliefert, die bis zu diesem Zeitpunkt empfangen wurden.

- „osci:newEntry“: Festlegung, ob alle Nachrichten oder nur neue Nachricht berücksichtigt werden sollen.
- „osci:maxListItems“: Maximale Anzahl von Einträgen, die je Zugriff - also auch bei den nachfolgenden Aufrufen der Methode getNextStatusList() - zurückgegeben werden soll.

Output.

Soap Part	Name	Type
Header	MsgBoxResponse	osci:MsgBoxResponseType
Body	MsgStatusList	osci:MsgStatusListType

Rückgabewerte:

- "osci:MsgBoxRequestID“: Die Ressourcenkennung für die weiteren Zugriffe auf den Iterator.
- „osci:NoMessageAvailable“: Angabe des Grundes, wenn zu den angegebenen Suchargumenten keine Daten gefunden wurden.
- „osci:ItemsPending“: Anzahl der gefundenen Nachrichten.
- „osci:MsgStatusList“: Wurden passende Nachrichten gefunden, dann werden je Nachricht folgende Informationen übermittelt:
  - „wsa:MessageID: Die MessageID des Transportauftrags.
  - „wsa:RelatesTo: Es können eine beliebige Anzahl von Referenzen auf eine Objekte wie z. B. Vorgängernachrichten vorhanden sein.
  - „wsa:From“: Kennung des Autoren.
  - „osci:TypeOfBusinessScenario“: Angabe (Kennung) des Geschäftsprozesses
  - „osci21:MsgSize“: Größe der Nachricht.
  - „osci21:ObsoleteAfter“: Nach dem Erreichen dieses Zeitpunkts ist die Nachricht nicht mehr auszuliefern sondern zu löschen
  - „osci21:Delivery“: Zeitpunkt des Eintreffens beim Empfänger inkl. dessen Transportinfrastruktur, z. B. einem OSCI Intermediärs.
  - „osci21:InitialFetch“: Zeitpunkt der erstmaligen Auslieferung zum Leser.

4.4.3.1.3 Beispielcode (Aufruf der Methode)

```
getStatusList(„2013-03-01T02:00:00“, „2013-03-01T12:00:00“, true, 5)
```

(Angabe des zu berücksichtigenden Zeitraums, Berücksichtigung aller (- also nicht nur neuer) Nachrichten, Abholung von max. 5 Nachrichten)

4.4.3.2 Methode getMessage (Abholen einer Nachricht)

Selektionskriterium Message ID:

Mit der Methode getMessage holt der Leser eine Nachricht vom Empfänger ab. In der ersten Version der XTA-WS wird ausschließlich das Selektionskriterium MessageID unterstützt.

Die Identifikation der Nachricht erfolgt also durch die MessageId, die als Ergebnis der Methode <getStatusList> zurückgegeben wurde.

Wenn der Leser gezielt eine MessageID benennt, erhält er die entsprechende Nachricht. Bei dieser gezielten Abholung ist nicht relevant, ob die Nachricht bereits früher abgeholt wurde.

Der Empfang abgeholter Nachrichten muss vom Leser gegenüber dem Empfänger (mit der Methode close) quittiert werden.

Meist ruft der Leser die Methode getMessage auf, bis alle Nachrichten, deren MessageIds durch die Methode getStatusList ermittelt wurden, abgeholt wurden.

Neben der unmittelbaren Abholung kann der Leser Nachrichten zur Abholung reservieren. Hierfür übergibt er dem Empfänger die Liste der entsprechenden MessageIDs. Er bekommt vom Empfänger eine Ressourcenkennung, mit der er die reservierten Nachrichten abholen kann. Sollte der Leser die Nachrichten nicht abholen, stehen sie nach der Freigabe der Ressourcen (Methode close (siehe Abschnitt 4.4.3.3 auf Seite 65) oder nach einem definierten Zeitraum wieder zur Verfügung.

Nutzung weiterer Selektionskriterien:

Perspektivisch hat der Leser neben der MessageID weitere Selektionskriterien zur Auswahl. Für die Methode getMessage sind ähnliche Parameter vorgesehen wie sie für die Methode getStatusList beschrieben sind, siehe außerdem Asynchroner Empfang, Variante II, die in Anhang C dokumentiert wird und nicht im vorleigenden XTA-WS umgesetzt ist: Der Leser kann milthilfe dieser Variante dann Nachrichten vom Empfänger unter Angabe von diesen Kriterien abholen. Hierfür kann der Leser einen **Iterator** verwenden. Mit diesem verwaltet der Empfänger für die Dauer der Abholung die Liste der noch abzuholenden Nachrichten. Damit muss diese Arbeit nicht vom Leser übernommen werden. Das ist vor allem vorteilhaft, wenn mehrere Leser gleichzeitig Nachrichten abholen wollen: Der Status der Nachricht wird beim Empfänger verwaltet. Der Empfänger liefert beim Erzeugen des Iterators die Ressourcenkennung des Iterators zusammen mit der ersten Nachricht, die den Selektionskriterien entspricht, zurück.

4.4.3.2.1 Ergebnisse

- Eine Nachricht wird im <GenericContentContainer> zurückgeliefert. Hierbei ist zu beachten, dass für eine abgeholte Nachricht der Status auf "abgeholt" ändert, nachdem die Transaktion durch Aufruf der Methode <close> bestätigt worden ist.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.
- Der technische Fehler (SoapFault) <InvalidMessageIdException> entsteht, wenn die angeforderte Nachricht dem Account nicht bekannt ist.

4.4.3.2.2 Operation getMessage

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MsgBoxFetchRequest	osci:MsgBoxRequestType

Wesentliche Parameter:

- „osci:MessageBoxRequestType“: Liste der MessageIDs der zu reservierenden Nachrichten
- „wsa:MessageID“: Angabe der MessageID der abzuholenden Nachricht

Output.

Soap Part	Name	Type
Header	MsgBoxResponse	osci:MsgBoxResponseType
Body	GenericContentContainer	xta:GenericContentContainer

4.4.3.2.3 Beispielcode (Aufruf der Methode)

Abholung einer Nachricht durch Angabe der MessageID:

```
getMessage(“ DE_NDS_KDO: 743ab5e0-8277-11e2-9e96-0800200c9a66 “)
```

4.4.3.3 Methode close (Quittierung der Abholung)

Mithilfe der Methode close soll sichergestellt werden, dass Nachrichten oder Listen nicht mehrfach verarbeitet werden: Der Leser bestätigt dem Empfänger durch eine entsprechende Quittierung, dass Nachrichten und (Teil-)listen erfolgreich abgeholt werden konnten. Diese Empfangsquittierung soll möglichst zeitnah erfolgen, so dass gewährleistet wird, dass jede vom Leser verarbeitete Nachricht beim Empfänger als gelesen markiert wurde. Der Verlust von Nachrichten und Listen kann erkannt werden, wenn die erwarteten Quittierungen fehlen.

Mit der Methode close wird damit eine Ressource bei Sender oder Empfänger wieder freigegeben. Dies kann ein Iterator sein, der beim Abruf von Teillisten benötigt wurde. Sie beendet also die Transaktion nach Erhalt einer Ergebnisliste, die durch die Methode getStatusList erzeugt wurde und sie bestätigt die Abholung von Nachrichten.

4.4.3.3.1 Ergebnisse

- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <InvalidMessageIdException> entsteht, wenn die angeforderte Nachricht dem Account nicht bekannt ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.

4.4.3.3.2 Operation close

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MsgBoxCloseRequest	osci:MsgBoxCloseRequestType

Wesentliche Parameter:

- „osci:MsgBoxRequestId“: Angabe der Ressourcenkennung, die beim Abholen der Nachrichten oder Teillisten verwendet wurde. Die Ressource wird hierdurch freigegeben.
- „wsa:AttributedURIType“: In diesem optionalen Parameter kann die zu quittierende Nachricht durch die Angabe der MessageID referenziert werden.

Output. Keine Rückgabewerte.

4.4.3.4 Wichtige Objekte der OSCI-msgBox-Schnittstelle

4.4.3.4.1 MsgSelector

- Selektionsmöglichkeiten der Pull-Mechanismen auf die OSCI Message-Box. Meist reicht der Zugriff über die dezidierte MessageId oder den Status (z.B. „nur neu eingegangene Nachrichten“). Es wird freigestellt, nur die jeweils benötigten Pull-Mechanismen zu implementieren.
- Für spezifische zusätzlich benötigte Zugriffsmöglichkeiten wird eine Selektion über flexible Mechanismen zugelassen, die sich auf alle Elemente und Attribute des Metadata-Headers beziehen.

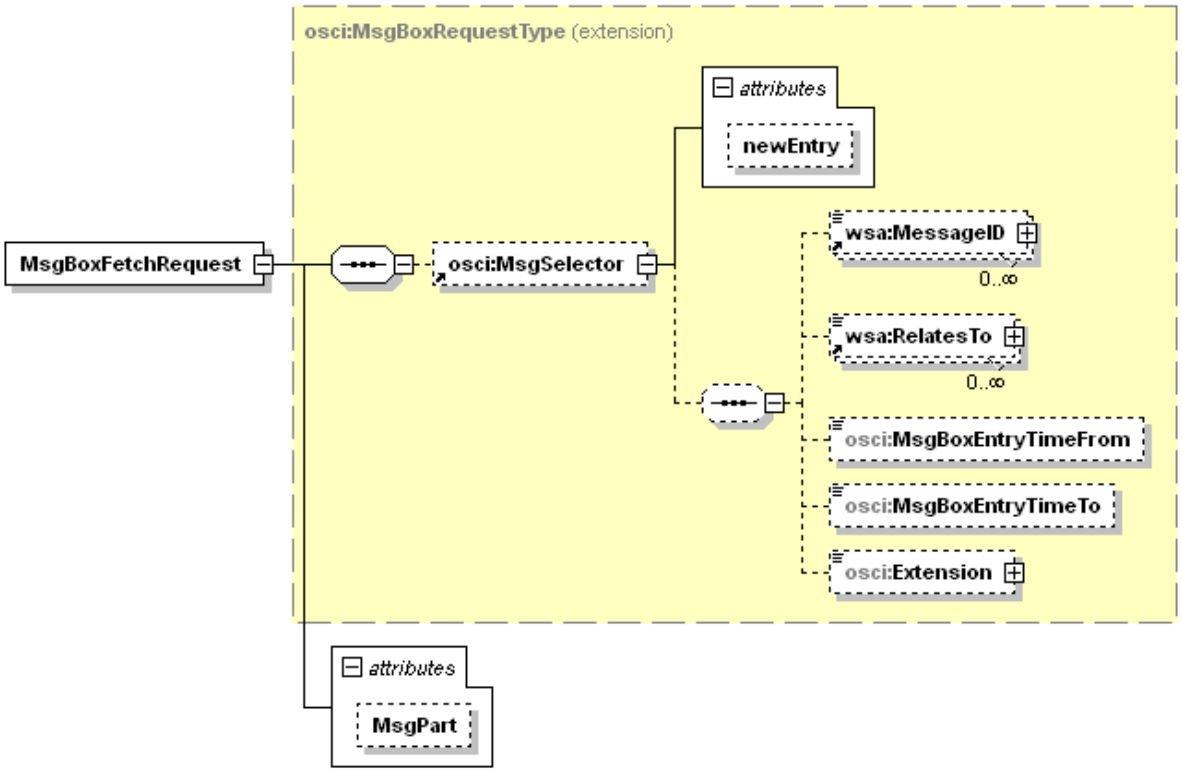
4.4.3.4.2 MsgBoxResponse

- MsgBoxResponse ist die Antwort zu MsgBoxFetchRequest.
  - Für den Body der Nachricht ist durch eine MsgBoxResponse der folgende Container spezifiziert:
    - Body (Inhaltsdaten) der zugestellten Nachrichten,
    - Metadaten-Container,
    - Security Token (Transport!) des WS-Security Headers
    - Message Time Stamps

    Außerdem, falls in der Ursprungsnachricht vorhanden:
    - Header-Elemente X509TokenContainer, xkms:CompoundResult
    - ReceptionReceiptDemand

4.4.3.4.3 MsgBoxFetchRequest

- Welche Informationen durch MsgBoxResponse bereitgestellt werden sollen, wird durch das Attribut **@MsgPart** in **MsgBoxFetchRequest** festgelegt:



- **@MsgPart** ist vom Typ **xs:NMTOKEN** mit folgenden Ausprägungen:
  - **Envelope:** Stellt alle oben genannten Informationen (Header- und Body-Blöcke der ursprünglichen Nachricht) als **s12:Envelope** Element im Body der **MsgBoxResponse** bereit.
  - **Body:** Stellt nur den Body der ursprünglichen Nachricht im Body der **MsgBoxResponse** innerhalb eines **s12:Body** Elements bereit.
  - **Header:** Stellt nur die oben genannten Header-Blöcke der ursprünglichen Nachricht im Body der **MsgBoxResponse** innerhalb eines **s12:Header** Elements bereit.

4.4.4 Schnittstellentyp sendSynchronPortType - Leser (Synchroner Versand einer Nachricht)

Mit der Methode sendMessageSync (- Leser) kann der Empfänger den synchronen Transportauftrag, den er vom Sender erhalten hat, direkt an den Leser zur Erfüllung kommunizieren. Diese Methode wird also als Service vom Leser angeboten. Die Antwort des Lesers wird innerhalb derselben Transaktion an den Sender weitergereicht, der dann dem Autor das Ergebnis als Reaktion auf dessen Anfrage (mithilfe der Methode sendMessageSync - Sender) übergeben kann.

Damit der Leser die Methode implementieren kann, ist sie im Auslieferungsumfang XTA in einer separaten WSDL enthalten.

Die mitzugebenden Informationsblöcke, Struktur, Fehler und Beispielcode sind dieselben wie im Fall der strukturell identischen Methode sendMessageSync (- Sender).

4.5 Das Informationsmodell

In diesem Abschnitt sind alle Informationsobjekte dokumentiert, die innerhalb des Standards XTA modelliert sind. Sie bilden das XTA-Informationsmodell.

Weitere Informationsobjekte werden aus externen Standards eingebunden (vgl. Anhang F, *Eingebundene externe Modelle*). Sie sind nicht Bestandteil des XTA-Informationsmodells, also auch nicht im vorliegenden Abschnitt dokumentiert.

4.5.1 Typen des XTA-Baukastens

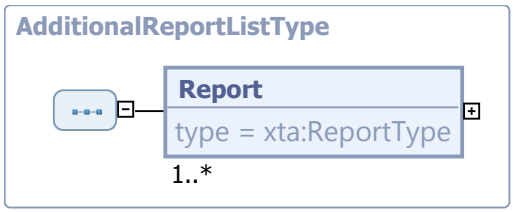
Hier werden die Bausteine beschrieben, aus denen sich die Geschäftsobjekte der Methodenaufrufe des XTA-WS zusammensetzen.

4.5.1.1 AdditionalReportListType

Typ: *AdditionalReportListType*

Hier können ergänzend spezielle Reports in den XTA-TransportReport eingetragen werden, deren Struktur außerhalb von XTA definiert ist.

Abbildung 4.1. AdditionalReportListType



Kindelement von *AdditionalReportListType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Report	*xta:ReportType*	**1..n**	4.5.1.15	75
Einer der ergänzend eingefügten Reports in einem Drittformat. Das Drittformat beinhaltet die technisch neutrale Darstellung (base-64-codiert) des Reports: Es ist somit kein konkretes XML-Format festgelegt, um die Struktur der Reportdaten abzubilden.				

4.5.1.2 Code.RecordType

Code	Code.RecordType
Beschreibung	Diese Tabelle enthält die Arten von Meldungen im Protokoll (TransportReport), also Arten von Fehler-, Warn- und Informationseinträgen. Die Schlüsseltabelle wird im XRepository (www.xrepository.de) im XML-Format OASIS Genericode über die ID urn:de:kosit:xta:codelist:Record bereitgestellt.
Codelisten-Nutzung	Typ: 3, nähere Beschreibung der Codeliste siehe Seite 119
Codelisten-URI	urn:de:kosit:xta:codelist:Record
Codelisten-Version	

4.5.1.3 Code.ReportType

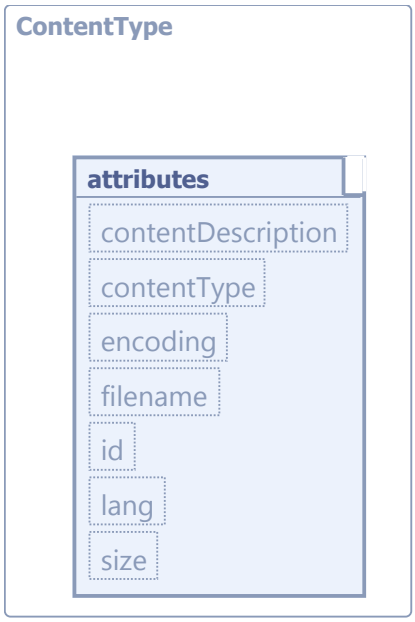
Code	Code.ReportType
Beschreibung	Diese Tabelle enthält die möglichen Arten von Reports, die in eigenen Formaten in dem Protokoll (TransportReport) eingefügt werden können. Die Schlüsseltabelle wird im XRepository (www.xrepository.de) im XML-Format OASIS Genericode über die ID urn:de:kosit:xta:codelist:Report bereitgestellt.
Codelisten-Nutzung	Typ: 3, nähere Beschreibung der Codeliste siehe Seite 120
Codelisten-URI	urn:de:kosit:xta:codelist:Report
Codelisten-Version	

4.5.1.4 ContentType

Typ: *ContentType*

Typ für die technisch neutrale (base64-kodierte) Darstellung von Information. Enthält den base64-codierten Inhalt, der zwischen WebService-Client und XTA transportiert wird. Die Attribute sind der MIME-Spezifikation, RFC 2183 entnommen.

Abbildung 4.2. ContentType



Dieser Typ ist eine Erweiterung des Basistyps *xs:base64Binary*.

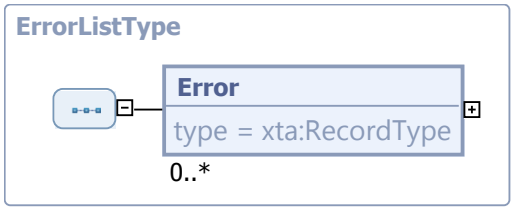
Kindelemente von *ContentType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
contentDescription	*osci21:NonEmptyStringType*	**0..1**	F.1	121
Beschreibung des fachlichen Inhalts, z.B. 'Angebot' oder 'Rechnung'.				
contentType	*osci21:NonEmptyStringType*	**1**	F.1	121
Dieses Attribut nennt den MIME-Typ des enthaltenen Inhalts, hat also Einträge wie text/xml, text/plain, application/gzip oder application/pdf				
encoding	*osci21:NonEmptyStringType*	**0..1**	F.1	121
Der Zeichensatz, der der Kodierung des Inhalts zugrunde gelegen hat.				
filename	*osci21:NonEmptyStringType*	**0..1**	F.1	121
Der Dateiname der Datenquelle, falls der Inhalt einer Datei entnommen worden ist. Bsp.: Für die Übermittlung von xdomea-Nachrichten ist dieses Attribut Pflicht.				
id	*xs:ID*	**0..1**		
Bietet die Möglichkeit, den Inhalt über z.B. eine laufende Nummer zu referenzieren.				
lang	*xs:language*	**0..1**		
Sprache, in der der Inhalt formuliert ist.				
size	*xs:positiveInteger*	**0..1**		
Die Größe des Inhalts in Bytes.				

4.5.1.5 ErrorListType

Typ: *ErrorListType*

Struktur für eine Liste von Fehlermeldungen.

Abbildung 4.3. ErrorListType



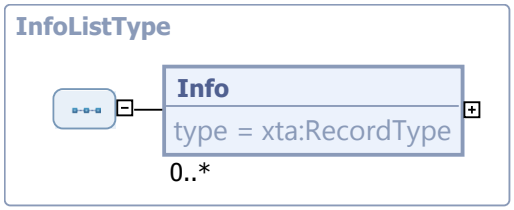
Kindelement von *ErrorListType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Error	*xta:RecordType*	**0..n**	4.5.1.14	75
Hier wird die Fehlermeldung mit ihren Parametern eingetragen.				

4.5.1.6 InfoListType

Typ: *InfoListType*

Struktur für eine Liste von Informationsmeldungen.

Abbildung 4.4. InfoListType



Kindelement von *InfoListType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Info	*xta:RecordType*	**0..n**	4.5.1.14	75
Hier wird die Informationsmeldung mit ihren Parametern eingetragen.				

4.5.1.7 IsServiceAvailableValueType

Typ: *IsServiceAvailableValueType*

Das Feld enthält die benötigten Attribute zum Ergebnis der Dienstanfrage: ob der Dienst angeboten wird oder nicht, oder ob diese Information generell nicht bekannt ist.

Abbildung 4.5. IsServiceAvailableValueType

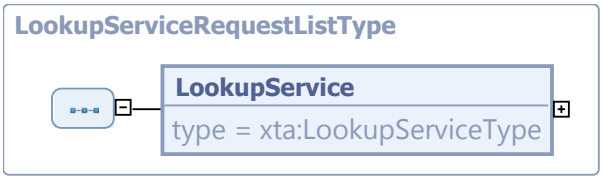
Kindelemente von *IsServiceAvailableValueType (Choice)*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
ServiceIsAvailable	*xs:boolean*	**1**		
Der Dienst wird angeboten (true) oder nicht angeboten (false).				
ServiceIsAvailableUnknown	*xs:boolean*	**1**		
Es ist nicht bekannt, ob der Dienst angeboten wird oder nicht.				

4.5.1.8 LookupServiceRequestListType

Typ: *LookupServiceRequestListType*

Dies ist die Struktur einer Service-Anfrage.

Abbildung 4.6. LookupServiceRequestListType



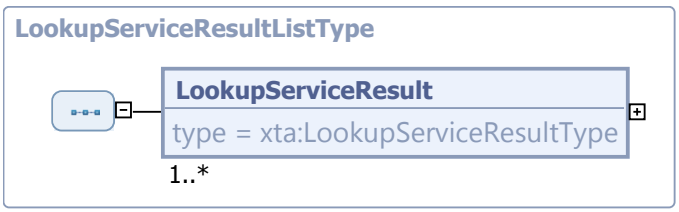
Kindelement von *LookupServiceRequestListType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
LookupService	*xta:LookupServiceType*	**1**	4.5.1.11	72
Dies ist eine Service-Anfrage. Sie enthält Daten zum potentiellen Diensteanbieter (Leser) und dem Dienst, der angefragt werden soll. Diese Anfrage dient dazu, zu ermitteln, ob der Dienst von diesem Anbieter angeboten wird, und über welche technischen Parameter er angesprochen werden kann.				

4.5.1.9 LookupServiceResultListType

Typ: *LookupServiceResultListType*

Eine Liste von Ergebnissen zu einer Liste von Diensteanfragen. Enthält die Liste der Anfragen, jeweils erweitert um die Ergebnisparameter.

Abbildung 4.7. LookupServiceResultListType



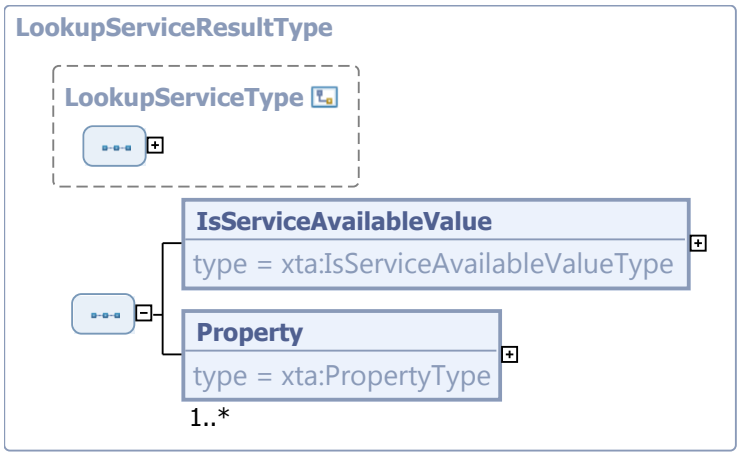
Kindelement von *LookupServiceResultListType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
LookupServiceResult	*xta:LookupServiceResultType*	**1..n**	4.5.1.10	71
Dies ist die Struktur der Liste von Ergebnissen zur Liste von Diensteanfragen.				

4.5.1.10 LookupServiceResultType

Typ: *LookupServiceResultType*

Das Ergebnis zu einer Dienstanfrage, das die Information enthält, ob der Dienst angeboten wird. Außerdem sind die nötigen technischen Paramter für die Erreichbarkeit vorhanden.

Abbildung 4.8. LookupServiceResultType



Dieser Typ ist eine Erweiterung des Basistyps *LookupServiceType* (siehe Abschnitt 4.5.1.11 auf Seite 72).

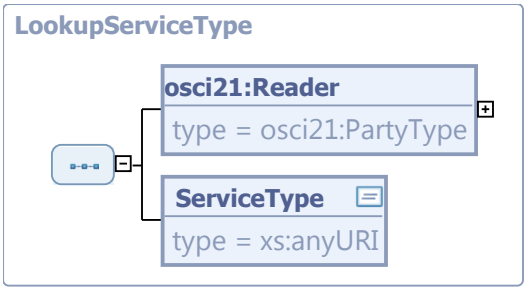
Kindelemente von *LookupServiceResultType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
IsServiceAvailableValue	*xta:IsServiceAvailableValueType*	**1**	4.5.1.7	70
Enthält das Ergebnis der Dienstanfrage: ob der Dienst angeboten wird oder nicht oder ob diese Information generell nicht bekannt ist.				
Property	*xta:PropertyType*	**1..n**	4.5.1.13	74
Enthält im Erfolgsfall die benötigten technischen Parameter für die elektronische Kommunikation mit dem Leser, z.B. das öffentliche Zertifikat des Lesers zur Inhaltsdatenverschlüsselung. Das Feld ist optional zu füllen, d.h. falls der angefragte Dienst angeboten und im Kontext der Parameter benötigt wird. Vom Fachszenario ist zu beschreiben, welche Parameter für die Erreichbarkeit der Dienste im Fachszenario anzuwenden sind.				

4.5.1.11 LookupServiceType

Typ: *LookupServiceType*

Dies ist die Struktur einer Service-Anfrage: Sie enthält die Daten über den Diensteanbieter (Leser) und den Dienst des Lesers, den der Autor in Anspruch nehmen will. Diese Anfrage dient dazu, zu ermitteln, ob der Dienst von diesem Anbieter angeboten wird und über welche technischen Parameter er angesprochen werden kann.

Abbildung 4.9. LookupServiceType



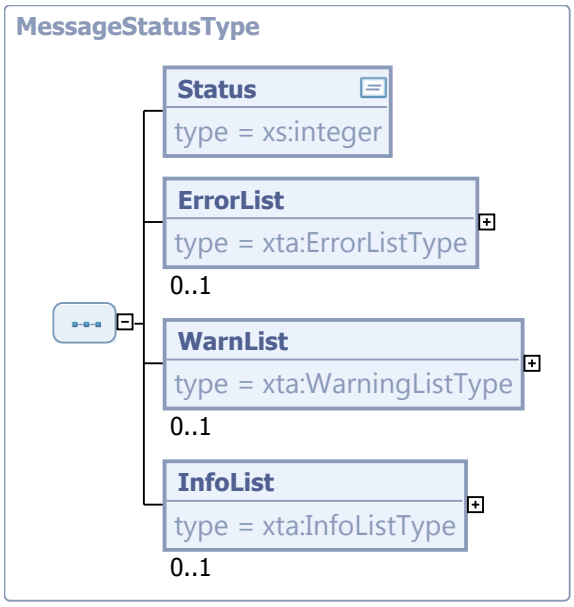
Kindelemente von *LookupServiceType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
osci21:Reader	*globales Element*	**1**	F.1	121
Dies ist die fachliche Identifizierung des Lesers. Der Wert entspricht z.B. dem DVDV-Behördenschlüssel.				
ServiceType	*xs:anyURI*	**1**		
Dies ist die Bezeichnung des anzufordernden Dienstes. Sie wird im Format einer URL übergeben, was den Vorteil hat, dass damit auch eine Versionsnummer eingeschlossen ist. Beispiel für Dienstebezeichnungen, wie sie im DVDV verwendet werden: http://www.osci.de/xmeld181/xmeld181Rueckmeldung.wsdl Abgrenzung: "Dienst" ist das, was gemäß Diensteeinteilung der Fachdomäne im Verzeichnisdienst als Service (im Sinne eines Web Service) eingetragen ist. Dadurch ist die Dienstebezeichnung weniger differenziert als der Nachrichtentyp. Typischerweise sind im Verzeichnisdienst mehrere Nachrichtentypen in einer Service-WSDL zusammengefasst.				

4.5.1.12 MessageStatusType

Typ: *MessageStatusType*

Gibt die Struktur für die Logging-Informationen über den Transportverlauf vor.

Abbildung 4.10. MessageStatusType



Kindelemente von *MessageStatusType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Status	*xs:integer*	**1**		
Wird durch Sender bzw. Empfänger fortgeschrieben. Wird der TransportReport noch fortgeschrieben, wird er hier mit 0=offen markiert. Nach Abschluss des TransportReports wird nach dem Max-Prinzip der höchste Ampelstatus aus den Elementen ErrorList, WarnList, InfoList hier numerisch dargestellt. • 0=offen: Die Nachricht befindet sich noch in der Verarbeitung. • 1=grün: Es sind keine Fehler oder Warnungen aufgetreten.				

Kindelemente von *MessageStatusType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
• 2=gelb: Es sind Warnungen, aber keine kritischen Fehler aufgetreten. • 3=rot: Es sind kritische Fehler aufgetreten.				
ErrorList	*xta:ErrorListType*	**0..1**	4.5.1.5	69
Liste der Fehlermeldungen.				
WarnList	*xta:WarningListType*	**0..1**	4.5.1.16	76
Liste der Warnungen.				
InfoList	*xta:InfoListType*	**0..1**	4.5.1.6	69
Liste der Infomeldungen.				

4.5.1.13 PropertyType

Typ: *PropertyType*

Das Feld enthält die benötigten Attribute für die elektronische Kommunikation mit dem Leser z.B. das öffentliche Zertifikat des Lesers zur Inhaltsdatenverschlüsselung. Das Feld ist optional.

Beispiel:

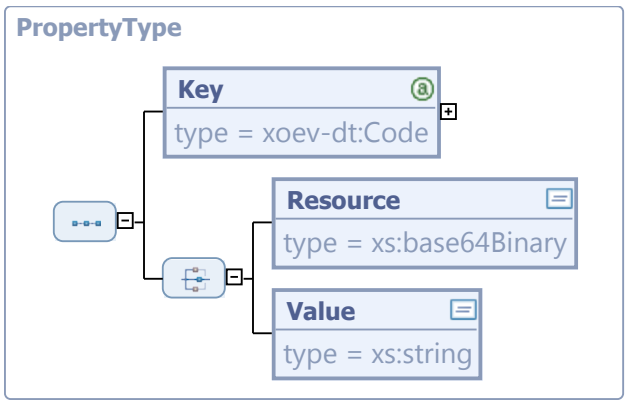
Der Term "Inhaltsdatenverschlüsselungszertifikat" wird in das Element code eingetragen, und das entsprechende Zertifikat in das Element xta:Resource (als base64String).

Beispiel Code

Property[0].Key.code = "InhaltsdatenVerschluesselungsZertifikat";

Property[0].Item = voDvdvDelivery.ReaderCipherCertificate.GetRawCertData();

Abbildung 4.11. PropertyType



Kindelemente von *PropertyType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Key	*xoev-dt:Code*	**1**	F.5	122
Hier wird die Art des Gegenstands (die Eigenschaft), um die es geht, bezeichnet. Ihre Bezeichnung muss als Code einer entsprechenden Codeliste definiert und verfügbar sein. Diese Bezeichnung wird ins Element xta:Key/code eingetragen. Auf die Codeliste wird in dieser Struktur referenziert (Attribute listURI und listVersionID).				

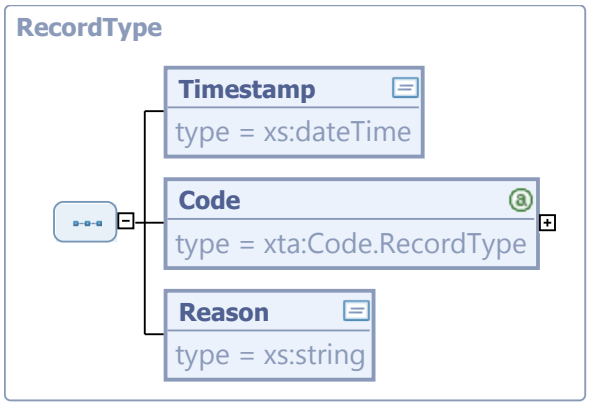
Kindelemente von *PropertyType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Choice		**1**		
Hier wird zum Key das für diesen Kontext gebrauchte Objekt oder der benötigte Wert eingetragen.				
Resource	*xs:base64Binary*	**1**		
Ein Objekt (base64-kodiert) zu einem im Kontext gegebenen Schlüssel, z.B. ein Zertifikat zum Schlüssel "Signaturzertifikat".				
Value	*xs:string*	**1**		
Ein Wert als String zu einem im Kontext gegebenen Schlüssel.				

4.5.1.14 RecordType

Typ: *RecordType*

Der Typ für alle Arten von Meldungen (Info-, Fehlermeldungen und Warnungen. Er sieht Meldungszeilen für Infos, Warnungen und Fehler vor.

Abbildung 4.12. RecordType



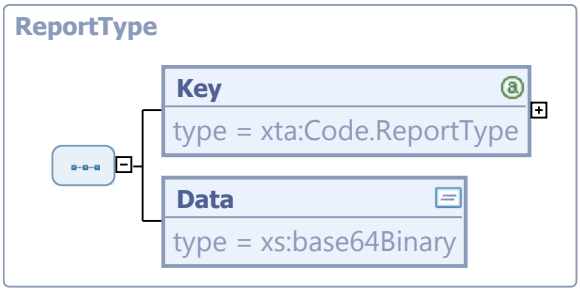
Kindelemente von *RecordType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Timestamp	*xs:dateTime*	**1**		
Zeitstempel				
Code	*xta:Code.RecordType*	**1**	4.5.1.2	68
Schlüssel der Fehlermeldung.				
Reason	*xs:string*	**1**		
Hier wird zur weiteren Erläuterung der Grund der Meldung als Freitext eingetragen.				

4.5.1.15 ReportType

Typ: *ReportType*

Die Struktur eines speziellen Reports, die außerhalb von XTA definiert ist. Die Art des Reports wird im Element Key angegeben (Beispiel: OSCI-Processcard). Der Report wird im Element Data eingetragen.

Abbildung 4.13. ReportType



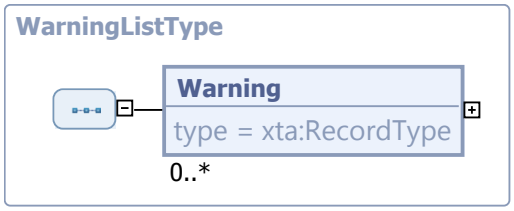
Kindelemente von *ReportType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Key	*xta:Code.ReportType*	**1**	4.5.1.3	68
Identifiziert den Typ des Reports auf der Basis einer Codeliste.				
Data	*xs:base64Binary*	**1**		
Hier wird der base64-codierte Report eingetragen.				

4.5.1.16 WarningListType

Typ: *WarningListType*

Struktur für eine Liste von Warnungen.

Abbildung 4.14. WarningListType



Kindelement von *WarningListType*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
Warning	*xta:RecordType*	**0..n**	4.5.1.14	75
Hier wird die Warnung mit ihren Paramtern eingetragen.				

4.5.2 Globale Elemente

Die in diesem Abschnitt beschriebenen Objekte sind in den Methodenaufrufen des XTA-WS eingebunden. Sie sind zusammengesetzt teils aus in XTA definierten Bestandteilen wie beschrieben in Section

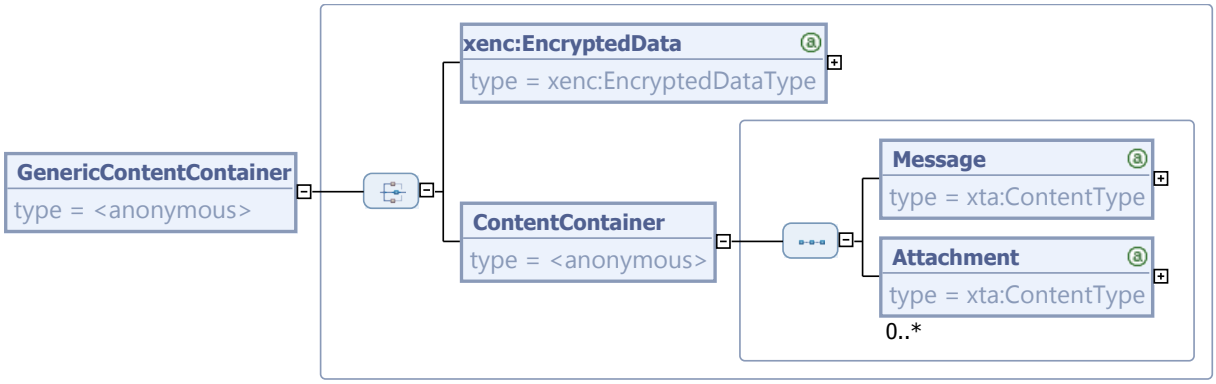
4.5.1 on page 67, teils aus Typen der externen Standards, die aufgeführt sind in Anhang F, *Eingebundene externe Modelle*)

4.5.2.1 GenericContentContainer

Nachricht: *GenericContentContainer*

Der GenericContentContainer nimmt den zu transportierenden oder abzuliefernden Inhalt auf, z.B. eine XÖV-Nachricht mit ihren Anlagen. Diese Inhalte können unverschlüsselt (Element ContentContainer) oder auch verschlüsselt (Element xenc:EncryptedData) hinterlegt werden. Die Verschlüsselung an dieser Stelle eignet sich für Ende-zu-Ende-Verschlüsselung durch den Autor, wenn dieses Objekt durch den Autor erstellt wird.

Abbildung 4.15. GenericContentContainer



Kindelemente von *GenericContentContainer (Choice)*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
xenc:EncryptedData	*globales Element*	**1**	F.3	121
Dieses Objekt ist dafür vorgesehen, den Container-Inhalt verschlüsselt zu hinterlegen. Im entschlüsselten Zustand müssen die Daten dem Schwester-Element ContentContainer entsprechen.				
ContentContainer		**1**		
Der ContentContainer enthält genau eine Nachricht (Element Message) und null bis beliebig viele Anlagen, die alle in technisch neutraler Darstellung (base64-codiert) eingefügt werden (Element Attachment). Die Gesamtgröße des Containers darf 40 MB nicht überschreiten.				
Message	*xta:ContentType*	**1**	4.5.1.4	68
Die zu übermittelnde Nachricht als primärer Inhalt dieses Containers ist optional durch Anhänge (Element Attachment) zu ergänzen. In die Attribute wird je nach Kontext Metainformation zur Nachricht eingetragen.				
Attachment	*xta:ContentType*	**0..n**	4.5.1.4	68
Hier können optional ergänzende Anhänge zur übermittelnden Nachricht eingefügt werden. Die Attribute transportieren je nach Kontext Metainformation zum enthaltenen Anhang.				

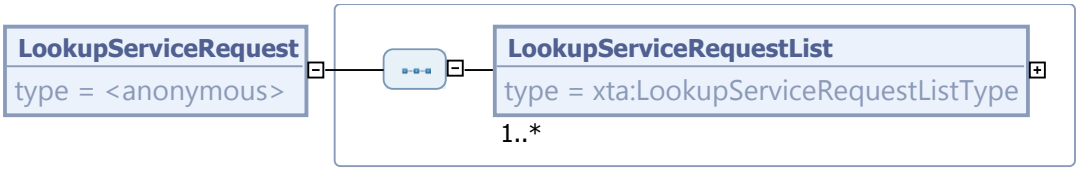
4.5.2.2 LookupServiceRequest

Nachricht: *LookupServiceRequest*

Dies ist eine Liste von Dienstanfragen.

Jede Anfrage dient dazu, zu ermitteln, ob der Dienst von diesem Anbieter angeboten wird, und über welche technischen Parameter er angesprochen werden kann.

Abbildung 4.16. LookupServiceRequest



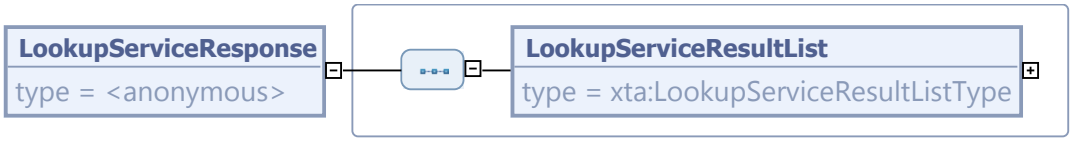
Kindelement von *LookupServiceRequest*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
LookupServiceRequestList	*xta:LookupServiceRequestListType*	**1..n**	4.5.1.8	70
Dies ist die Struktur für eine Liste von Dienstanfragen.				

4.5.2.3 LookupServiceResponse

Nachricht: *LookupServiceResponse*

Dies ist das Ergebnis zu einer Liste von Dienstanfragen, also eine Liste von Dienstanfrageergebnissen. Die Anfrage wird jeweils zitiert und das zugehörige Ergebnis ausgegeben.

Abbildung 4.17. LookupServiceResponse



Kindelement von *LookupServiceResponse*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
LookupServiceResultList	*xta:LookupServiceResultListType*	**1**	4.5.1.9	71
Die Struktur einer Liste von Dienstanfrageergebnissen.				

4.5.2.4 TransportReport

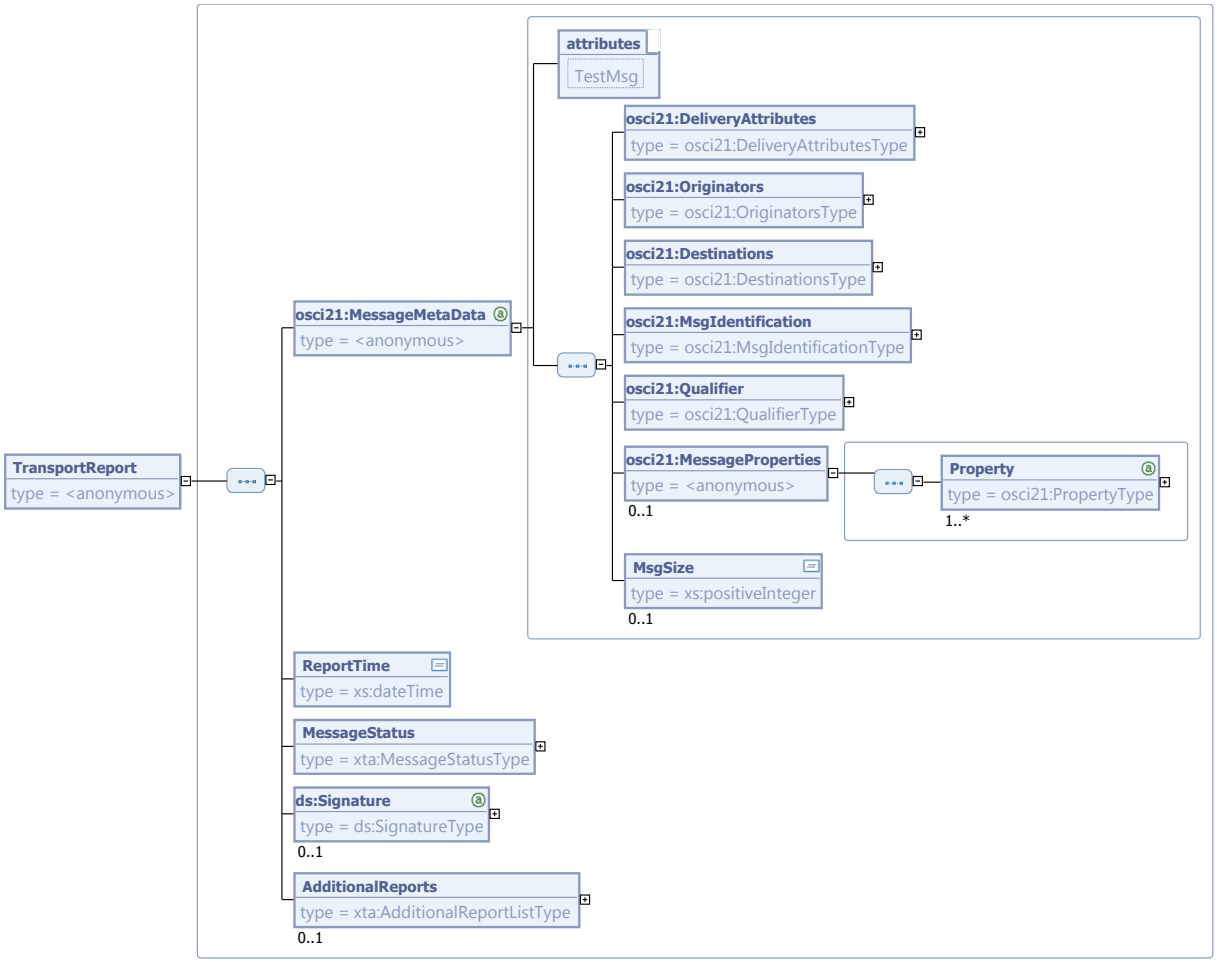
Nachricht: *TransportReport*

Der TransportReport ist die Struktur des durch XTA standardisierten Transportprotokolls. Neben den übermittelten Nachrichten ruft das Fachverfahren (in den Rollen Autor und Leser) über den Webservice-Client Zusatzinformationen über den Transportauftrag und die Transportereignisse vom XTA-WS ab.

Um Autor und Leser die Möglichkeit zu geben, die Abarbeitung ihrer Transportaufträge zu überwachen, erstellen Sender und Empfänger Transportprotokolle, die in einer XML-Struktur des Typs TransportReport dargestellt und für Abruf und Auswertung bereit liegen.

Die Datenstruktur aggregiert die Information zum erteilten Transportauftrag, zum Verlauf des sich anschließenden Transports einschließlich Zertifikatsüberprüfungen mit Ergebnissen.

Abbildung 4.18. TransportReport



Kindelemente von *TransportReport*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
osci21:MessageMetaData	*globales Element*	**1**	F.1	121
Der Container osci21:MessageMetaData umfasst alle Daten des erteilten Transportauftrags, auf dessen Ausführung sich der TransportReport bezieht. Zu den Informationen gehören die Identifizierung von Absender und (einem oder mehreren) Empfängern, Metainformation zu Inhalt und Identität der zu transportierenden Nachricht (Payload) sowie weitere Attribute, die die Auslieferung, Quittungen und Servicequalität betreffen. Weitere Informationen zu MessageMetaData sind in Abschnitt 4.4.2.3.1.1 auf Seite 54 zu finden.				
ReportTime	*xs:dateTime*	**1**		
Zeitpunkt der letzten Aktualisierung des Protokolls. Ist bei Fortschreibung des Protokolls zu überschreiben.				
MessageStatus	*xta:MessageStatusType*	**1**	4.5.1.12	73
Enthält Information über den Veralauf des Transports. Es werden hier Listen mit aufgetretenen Fehler-, Warnungs- und Informationsmeldungen geführt. Außerdem ist nach Schließung des Transportauftrags im Feld Status eine "Schnell-Info" verfügbar.				

Kindelemente von *TransportReport*				
Kindelement	**Typ**	**Anz.**	**Ref.**	**Seite**
ds:Signature	*globales Element*	**0..1**	F.4	122
Falls der TransportReport signiert ist, findet sich hier die Signatur.				
AdditionalReports	*xta:AdditionalReportListType*	**0..1**	4.5.1.1	67
Die AdditionalReports sind weitere Prüfberichte, z.B. die OSCI Processcard. Sie werden hier base64-codiert abgelegt.				

4.6 Fehler

4.6.1 Exceptions

Fehler des XTA-WS werden als SOAP 1.2 Exceptions geworfen. Es sind eine Reihe solcher Exceptions für den XTA-WS definiert:

- PermissionDeniedException
- ParameterIsNotValidException
- XTAWSTechnicalProblemException
- MessageSchemaViolationException
- MessageVirusDetectionException
- InvalidMessageIdException
- SyncAsyncException

Das nachfolgende Beispiel einer SOAP-Exception verdeutlicht die angewendete Struktur:

```
<?xml version="1.0! encoding="UTF-8"?>
<s:Envelope
 xmlns:s="http://www.w3.org/2003/05/soap-envelop"
 xmlns:xsi="http://www.23.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
                     http://www.w3.org/2003/05/soap-envelop/"
 <s:Body>
  <s:Fault>
   <s:Code>
    <s:Value>s:Receiver</s:Value>
    <s:Subcode>
     <s:Value xmlns:xta="http://xoev.de/xta/2013/07">
      xta:PermissionDeniedExeption
     </s:Value>
    </s:Subcode>
   </s:Code>
   <s:Reason>
    <s:Text xml:lang="de">
     9010: Authentisierung/Zertifikat ist abgelaufen
    </s:Text>
   </s:Reason>
   <s:Detail></s:Detail>
  </s:Fault>
 </s:Body>
```

```
</s:Envelope>
```

Die Mitteilung von Fehlern bei Verarbeitung innerhalb der Methoden des XTA-WS wird mit dem Element <Fault> realisiert, welches vom SOAP-Standard für Fehlerbehandlung vorgesehen ist. Es ist vom Webservice-Client auszuwerten, so dass der Anwender möglichst genau weiß, welcher Fehler aufgetreten ist bzw. was geändert werden muss, um die Nachricht korrigiert übersenden zu können.

- Im Element <Code> wird die Identität des Fehlers genannt gemäß Spezifikation XTA. Es muss vom XTA-WS-Client ausgewertet werden. Im Element <Reason> sind ergänzende Informationen zur genaueren Eingrenzung der Fehlerursache dargestellt.
- Im Element <Reason> wird die Fehlernummer (ErrorCode) und eine sinnvolle textuelle Repräsentation eingetragen. Es muss gefüllt, aber nicht ausgewertet werden. Das xml:lang -Attribut ist optional.

Umsetzungshinweis für die Implementierung von Webservice-Clients: Bei fast allen Exceptions macht ein automatisierter erneuter Aufruf der entsprechenden Methode keinen Sinn. Es ist erst eine Fehlerklärung erforderlich. Eine Ausnahme bildet nur die Exception XTAWSTechnicalProblemException, bei der ein automatisierter erneuter Aufruf sinnvoll sein kann.

Folgende Exceptions können als Reaktion auf den Aufruf einer der Webservice-Methoden auftreten.

4.6.1.1 ParameterIsNotValidException

Diese Exception wird geworfen, wenn ein Parameter nicht korrekt an die Methode übergeben wurde.

4.6.1.2 XTAWSTechnicalProblemException

Diese Exception wird allgemein geworfen, wenn ein technisches Problem im XTA-WS aufgetreten ist. Diese Exception kann z. B. durch ein Problem beim Zugriff auf die interne Datenbank des XTA verursacht worden sein.

4.6.1.3 MessageSchemaViolationException

Diese Exception wird geworfen, wenn eine Nachricht nicht der jeweiligen Schema-Definition entspricht.

4.6.1.4 MessageVirusDetectionException

Diese Exception wird geworfen, wenn schadhafter Code in einem der übergebenen Container ermittelt wurde.

4.6.1.5 InvalidMessageIdException

Diese Exception wird geworfen wenn eine nicht bekannte MessageId übergeben wurde.

4.6.1.6 SyncAsyncException

Diese Exception wird geworfen falls dem XTA-WS übergeben wurde:

- eine Nachricht, die nur für die synchrone Weiterleitung gültig ist, für die asynchrone Weiterleitung bzw.
- eine Nachricht, die nur für die asynchrone Weiterleitung gültig ist, für die synchrone Weiterleitung.

4.6.2 Fehlernummern (ErrorCodes)

Fehlernummern geben zur Art des aufgetretenen Fehlers näheren Aufschluss. Die folgende Tabelle gibt eine Übersicht über die in XTA zu verwendenden Fehlernummern (ErrorCodes) und ordnet sie den Exceptions zu, in deren Kontext sie auftreten können:

ErrorCode	Beschreibung	Exception
9000	Unspezifizierter Fehler, als <Freitext> beschrieben	**Alle Exceptions**
9010	Authentisierung/Zertifikat ist abgelaufen.	**PermissiondeniedException**
9011	Account ist gesperrt.	PermissiondeniedException
9012	Account nicht vorhanden.	PermissiondeniedException
9013	Dienst ist nicht gebucht.	PermissiondeniedException
9014	Authentisierung/Zertifikat passt nicht zur Absenderkennung.	PermissiondeniedException
9020	Keine Parameter vorhanden	**ParameterIsNotValidExeption**
9021	Keine gültige URI	ParameterIsNotValidExeption
9022	Ungültige Parameterkombination	ParameterIsNotValidExeption
9023	Die Nachricht überschreitet die Größenbeschränkung.	ParameterIsNotValidExeption
9024	MessageId ist bereits vergeben.	ParameterIsNotValidExeption
9030	Interner Fehler im XTA-Server	**XTAWSTechnicalProblemException**
9031	Fehler beim externen Verzeichnisdienst	XTAWSTechnicalProblemException
9032	Fehler bei der Zustellung	XTAWSTechnicalProblemException
9050	Nachricht ist nicht schemakonform.	**MessageSchemaViolationException**
9051	Nachricht trägt ein falsches Encoding.	MessageSchemaViolationException
9052	Nachricht verletzt standardspezifisches Wesensprofil.	MessageSchemaViolationException
9060	Es wurde schadhafter Code ermittelt.	**MessageVirusDetectionException**
9070	MessageId für den Account nicht bekannt.	**InvalidMessageIdException**
9080	Der Dienst wird nur asynchron angeboten.	**SyncAsyncException**
9081	Der Dienst wird nur synchron angeboten.	SyncAsyncException

5 Glossar

- **Dialogverfahren**: Wird innerhalb eines Kommunikationsszenarios eine unmittelbare Antwort vom Leser erwartet, z.B. weil der Autor diese im Rahmen eines Beratungsgespräches benötigt, liegt ein Dialogverfahren vor.
- **Fachverfahren**: Fachverfahren sind im Sinne der XTA-Spezifikation die IT-Verfahren, die in Behörden für die Vorgangsbearbeitung der jeweiligen Fachdomäne (z.B. Personenstandswesen, Pass- und Ausweisbehörde) eingesetzt werden.
- **Integrität**: Sicherstellung der Korrektheit (Unversehrtheit) von Daten und der korrekten Funktionsweise von Systemen. Integrität meint, dass die Daten vollständig und unverändert sind.

  In der Informationstechnik wird der Begriff in der Regel aber weiter gefasst und auf "Informationen" angewendet. Der Begriff "Information" wird dabei für "Daten" verwendet, denen je nach Zusammenhang bestimmte Attribute wie z. B. Autor oder Zeitpunkt der Erstellung zugeordnet werden können. Der Verlust der Integrität von Informationen kann daher bedeuten, dass diese unerlaubt verändert, Angaben zum Autor verfälscht oder Zeitangaben zur Erstellung manipuliert wurden. Die Integrität ist Schutzziel der IT-Sicherheit.

  Integritätssicherung auf der Transportstrecke meint nicht die nachweisbare Abgabe einer Willenserklärung, sondern die mathematische Nachprüfbarkeit der Unverfälschtheit der Nachricht.
- **Intervenierbarkeit**: Durch Intervenierbarkeit wird sichergestellt, dass Betroffene bzw. Akteure mit legitimierten Eingriffsbefugnissen auf ein Verfahren mit Personenbezug wirksam eingreifen und bestehende Verfahren ändern können. Die Intervenierbarkeit ist Schutzziel des Datenschutzes.

  Weil es sich bei XTA um eine Infrastruktur handelt, die nur mittelbar in einer Beziehung zum betroffenen Bürger steht, stehen Maßnahmen zur Steuerung der Infrastruktur und Maßnahmen zur Steuerung des unmittelbaren Workflows des Kommunikationsvollzugs im Vordergrund.
- **Maximale Durchlaufzeit**: Zeitspanne, die innerhalb eines Szenarios für den Transport der Daten maximal benötigt werden darf.
- **Nachricht**: Die Nachricht besteht aus dem Transportauftrag mit dem zugehörigen Payload.
- **Nichtverkettbarkeit**: Durch die Nichtverkettbarkeit wird sichergestellt, dass die Datenverarbeitung bzw. der Transport von Nachrichten der Zweckbestimmung des Verfahrens insgesamt folgt. Es soll eine vorsätzliche oder fahrlässige oder funktional-fehlerhafte Datenverarbeitung, die ein Risiko für die Einhaltung der Zweckbindung darstellt, wesentlich erschwert werden.

  Die Nichtverkettbarkeit im Kontext von IT-Infrastrukturen wird insbesondere durch organisatorische – und nicht technische - Maßnahmen erreicht. Die Nichtverkettbarkeit ist Schutzziel des Datenschutzes.
- **Payload**: Der Payload ist der fachliche Inhalt der Nachricht, der vom Autor für den Leser erstellt wird. Er umfasst die Gesamtheit der zu übermittelnden Informationen einschließlich Nachrichtenkopf mit den Angaben zu Absender, Adressat, Thema und Datum. Wenn XÖV-Nachrichten zu übermitteln sind, ist der Payload eine (komplette) XÖV-Nachricht.

Der Payload kann vom Autor für den Leser verschlüsselt werden. Deswegen muss der Sender seine Aufgaben mit ausschließlicher Kenntnis des Transportauftrags erfüllen können.

- **Schutzziele des Datenschutzes**: Die Schutzziele der IT-Sicherheit werden ergänzt durch Schutzziele des Datenschutzes: Intervenierbarkeit, Nichtverkettbarkeit und Transparernz. Das Konzept der Schutzziele des Datenschutzes dient der transparenten und integren Operationalisierung von normativen Anforderungen. Es ist Bestandteil im Kommentar der EU-Kommission zum EU-Entwurf der Datenschutzverordnung - „Albrecht-Papier"; siehe auch http://www.europarl.europa.eu, abgerufen 4.4.2013.
- **SOAP**: Protokoll für Request-Response-Kommunikation von Webservices. Eine SOAP-Message hat eine XML-Struktur, unterteilt in Header und Body. Der Body ist dafür gedacht, den zu transportierenden Content – den Payload – aufzunehmen. Der Header nimmt die Nutzungsdaten auf.
- **TLS**: TLS (Transport Layer Security) ist ein hybridesVerschlüsselungsprotokoll auf der Transportebene zur sicheren Datenübertragung im Internet. Das SSL-Protokoll wird seit der Version 3.0 unter diesem neuen Namen weiterentwickelt und standardisiert.
- **Transparenz**: Es soll erreicht werden, dass alle Beteiligten nachweisen können, dass sie den rechtlichen Anforderungen (gemäß dem anzuwendenden Schutzprofil) genügt haben. Transparenz soll die Prüffähigkeit des gesamten Verfahrens sowie einzelner Komponenten sicherstellen. Die Transparenz ist Schutzziel des Datenschutzes.

  Der Nachweis der Rechtskonformität, bzw. die Prüfbarkeit ist für folgende Bereiche relevant:
  - im Binnenverhältnis Autor-Sender und Leser-Empfänger
  - für die funktionale Aufsicht bzgl. der deutschlandweiten (europaweiten) Sicherstellung von Interoperabilität und Effizienz für den IT-Planungsrat
  - für die Prüfbarkeit von Informationssicherheit und Datenschutz durch die Aufsichtsbehörden.
  - Die Sicherung der Transparenz ist Voraussetzung für ein Qualitätsmanagement durch den Auftraggeber (IT-Planungsrat), zu der auch eine externe Auditierung des Verfahrens zählen kann.
- **Transportauftrag**: Der Transportauftrag enthält alle erforderlichen Angaben, um die fachliche Nachricht gemäß der Intention des Autors zum Empfänger zu transportieren. Über den Transportauftrag wird die Qualität der Protokollierung der beteiligten Systeme unter Angabe des anzuwendenden Schutzprofils gesteuert. Jeder Transportauftrag ist eindeutig identifizierbar. Der Transportauftrag wird durch das Objekt MessageMetaData im XTA-WS repräsentiert.
- **Transportstandard**: Im XTA-Kontext werden hiermit Standards wie OSCI 1.2, OSCI 2.01 oder auch SOAP über HTTPS bezeichnet, also Standards im Gegensatz zu fachlichen (XÖV-) Standards.
- **Transportverfahren**: Als Transportverfahren werden im XTA-Kontext IT-Verfahren bezeichnet, die in der Lage sind, Nachrichten zu senden und zu empfangen oder auch an sonstigen Aspekten der Übermittlung mitzuwirken – unabhängig davon, welcher Fachdomäne der Inhalt der Nachricht angehört.
- **Verfügbarkeit**: Die Verfügbarkeit von Dienstleistungen, Funktionen eines IT-Systems, IT-Anwendungen oder IT-Netzen oder auch von Informationen ist vorhanden, wenn diese von den Anwendern stets wie vorgesehen genutzt werden können. Die Verfügbarkeit ist Schutzziel der IT-Sicherheit.
- **Vertraulichkeit**: Vertraulichkeit ist der Schutz vor unbefugter Preisgabe von Informationen. Vertrauliche Daten und Informationen dürfen ausschließlich Befugten in der zulässigen Weise zugänglich sein. Die Vertraulichkeit ist Schutzziel der IT-Sicherheit.
- **Webservice, Web Service, Web-Service**: Ein Webservice ist ein in der Sprache WSDL definierter Service, der von Remote-Systemen über das SOAP-Protokoll angesprochen werden kann. Er funktioniert als Request / Response von SOAP-Messages.
- **XÖV-Nachricht**: XML-Instanz oder Typ solcher Instanzen gemäß der Spezifikation eines XÖV-Fachstandards. Die XÖV-Standards der Innenverwaltung definieren ihre Nachrichten bestehend aus Nachrichtenkopf (Absender, Empfänger, Datum, Identifizierung dieser Nachricht) und Fachdaten (z.B., wenn es um die Änderung einer Religionszugehörigkeit geht, <Religionszugehörigkeit vor Änderung> und <Religionszugehörigkeit nach Änderung>).

6 Versionshistorie

6.1 Release XTA-WS 2.0 (30.06.2013)

- Überarbeitung des WS mit dem Ziel der OSCI 2 Profilierung
- Authentifizierung und Autorisierung geändert

Einführung von PortTypes (managementPortType, sgBoxPortType, sendPortType)

- Methode IsServiceAvailable in lookupService umbenannt
- Methode getTransportreport überarbeitet
- Methode cancelMessaeg neu aufgenommen
- Methode createMessageId neu aufgenommen
- Methode sendMessage überarbeitet
- Methode sendMessageSync überarbeitet
- Methode getMessageIdList in getStatusList umbenannt
- Methode getNextMessage neu aufgenommen
- Methode getNextStatusList neu aufgenommen
- Methode close neu aufgenommen

Parameter und Returnwerte geändert

- umbenannt in ServiceType
- Präfix entfällt
- neu X509TokenContainer
- umbenannt in ReaderIdentifier
- Nachricht/NachrichtResponse zu GenericContantContainer geändert
- MessageID definition überarbeitet (Aufbau der uri korrigiert)
- Transportreport neue Struktur
- Input/Output Struktur von lookupService geändert
- MessageMetaData Container neu hinzugefügt
- neu MsgBoxStatusListRequestType
- neu MsgBoxRequestType
- neu MsgSelector
- neu MsgStatusList
- neu MsgBoxResponseType

- neu MsgBoxCloseRequestType
- Exceptions von SOAP 1.1 auf SOAP 1.2 umgestellt
- Technische Schnittstelle entfernt
- Exceptions geändert
- Adressierung angepasst (Partyidentifier)
- MessageMetaData notbefore / obsoleteafter neu hinzugefügt
- PropertyType geändert (Property Listen werden nach XÖV-Muster als Codelisten strukturiert)
- MessageMetaData in OSCI 2.01 ausgelagert und mit P23R abgestimmt

Synchrone Schnittstelle für den Leser neu aufgenommen (sendSynchronPortType)

- Methode sendMessageSync neu aufgenommen

6.2 Release XTA-WS 1.1 (18.09.2011)

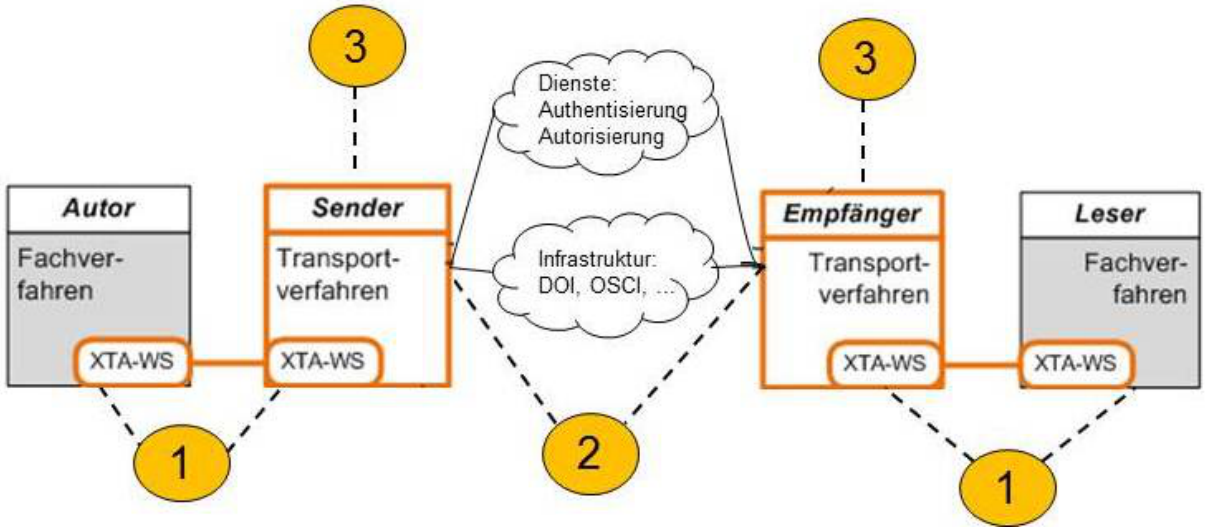
Erstes finales Release der Schnittstellenbeschreibung XTA-WS.

A Modell der Rollen und Verantwortlichkeiten

A.1 Überblick

Die Infrastruktur im XTA-Kontext lässt sich als ein System aus Fach- und Transportverfahren darstellen, die bei den Prozessen der Nachrichtenübermittlung kooperieren. In der nachfolgenden Abbildung wird sie dargestellt als Interaktion von vier Rollen mit zwei Typen von Schnittstellen (1 und 2). Die Transportverfahren (3) sind an beiden Schnittstellen beteiligt.

Abbildung A.1. Infrastruktur der Nachrichtenübermittlung: Kooperation von Fach- und Transportverfahren



Von solchen Gegebenheiten der Implementierung soll in dem vorliegenden Kapitel abstrahiert werden, wenn die Infrastruktur aufgeteilt in Rollen analysiert wird. Die Rollen *Autor* und *Leser* sind dabei der Infrastrukturkomponente *Fachverfahren* zugeordnet, die Rollen Sender und Empfänger der Infrastrukturkomponente *Transportverfahren*.

Dieser Anhang charakterisiert die Rollen und grenzt sie voneinander ab, wie das durch die an XTA beteiligten Parteien abgestimmt worden ist.

Die Definition und Abgrenzung der Rollen geschieht in Form von *Sätzen*. Im Abschnitt A.2.1 auf Seite 89 werden zu jeder der vier Rollen Sätze formuliert, welche entsprechende Festlegungen treffen, also normativ zu verstehen sind.

Die Sätze sind zu den einzelnen Themenbereichen gruppiert, siehe auch Tabelle auf Seite 88. Zu einigen Sätzen sind für ein besseres Verständnis Erläuterungen oder Anmerkungen hinzugefügt.

A.2 Die Rollen

Nr.	Thema	Stichworte	Seite
Autor			
A 1	Aufgabe		Seite 89
A 2	Zuständigkeitsprüfung / Vorbereitung der Kommunikation mit dem Leser		Seite 89
A 3	Signatur		Seite 89
A 4	Verschlüsselung		Seite 90
A 5	Kommunikationszenario		Seite 90
A 6	Service Qualität		Seite 90
A 7	Eindeutige Identifizierung des Transportauftrages		Seite 90
A 8	Überwachung der Übermittlung		Seite 90
A 9	Aufbewahrung		Seite 91
A 10	Vertraglicher und rechtlicher Rahmen		Seite 91
Sender			
B 1	Aufgabe		Seite 91
B 2	Prüfung Identität Autor		Seite 91
B 3	Kommunikation mit dem Leser		Seite 92
B 4	Transportkanal		Seite 92
B 5	Verschlüsselung und Signatur		Seite 92
B 6	Zustellung		Seite 92
B 7	Protokollierung		Seite 92
B 8	Daten löschen		Seite 92
B 9	Service Qualität		Seite 92
Empfänger			
C 1	Aufgabe		Seite 92
C 2	Prüfung Identität Leser		Seite 93
C 3	Prüfung Berechtigung Autor		Seite 93
C 4	Entschlüsselung und Signaturprüfung		Seite 93
C 5	Zustellung		Seite 93
C 6	Protokollierung		Seite 94
C 7	Daten löschen		Seite 94
C 8	Service Qualität		Seite 94
Leser			
D 1	Aufgabe		Seite 94

Nr.	Thema	Stichworte	Seite
D 2	Zuständigkeitsprüfung mit dem Autor		Seite 94
D 3	Signatur		Seite 94
D 4	Entschlüsselung		Seite 95
D 5	Kommunikationszenario		Seite 95
D 6	Service Qualität		Seite 95
D 7	Eindeutige Identifizierung des Transportauftrages		Seite 95
D 8	Überwachung Empfang		Seite 95
D 9	Aufbewahrung		Seite 95
D 10	Vertragl. und rechtlicher Rahmen		Seite 95

A.2.1 Der Autor

A.2.1.1 Aufgabe des Autors

A 1.1 Der Autor ist fachlich zuständig, d.h. er ist für den Inhalt der zu transportierenden Nachricht verantwortlich.

A 1.2 Der Autor erstellt den Inhalt der zu transportierenden Nachricht. Er erstellt die zu transportierende Nachricht gemäß den Regeln des zu Grunde liegenden Standards (z.B. OSCI-XMeld) in einer bestimmten Version.

Amerkung:

- *Der vollständige Inhalt der vom Autor erstellten Nachricht ist für den Leser relevant. Und alles, was für den Leser relevant ist, sollte in der Nachricht enthalten sein. Dies betrifft auch die Informationen, die im Nachrichtenkopf einer XÖV-Nachricht (vergleichbar dem Inhalt eines Briefkopfes), enthalten sind, wie z.B. der AGS von Absender und Empfänger sowie die Nachrichten-Identifizierung.*
- *Es ist nicht ausgeschlossen, dass einzelne Informationen aus dem Briefkopf auch für den Sender relevant sind. Dies kann der Fall sein, wenn der Sender die Informationen benötigt, um die technischen Adressdaten des Lesers / Empfängers zu ermitteln*

A 1.3 Der Autor ist verantwortlich dafür, dass die Nachricht entsprechend spezifikationskonform ist. Das schließt ein, dass die Nachricht valide bezüglich des für den Standard (in der entsprechenden Version) gültigen Schemas ist.

A.2.1.2 Zuständigkeitsprüfung des Lesers durch den Autor

A 2.1 Der Autor ist für die fachliche Adressierung des Lesers zuständig.

A 2.2 Der Autor kann prüfen, ob der Leser in einem bestimmten fachlichen Kontext elektronisch erreichbar (z. B. DVDV, SAFE) ist. (Hiermit ist nicht die Prüfung gemeint, ob der Leser aktuell verfügbar ist.) Hierbei ist der Sender einbezogen bzw. er stellt eine entsprechende Funktionalität zur Verfügung. Diese Prüfung erfolgt durch qualitätsgesicherte Verzeichnisse der öffentlichen Verwaltung.

A 2.3 Der Autor muss benötigte Attribute für die elektronische Kommunkation mit dem Leser abrufen können, sofern dies im fachlichen Kontext notwendig ist. Hierbei ist der Sender einbezogen bzw. er stellt eine entsprechende Funktionalität zur Verfügung.

A.2.1.3 Signatur

A 3.1 Der Autor kann die zu transportierende Nachricht oder Teile von dieser signieren.

Anmerkungen:

- *Die XhD-Spezifikation ist ein Beispiel, in der Teile einer Nachricht signiert werden.*

A 3.2 Der Autor ist zuständig für die Signatur der Nachricht, nicht der Sender. Ein Sender kann im Auftrag des Autors diese Aufgabe wahrnehmen. Der Autor muss immer über die Signatur identifizierbar bleiben.

A.2.1.4 Verschlüsselung

A 4.1 Der Autor kann die zu transportierende Nachricht oder Teile von dieser verschlüsseln.

A 4.2 Der Autor ist zuständig für die Verschlüsselung der Nachricht, nicht der Sender. Ein Sender kann im Auftrag des Autors diese Aufgabe wahrnehmen.

A.2.1.5 Kommunikationszenario

A 5.1 Die Wahl des Kommunikationsszenarios (synchron, asynchron) ist durch rechtliche Normen und Regeln in einer fachlichen Spezifikation vorgegeben.

A 5.2 Der Autor erteilt den Transportauftrag. Dabei übergibt er die zu transportierende Nachricht an den Sender.

A 5.3 Der Autor muss sicherstellen, dass er den korrekten Sender adressiert.

A.2.1.6 Service Qualität

A 6.1 Der Autor wählt eine bestimmte Service Qualität aus den vom Sender unter Berücksichtigung der rechtlichen und fachlichen Vorgaben angebotenen Optionen aus.

Anmerkungen:

- *Die benötigte Servicequalität muss durch den Autor mit dem Sender vertraglich vereinbart sein.*
- *Beispiele für Service Qualitäten sind "Zeit bis Abschluss Geschäftsprozess" und "Transportquittung(en) angefordert".*
- *Ein weiterer Ausgangspunkt für die Definition von Service Qualitäten sind die fachlich festgelegten Schutzbedarfe (aus Vertraulichkeit, Integrität, Verfügbarkeit) und den daraus abgeleiteten Anforderungen Transparenz (Überprüfbarkeit), Intervenierbarkeit (Changemanagement, Nichtverkettbarkeit).*

A.2.1.7 Eindeutige Identifizierung des Transportauftrages

A 7.1 Der Autor ist verantwortlich für die Erzeugung einer Identifizierung des Transportauftrags (MessageID).

Anmerkung:

- *Diese eindeutige Identifizierung eines Transportauftrags meint nicht die eindeutige Identifizierung einer Nachricht. Wenn eine zu transportierende Nachricht also mehrfach versendet werden muss, erhält sie mit jedem Versand eine neue eindeutige Transport-Identifizierung.*
- *Diese eindeutige identifizierung soll bis zum Leser durchgereicht werden.*

A 7.2 Der Autor vergibt eine MessageID bei der Beauftragung des Transports oder lässt alternativ diese ID vom Sender erzeugen, um sie sich anschließend mitteilen zu lassen.

A 7.3 Die MessageID ist im Transportauftrag enthalten. Die MessageID soll eine durchgehende ID für den gesamten Transportauftrag, für die vollständige Zustellungskette sein.

A.2.1.8 Überwachung der Übermittlung der Nachricht

A 8.1 Der Autor ist für die Überwachung der Übermittlung und die Einhaltung der (rechtlich- organisatorischen Vorgaben für) Übermittlungfristen der Nachricht an den Empfänger bzw. Leser zuständig.

A 8.2 Der Autor ist für das Zurückziehen eines offenen Transportauftrags zuständig.

Anmerkungen:

- *Ein Transportauftrag gilt als "offen", wenn der Sender den Auftrag noch nicht an den Empfänger weitergegeben hat. Der Transportauftrag befindet sich also noch beim Sender.*
- *Für das Zurückziehen stellt der Sender eine Funktionalität zur Verfügung.*

A.2.1.9 Aufbewahrung

A 9.1 Der Autor ist für die Aufbewahrung der versandten Nachrichten und der relevanten Transportinformation zuständig und dafür, dass fristgerecht gelöscht wird. Für die Aufbewahrung kann er sich eines Dienstleisters bedienen.

A 9.2 Der Autor legt fest, wie lange beim Sender die Nachrichten und die Protokolle der Nutzungsdaten gespeichert werden. Die Löschfristen werden vereinbart.

Anmerkungen:

- *Die Rahmenbedingungen hierfür werden durch rechtliche und fachliche Vorgaben gesetzt.*

A.2.1.10 Vertraglicher und rechtlicher Rahmen

A 10.1 Das rechtliche Verhältnis zwischen Autor und Sender muss geklärt sein.

A.2.2 Der Sender

A.2.2.1 Aufgabe des Senders

B 1.1 Der Sender ist gemäß Transportauftrag des Autors für die Abwicklung des Transports zuständig. Der Sender unterhält dafür die Infrastruktur und gibt dem Autor einen entsprechenden Zugang.

Anmerkungen:

- *Der Autor entscheidet über Eigenschaften bzgl. der Servicequalität. Hierbei müssen z.B. Vorgaben des Landes berücksichtigt werden. Der Sender muss nachweisen, dass er diese gewährleisten kann.*
- *Der Sender trägt die Verantwortung bei Entscheidungen, die das Transportprotokoll (HTTPS, OSCI-Transport,...) betreffen. Die Anforderungen werden im Rahmen der Service-Qualität definiert.*

A.2.2.2 Prüfung der Identität des Autors durch den Sender

B 2.1 Der Sender ist für die Authentifizierung des Autors zuständig, d. h. er prüft die Identität des Autors.

Anmerkung:

- *Der Sender überprüft, ob ihm die Authentisierungsinformationen des Autors bekannt sind.*

B 2.2 Der Sender ist verpflichtet, die Angaben der Authentifizierung auf Konsistenz mit den Absenderangaben des Transportauftrages zu prüfen.

Anmerkung:

- *Durch diese Prüfung wird geklärt, ob die authentisierte Behörde in diesem Fachkontext mit dieser Behördenidentität (=z.B. ags:12343123 für Oldenburg im Meldewesen) auftreten darf.*

B 2.3 Abgrenzung: Der Sender ist nicht dafür zuständig, die fachliche Zuständigkeit des Autors für den Inhalt der Nachricht zu prüfen. Dies geschieht durch den Leser.

A.2.2.3 Kommunikation des Senders mit dem Leser

B 3.1 Der Sender prüft, ob für den Leser ein Zugang eröffnet ist. Der Sender ermittelt die technische Adresse des Lesers anhand dessen fachlicher Adresse. Er verwendet hierfür ein Verzeichnis wie DVDV oder S.A.F.E.

B 3.2 Der Sender stellt, falls nötig, dem Autor die technischen Attribute des Lesers zur Verfügung.

A.2.2.4 Transportkanal

B 4.1 Der Sender entscheidet über den zu nutzenden Transportkanal (technische Alternativen im Rahmen seiner Vereinbarung mit dem Autor).

Anmerkung: Der Sender entscheidet zwischen technischen Alternativen (z.B. landesinterne Infrastruktur, OSCI-DVDV-Infrastuktur), die in der Vereinbarung mit dem Autor als mögliche Alternativen für eine bestimmte Qualität (Quittungen, Zustellfristen, ...) benannt sind. Technisch erfolgt diese Vereinbarung über „Wesensprofile“, siehe Abschnitt Abschnitt 3.4, „Wesensprofile“.

A.2.2.5 Verschlüsselung und Signatur

B 5.1 Der Sender bringt je nach Policy für den jeweiligen Transportkanal ggf. die Transportsignatur an.

B 5.2 Der Sender verschlüsselt die zu transportierende Nachricht je nach Policy für den jeweiligen Transportkanal ggf. für den Empfänger.

A.2.2.6 Zustellung durch den Sender

B 6.1 Der Sender führt den Transport der übergebenen Nachricht zum Empfänger durch.

A.2.2.7 Protokollierung durch den Sender

B 7.1 Der Sender erstellt Transportprotokolle und stellt sie dem Autor zur Verfügung. Der Sender hält die Transportprotokolle zu Nachweiszwecken vor.

B 7.2 Der Sender trägt Hindernisse für die Auftragserfüllung in die Protokolle ein.

A.2.2.8 Daten löschen

B 8.1 Der Sender löscht alle zum Transportauftrag gehörenden Daten, sobald sie nicht mehr benötigt werden.

A.2.2.9 Service Qualität

B 9.1 Der Sender muss den Transportauftrag entsprechend der Vorgaben des Autors behandeln.

A.2.2.10 Eindeutige Identifizierung des Transportauftrags

B 10.1 Der Sender ist verantwortlich dafür, die Identifizierung des Transportauftrags (MessageID) eindeutig zu erzeugen.

A.2.3 Der Empfänger

A.2.3.1 Aufgabe des Empfängers

C 1.1 Der Empfänger ist vom Leser mit der Entgegennahme von Nachrichten beauftragt.

Anmerkungen:

- *Ein Intermediär ist Bestandteil der Empfänger-Infrastruktur.*

C 1.2 Der Empfänger unterhält für die Entgegennahme von Nachrichten die Infrastruktur.

C 1.3 Der Empfänger stellt dem Leser die entgegengenommenen Nachrichten zur Verfügung.

A.2.3.2 Prüfung Identität Leser

C 2.1 Der Empfänger ist für die Authentifizierung des Lesers zuständig, d. h. er hat die Identität des Lesers zu prüfen.

Anmerkung:

- *Die Prüfung dient der Zugriffskontrolle im Kontext der Autorisierung nach Absprache mit dem Leser.*

C 2.2 Der Empfänger ist für die Prüfung zuständig, ob die Identität des Lesers (Authentisierung gegenüber dem Empfänger) konsistent ist mit der Identität des Lesers für die Fachkommunikation im Rahmen des Transportauftrags.

Anmerkungen:

- *Falls der Empfänger die Nachricht keinem Leser zuordnen kann, schickt er dem Sender eine Fehlermeldung (ggf. eine administrativ-technische RTS-Nachricht). Entsprechend kann der Erfolgsfall durch eine Quittung (auf Transportebene) bestätigt werden.*
- *Die Prüfung ist ein Aspekt der Service-Qualität des Empfängers.*

C 2.3 Abgrenzung: Der Empfänger ist nicht dafür zuständig, die fachliche Zuständigkeit des Lesers für den Inhalt der Nachricht zu prüfen. (Dies geschieht durch den Leser).

A.2.3.3 Prüfung der Berechtigung des Autors durch den Empfänger

C 3.1 Der Empfänger prüft in Abhängigkeit vom Nutzungsszenrario, ob der Autor berechtigt ist, diese Nachricht zu senden.

C 3.2 Abgrenzung: Der Empfänger ist nicht zuständig, die fachliche Zuständigkeit des Autors für den Inhalt der Nachricht zu prüfen. (Das prüft der Leser).

A.2.3.4 Entschlüsselung und Signaturprüfung

C 4.1 Der Empfänger prüft je nach Policy für den jeweiligen Transportkanal ggf. die Transportsignatur.

C 4.2 Der Empfänger entschlüsselt je nach Policy für den jeweiligen Transportkanal ggf. die Transportverschlüsselung für den Leser.

A.2.3.5 Zustellung durch den Empfänger

C 5.1 Der Empfänger nimmt das transportierte Nachrichtendokument entgegen und stellt es dem Leser zur Verfügung.

Anmerkungen:

- *Die Zustellung kann asynchron oder synchron erfolgen. Synchrone Kommunikation ist dadurch charakterisiert, dass der Prozess blockiert bis der Leser die Reaktion geliefert hat.*

C 5.2 Der Empfänger stellt die Empfangsquittung für den Sender / Autor zur Verfügung, falls diese Service-Qualität vom Autor angefordert worden ist.

C 5.3 Fehlerbehandlungen und Ausnahmeregelungen erfolgen entsprechend der fachlichen Spezifikationen und der vertraglichen Vereinbarungen. Falls der Leser nicht ermittelt werden kann, erfolgt keine Zustellung. Dies wird protokolliert und der Autor erhält eine entsprechende Benachrichtigung.

A.2.3.6 Protokollierung durch den Empfänger

C 6.1 Der Empfänger erstellt Transportprotokolle und stellt diese dem Leser zur Verfügung. Er hält diese entsprechend der fachlichen und vertraglichen Vorgaben vor.

C 6.2 Der Empfänger trägt Hindernisse für die Auftragserfüllung in die Protokolle ein.

A.2.3.7 Daten löschen

C 7.1 Der Empfänger löscht alle zum Transportauftrag gehörenden Daten, sobald sie nicht mehr benötigt werden.

Anmerkung:

- *Im Rahmen der Definition der Schutzbedarfe wird festgelegt, wann wie sicher gelöscht werden muss.*

A.2.3.8 Service Qualität

C 8.1 Der Empfänger erfüllt die Verpflichtungen, die mit dem Leser vereinbart sind.

A.2.4 Der Leser

A.2.4.1 Aufgabe des Lesers

D 1.1 Der Leser ist fachlich zuständig: Er ist für die Auswertung des Inhalts der empfangenen Nachricht verantwortlich.

D 1.2 Der Leser prüft, ob die transportierte Nachricht spezifikationskonform ist.

D 1.3 Der Leser prüft, ob die Nachricht valide bezüglich des für den Standard gültigen Schemas ist.

A.2.4.2 Zuständigkeitsprüfung des Autors durch den Leser

D 2.1 Der Leser prüft seine eigene Zuständigkeit.

Anmerkungen:

- *Hintergrund für diese Prüfung ist der Wunsch, eine falsche Adressierung auszuschließen.*
- *Der Umgang mit falsch adressierten Nachrichten ist gesondert geregelt.*

D 2.2 Der Leser prüft Zuständigkeit und Berechtigung des Autors.

Anmerkungen:

- *Die Prüfung erfolgt, weil der Leser aus dem Ergebnis ableiten kann, wie er die erhaltenen Informationen verarbeitet, ob z.B. ein Register fortgeschrieben werden muss.*
- *Diese Prüfung kann an den Empfänger delegiert werden.*

D 2.3 Der Leser kann ggf. benötigte Attribute über den Empfänger für die Überprüfung des Autors abrufen.

A.2.4.3 Signatur

D 3.1 Der Leser ist zuständig für die Prüfung der (Autor-)Signatur der Nachricht.

D 3.2 Der Leser bewertet das Ergebnis der Prüfung der (Autor-)Signatur. Dies geschieht auch dann, wenn ein Dritter die (technische) Prüfung der Signatur durchgeführt hat.

Anmerkungen:

- *Die technische Prüfung kann grundsätzlich deligiert werden.*

- *Der Leser benötigt für seine Prüfung eine Liste der Signatur-Zertifikate, die ihm übermittelt werden müssen.*

A.2.4.4 Entschlüsselung

D 4.1 Der Leser ist für die Entschlüsselung der erhaltenen Nachricht zuständig.

Anmerkung:

- *Die Entschlüsselung kann grundsätzlich deligiert werden.*

A.2.4.5 Kommunikationszenario

D 5.1 Asynchrones Kommunikationszenario: Der Leser ist verpflichtet, Nachrichten und Transportinformation vom Empfänger abzurufen oder entgegenzunehmen.

Anmerkung:

- *Eine "Entgegennahme" setzt voraus, dass eine direkte Zustellung ("Push") eingerichtet ist.*

D 5.2 Synchrones Kommunikationsszenario: Der Leser bedient die Anfrage des Autors unmittelbar.

A.2.4.6 Service Qualität

D 6.1 Der Leser reagiert gemäß der vom Autor ausgewählten Service Qualität.

A.2.4.7 Eindeutige Identifizierung des Transportauftrages

D 7.1 Der Leser verwendet eine MessageID, um Informationen aus der Transport-Historie auf die Nachricht zu beziehen.

A.2.4.8 Überwachung des Empfangs durch den Leser

D 8.1 Der Leser ist für die Auswertung der Transportinformationen verantwortlich.

D 8.2 Der Leser überprüft die Identität des Empfängers.

D 8.3 Der Leser ist dafür verantwortlich, im Rahmen seiner vertraglichen und rechtlichen Verpflichtungen für den Autor erreichbar zu sein. Dies impliziert die technische Erreichbarkeit für den Empfänger bei synchronen Transportszenarien.

A.2.4.9 Aufbewahrung

D 9.1 Der Leser ist für die Aufbewahrung der empfangenen Nachrichten und der relevanten Transportinformation zuständig und dafür, dass fristgerecht gelöscht wird. Für die Aufbewahrung kann er sich eines Dienstleisters bedienen. Die Löschfristen sind vertraglich festzulegen.

D 9.2 Der Leser legt im Rahmen der fachlichen, vertraglichen und rechtlichen Vorgaben fest, wie lange beim Empfänger die Nachrichten und die Protokolle der Nutzungsdaten gespeichert werden.

A.2.4.10 Vertraglicher und rechtlicher Rahmen

D 10.1 Das rechtliche Verhältnis zwischen Leser und Empfänger muss geklärt sein.

B XTA-Profile und XTA-Konformität

Die hier definierten prototypischen Schutzprofile I bis IV werden in der Erprobungsphase auf ihre Vollständigkeit und Korrektheit überprüft. Die Prozesse zur Ergänzung der Menge der Schutzprofile oder Änderungen an ihnen sind noch zu definieren und abzustimmen.

B.1 Schutzprofile

B.1.1 Schutzprofil I: "Normal"

Das Schutzprofil I (“normal”) bietet einen “normal” abgesicherten Nachrichtenaustausch auf Basis des IT-Grundschutzes. Es dürfen nur personenbezogene Daten im geringen Umfang und mit geringem Schutzwert (z.B. öffentliche Adressen) transportiert werden.

- **Vertraulichkeit**: Die Vertraulichkeit wird insbesondere durch Verschlüsselung hergestellt. Dies erfolgt durch Nutzung sicherer Netze wie insbesondere des DOI, bzw. der Verbindungsnetze, der Landesnetze oder durch Leitungsverschlüsselung (Hardware / SSL) oder durch eine Transportverschlüsselung, wie sie für XTA durch OSCI vorgesehen ist.
- **Integrität**: Es müssen keine besonderen Maßnahmen, die über die Anforderungen des BSI – Grundschutz hinausgehen, getroffen werden.
- **Verfügbarkeit**: Maximale Laufzeit einer Nachricht von bis zu 3 Werktagen zwischen Sender und Empfänger
- **Transparenz**: Transparenz wird durch Dokumentation und System-Protokollierung erreicht. Die Protokollierung erfolgt durch die beteiligten Instanzen. Sie muss dazu geeignet sein, die Ordnungsmäßigkeit des rechtlichen Binnenverhältnisses und auch die des gesamten Systems festzustellen.

  Wesentlich ist die Dokumentation der Systeme, die durch Protokollierung erfolgt. Dies betrifft insbesondere die Schnittstellen der IT-Systeme (Fachverfahren, IT-Infrastruktur / Clearingstellen) sowie die zugehörigen Rechte- und Rollenkonzepte zur technischen und organisatorischen Verwaltung der Systeme. Jede datenverarbeitende Stelle muss eine Dokumentation der verwendeten IT und der lokalen Infrastruktur sowie eine Dokumentation der Sicherheitsmaßnahmen vorlegen können.

  Bei Dienstleistern, bei denen mehrere Verwaltungen gehostet werden, wie z.B. Clearingstellen, muss insbesondere auf eine klare Mandantentrennung bei Datenverarbeitung und für Verfahren geachtet werden.
- Über den Transportauftrag wird die Qualität der Protokollierung der beteiligten Systeme entsprechend Schutzprofil gesteuert.

  Optionen:
  - keine Protokollierung

  - systemseitige „Standard“-Protokollierung
  - kryptisch gesicherte systemseitige Protokollierung
  - kryptisch gesicherte Protokollierung auf dediziertem Protokollserver
  - Feld für die Anforderung einer Quittierung, die der Autor (bzw. Sender) mit dem Eingang der Nachricht beim Leser (bzw. Empfänger) erhalten soll
- Jedes beteiligte System führt ein Protokoll, das mindestens folgende Einträge enthält:
  - Zeitstempel
  - AutorID
  - LeserID
  - type-of-business
  - Transportauftrag (ohne Payload)
- Über Zeitstempel und der MessageID auf dem Transportauftrag können die Aktionen der Beteiligten nachvollzogen werden:
  - Nachweis des Senders gegenüber dem Autor, dass er nach den Vorgaben des Transportauftrages korrekt an den Empfänger ausgeliefert hat.
  - Nachweis des Empfängers gegenüber dem Leser, dass er die Nachricht korrekt entgegengenommen und weitergeleitet hat.
- Die Protokollierung erfolgt für alle beteiligten Akteure / Instanzen. Für Operationen, die das Senden und Empfangen oder Aspekte des Schutzniveaus betreffen, wird jeweils protokolliert:
  - Welche Operation mit Angabe der verwendeten Eigenschaft wurde durchgeführt?
  - Wer sind die beteiligten Kommunikationspartner?
  - Wann wurde diese Operation durchgeführt?
  - Welche Entität hat die Operation ausgelöst? Bei normalem Schutzniveau kann dies durch eine Selbstauskunft des Systems erfolgen.
  - Dokumentation von Senden und Empfangen
  - Überprüfbarkeit des Schutzniveaus durch Übernahme des Transportauftrags in das Protokoll
  - Bei der Angabe der Zeitstempel kann die jeweilige (unsignierte) Systemzeit verwendet werden.
- Hierbei ist zu beachten, dass auch für Protokollierungen der Grundsatz der Zweckbindung und der Datensparsamkeit gilt.

Intervenierbarkeit: Nachvollziehbar wird die Intervenierbarkeit durch ein geregeltes Change-Management, das Teil des Betriebskonzepts ist.

- Bei XÖV-Verfahren erfolgen die Ressourcenplanung und ein Monitoring sowohl durch die ausführenden Behörden als auch durch KoSIT bzw. IT-Planungsrat.
- Bei normalem Schutzbedarf müssen im Transportauftrag keine zusätzlichen Anforderungen umgesetzt werden.

  Es muss allerdings organisatorisch sichergestellt sein, dass der Autor die Kontaktadresse „seines“ Senders und Lesers ermitteln kann. Dies gilt auch entsprechend für den Leser, der die Kontaktadressen des Empfängers sowie des Autors ermitteln können muss, so dass auch von der Maschine-Maschine-Kommunikation abweichende Optionen zur Kontaktaufnahme ermöglicht werden. Es wird daher empfohlen, in den Adressierungsdiensten oder in behördlichen Zuständigkeitsfindern einen „Single Points of Contact“ zu hinterlegen, z.B. durch Angabe einer Telefonnummer.

Nichtverkettbarkeit: Weil die Zweckbindung einer personenbezogenen Datenverarbeitung rechtlich immer sichergestellt werden muss, wird der Schutzbedarf nicht differenziert. Es gilt in allen Schutzprofilen:

- Das **Gesamtsystem** (Fachverfahren, Kommunikationsverfahren, IT-Infrastruktur) darf grundsätzlich nur über die Daten, Schnittstellen, IT-Systeme und Prozesse verfügen, die zweckgemäß sind. Die zu nutzenden Infrastrukturprofile entscheiden hierbei über die einsetzbaren IT-Systeme.
- Die Umsetzung wird durch Dokumentation und Protokollierung nachgewiesen, was durch das Schutzziel Transparenz erfolgt.
- Die Sicherstellung der ausschließlich zweckgebundenen Verkettbarkeit setzt die Sicherstellung von Vertraulichkeit (es erfolgt keine unbefugte Kenntnisnahme) und von Integrität (kein Bruch der Prozesse, korrekte Schutzmaßnahmen, korrekte Adressierungen) voraus. Vorsätzliche Abweichungen werden mit Verweis auf rechtliche Ermächtigungsregelungen begründet und werden operativ ausgewiesen.
- Die Umsetzung der Nichtverkettbarkeit im **Transport-Auftrag** erfolgt durch eine Selbsterklärung des Fachverfahrens, die aussagt, welcher Payload in welchem Transport- und Verfahrenskontext behandelt wird. zusätzlich muss sichergestellt sein, dass die Adressierung des Empfängers korrekt erfolgt. Die Selbsterklärung kann auch durch entsprechende Angaben im Wesensprofil ersetzt werden.

Weitere Merkmale:

- **Maximale Durchlaufzeit**: Dies ist die Zeitspanne, die innerhalb eines Szenarios für den Transport der Daten maximal benötigt werden darf. Ist die vorgegebene maximale Durchlaufzeit zur Sicherstellung der rechtlichen Regelungen zu gering, ist gleichwohl sicherzustellen, dass die Umsetzung der rechtlichen Regelung erfolgen kann und ist eine Lösung zu finden.
- **Dialogverfahren**: Wenn das jeweilige Wesensprofil ein Dialogverfahren fordert, muss der Leser den XTA-Service zur synchronen Kommunikation („SendMsgSync“) implementiert haben. Wenn der Autor eine direkte Antwort erwartet / erhofft, er aber auch darauf eigerichtet ist, dass diese ggf. nicht erfolgt, kann das Merkmal optional verwendet werden.

B.1.2 Schutzprofil II: "hoch"

Das Schutzprofil “hoch” baut auf einen “normalen” Schutzbedarf auf. Mit ihm wird eine Umsetzung des “hohen” Schutzbedarfs bzgl. Vertraulichkeit und Integrität erreicht. Dieses Schutzprofil ist sowohl für hoch schutzwürdige Daten und auch allgemein für persönliche Daten geeignet.

- **Vertraulichkeit**: Aufgrund des hohen Schutzbedarfs ist auf eine strikte Zweckbindung bei der Möglichkeit zur Einsichtnahme in die Daten zu achten.
  - Auch bei Nutzung von Verbindungs- oder Landesnetzen ist der Payload zwischen Autor und Leser zusätzlich zu verschlüsseln. Der XTA – WS stellt Autor und Leser hierfür Hilfen wie z.B. Anbindung an entsprechende Verzeichnisdienste (DVDV, S.A.F.E.) zur Verfügung.
  - Kann die Vertraulichkeit nicht durch Verschlüsselung des Payloads zwischen Autor-Sender und / oder Empfänger-Leser sichergestellt werden, ist dies durch andere Maßnahmen, die im Betriebskonzept beschrieben werden, zu gewährleisten. (Beispiel: geteilte Admin-Passwörter, 4-Augenprinzip, erweiterte Protokollsicherung).
  - Fallen die Rollen Autor-Sender oder Empfänger-Leser zusammen, ist dies im Sicherheitskonzept zu vermerken.
- **Integrität**: Die Sicherstellung der Integrität erfolgt i.d.R. durch eine Signatur des Autors und / oder durch die Verschlüsselung des Payloads.
  - Erfolgt keine Verschlüsselung des Payload zwischen Autor-Sender und / oder Empfänger-Leser, ist die Integritätssicherung durch andere Maßnahmen zu kompensieren, siehe Anmerkungen zur Vertraulichkeit.
  - Wenn der Payload durch den Autor signiert wird, geschieht dies auf der fachlichen Ebene. Es betrifft damit nicht den Transport.

- **Verfügbarkeit**: Die Durchleitung der Nachrichten erfolgt zu 98.5% mit einer maximalen Laufzeit einer Nachricht bis zu 3 Werktage 24/7 zwischen Sender und Empfänger.
- **Transparenz**: Die Transparenz wird wie beschrieben beim Schutzprofil I („normal“) umgesetzt. Verschärfend ist zu beachten, dass die zu liefernden Zeitstempel verlässlich und nicht veränderbar sein müssen. Ebenso ist notwendig, dass für die Dokumentation der Entitätsbezeichner nicht eine Selbstauskunft ausreicht, sondern ein signierter Systembezeichner eingesetzt werden muss. Die gesicherte, signierte System-Protokollierung soll möglichst unter Einsatz eines eigenen Protokollierungsservers erreicht werden.
- **Intervenierbarkeit**: Nachvollziehbar wird die Intervenierbarkeit durch ein geregeltes Change-Management, das Teil des Betriebskonzepts ist.
  - Bei XÖV-Verfahren erfolgen die Ressourcenplanung und ein Monitoring sowohl durch die ausführenden Behörden als auch durch KoSIT bzw. IT-Planungsrat.
  - Klärung,wie Sender bzw. Empfänger bei (standardisierten) Vorfällen, in denen z.B. Autor oder Leser eine Unregelmäßigkeit feststellen, agieren müssen.
  - Klärung, wie Sender und Empfänger auf (standardisierte) Vorfälle in ihrer Zusammenarbeit agieren müssen.
- **Nichtverkettbarkeit**: siehe Anforderungen zu Schutzprofil I („normal“).

Weitere Merkmale:

- **Maximale Durchlaufzeit**: Ist die vorgegebene maximale Durchlaufzeit zur Sicherstellung der rechtlichen Regelungen zu gering, ist gleichwohl sicherzustellen, dass die Umsetzung der rechtlichen Regelung erfolgen kann und ist eine Lösung zu finden.
- **Dialogverfahren**: Um im Kommunikationsprozess grundsätzlich eine direkte Antwort auf eine Anfrage geben zu können, muss der Leser den XTA-Service zur synchronen Kommunikation („SendMsgSync“) implementiert haben. Das Merkmal kann optinal gesetzt werden, wenn es dem Autor zumutbar ist, dass er keine direkte Antwort erhält.
- **Erweiterte Protokollregeln**: Die erhöhten Anforderungen bezüglich der Datenschutzziele sind primär vom Fachverfahren umzusetzen. Ggf. ist aufgrund erhöhter Anforderungen an Vertraulichkeit und Datenschutz eine erweiterte Protokollsicherung notwendig. Diese werden durch die Fachverfahrensverantwortlichen festgelegt.

B.1.3 Schutzprofil III: "normal & schnell"

Das Transportprofil “normal & schnell” ist für den “normalen” Datenaustausch mit hohen Anforderungen an die Verfügbarkeit – insbesondere bei Auskunfts- oder Dialogverfahren – geeignet.

- **Verschlüsselung**: Siehe Anforderungen zum Schutzprofil I („normal“).
- **Integrität**: Siehe Anforderungen zum Schutzprofil I („normal“).
- **Verfügbarkeit**: In 98,5% der Datenübermittlungen werden Nachrichten zwischen Sender und Empfänger in durchschnittlich 7,5 Sekunden übermittelt.
- **Transparenz**: Siehe Anforderungen zum Schutzprofil I („normal“).
- **Intervenierbarkeit**: Siehe Anforderungen zum Schutzprofil I („normal“).
- **Nicht-Verkettbarkeit**: Siehe Anforderungen zum Schutzprofil I („normal“).

Weitere Merkmale:

- **Maximale Durchlaufzeit**: Ist die vorgegebene maximale Durchlaufzeit zur Sicherstellung der rechtlichen Regelungen zu gering, ist gleichwohl sicherzustellen, dass die Umsetzung der rechtlichen Regelung erfolgen kann und ist eine Lösung zu finden.
- **Dialogverfahren**: Um im Kommunikationsprozess grundsätzlich eine direkte Antwort auf eine Anfrage geben zu können, muss der Leser den XTA-Service zur synchronen Kommunikation („SendMsgSync“)

implementiert haben. Das Merkmal kann optinal gesetzt werden, wenn es dem Autor zumutbar ist, dass er keine direkte Antwort erhält.

B.1.4 Schutzprofil IV: "hoch & schnell"

Das Schutzprofil "hoch & schnell" ist dazu geeignet, um Daten mit hohem Schutzbedarf schnell und / oder mit hoher Verfügbarkeit übermitteln zu können:

- **Vertraulichkeit**: Aufgrund des hohen Schutzbedarfs ist auf eine strikte Zweckbindung bei der Möglichkeit zur Einsichtnahme in die Daten zu achten.
  - Auch bei Nutzung von Verbindungs- oder Landesnetzen ist der Payload zwischen Autor und Leser zusätzlich zu verschlüsseln. Der XTA – WS stellt Autor und Leser hierfür Hilfen wie z.B. Anbindung an entsprechende Verzeichnisdienste (DVDV, S.A.F.E.) zur Verfügung.
  - Kann die Vertraulichkeit nicht durch Verschlüsselung des Payloads zwischen Autor-Sender und / oder Empfänger-Leser sichergestellt werden, ist dies durch andere Maßnahmen, die im Betriebskonzept beschrieben werden, zu gewährleisten. (Beispiel: geteilte Admin-Passwörter, 4-Augenprinzip, erweiterte Protokollsicherung).
  - Fallen die Rollen Autor-Sender oder Empfänger-Leser zusammen, ist dies im Sicherheitskonzept zu vermerken.
- **Integrität**: Die Sicherstellung der Integrität erfolgt i.d.R. durch eine Signatur des Autors und / oder durch die Verschlüsselung des Payloads.
  - Erfolgt keine Verschlüsselung des Payload zwischen Autor-Sender und / oder Empfänger-Leser, ist die Integritätssicherung durch andere Maßnahmen zu kompensieren, siehe Anmerkungen zur Vertraulichkeit.
  - Wenn der Payload durch den Autor signiert wird, geschieht dies auf der fachlichen Ebene. Es betrifft damit nicht den Transport.
- **Verfügbarkeit**: In 98,5% der Datenübermittlungen werden Nachrichten zwischen Sender und Empfänger in durchschnittlich 7,5 Sekunden übermittelt.
- **Transparenz**: Die Transparenz wird wie beschrieben beim Schutzprofil I („normal“) umgesetzt. Verschärfend ist zu beachten, dass die zu liefernden Zeitstempel verlässlich und nicht veränderbar sein müssen. Ebenso ist notwendig, dass für die Dokumentation der Entitätsbezeichner nicht eine Selbstauskunft ausreicht, sondern ein signierter Systembezeichner eingesetzt werden muss. Die gesicherte, signierte System-Protokollierung soll möglichst unter Einsatz eines eigenen Protokollierungsservers erreicht werden.
- **Intervenierbarkeit**: Nachvollziehbar wird die Intervenierbarkeit durch ein geregeltes Change-Management, das Teil des Betriebskonzepts ist.
  - Bei XÖV-Verfahren erfolgen die Ressourcenplanung und ein Monitoring sowohl durch die ausführenden Behörden als auch durch KoSIT bzw. IT-Planungsrat.
  - Klärung,wie Sender bzw. Empfänger bei (standardisierten) Vorfällen, in denen z.B. Autor oder Leser eine Unregelmäßigkeit feststellen, agieren müssen.
  - Klärung, wie Sender und Empfänger auf (standardisierte) Vorfälle in ihrer Zusammenarbeit agieren müssen.
- **Nichtverkettbarkeit**: siehe Anforderungen zu Schutzprofil I („normal“).

Weitere Merkmale:

- **Maximale Durchlaufzeit**: Ist die vorgegebene maximale Durchlaufzeit zur Sicherstellung der rechtlichen Regelungen zu gering, ist gleichwohl sicherzustellen, dass die Umsetzung der rechtlichen Regelung erfolgen kann und ist eine Lösung zu finden.
- **Dialogverfahren**: Um im Kommunikationsprozess grundsätzlich eine direkte Antwort auf eine Anfrage geben zu können, muss der Leser den XTA-Service zur synchronen Kommunikation („SendMsgSync“)

implementiert haben. Das Merkmal kann optinal gesetzt werden, wenn es dem Autor zumutbar ist, dass er keine direkte Antwort erhält.

- **Erweiterte Protokollregeln**: Die erhöhten Anforderungen bezüglich der Datenschutzziele sind primär vom Fachverfahren umzusetzen. Ggf. ist aufgrund erhöhter Anforderungen an Vertraulichkeit und Datenschutz eine erweiterte Protokollsicherung notwendig. Diese werden durch die Fachverfahrensverantwortlichen festgelegt.

B.2 Infrastrukturprofile

Die in den Infrastrukturprofilen gebündelten IT-Komponenten sollen in diesen Kombinationen geeignet sein, die fachlichen Anforderungen an den Datenaustausch angemessen zu erfüllen.

Neben den hier aufgeführten, beispielhaften Infrastrukturprofilen können grundsätzlich weitere definiert werden, die beispielsweise aus der Umsetzung des e-Government-Gesetzes resultieren.

Für den landesinternen Datenaustausch können funktional äquivalente Infrastrukturprofile erstellt werden, druch die insbesondere IT-Komponenten genutzt werden können, die nur innerhalb eines Landes verfügbar sind.

Infrastrukturprofil I bündelt die Komponenten: DOI oder Internet, OSCI-Transport 1.2, OSCI-Intermediär, DVDV, V-PKI.

Diese Komponenten werden in dieser Kombination insbesondere in der Innenverwaltung eingesetzt.

Infrastrukturprofil II bündelt die Komponenten: OSCI-Transport 1.2 gemäß des OSCI-Transportprofils in OSCI-Intermediär, S.A.F.E, V-PKI.

Diese Komponenten werden in dieser Kombination insbesondere im Justizwesen eingesetzt, finden aber auch in der Innenverwaltung Anwendung.

B.3 Wesensprofile

Ein Wesensprofil ist jeweils eine definierte Konfiguration von Infrastrukturkomponenten („Infrastrukturprofil“) für eines der „Schutzprofile“. Die Wesensprofile sollen insbesondere durch die jeweiligen Gremien der XÖV-Standards, ggf. mit Unterstützung der KoSIT, erstellt werden. Bisher entspricht diese Aufgabe im Wesentlichen der Entwicklung und Pflege der in den XÖV-Standards befindlichen „OSCI-Transport-Profilen“, die perspektivisch hierdurch ersetzt werden sollen.

Die hier aufgeführten Beispiele sollen insbesondere die in den Wesensprofilen zu definierenden Inhalte verdeutlichen. In der Erprobungsphase werden sie auf ihre Vollständigkeit und Korrektheit überprüft und entsprechend weiterentwickelt.

Die Prozesse zur Ergänzung der Menge der Wesensprofile oder Änderungen an den einzelnen Profilen sind noch zu definieren und abzustimmen.

B.3.1 Beispielhaftes Wesensprofil: Meldewesen

Dieses Wesensprofil stellt einen Entwurf dar, der in der Erprobungsphase für das Meldewesen (für asynchron durchzuführende Rückmeldungen) verwendet werden kann, sich aber auch als Vorlage für Kommunikationsszenarien im Personenstandswesen und Ausländerwesen anbietet.

Tabelle B.1. Wesensprofil Meldewesen (Rückmeldung)

Eigenschaft	Beispielwert
Transportauftrag	
MessageID	urn:uuid:1234
ID des Payload	ID der fachlichen Nachricht
Aktenzeichen	az1234

Eigenschaft	Beispielwert
Adresse Leser	ags:12345678
Adresse Autor	ags:12345678
Bezeichnung Dienst Leser (nur DVDV)	http://www.osci.de/xmeld181/xmeld181Rueckmeldung.wsdl
Nachrichtentyp	rueckmeldung.anmeldunginland.0201
Anforderung Zeitstempel	false
Lieferzeitraum (von - bis)	
Kontaktdaten von Sender für Empfänger	Wird vom Sender gefüllt - Originators/Sender
Wesensprofil	
Bezeichnung des Intermediärs im VD	TESTAIntermediär/ InternetIntermediär
Bezeichnung Empfänger im VD	TESTAEmpfänger/ InternetEmpfänger
Bezeichnung Leser für Verschlüsselung im OSCI-Bereich im VD	Inhaltsdaten-Verschlüsselungszertifikat
Bezeichnung Leser für Verschlüsselung der XÖV-Nachricht im VD	
Niveau Signatur (Content-Container z.B. der OSCI-Nachricht)	F, SHA 256, RSA
Verschlüsselung (Content-Container z.B. der OSCI-Nachricht)	AES 256
Niveau Signatur Transportebene	optional
Verschlüsselung auf Transportebene	SHA256
Signatur der XÖV-Nachricht	nein
Verschlüsselung der XÖV-Nachricht	nein
Quittierung	Delivery Receipt
Transportstruktur OSCI (Übersetzung der XTA-Nachrichtenstruktur und Scope von Verschlüsselung/Signatur)	OSCI-Struktur: Ein Content im ersten ContentContainer, verschlüsselt (EncryptedDataContainer) & signiert + Es gibt nur einen CC bzw. EDC, der heißt 'XMELD_DATA' XTA-Struktur: im GenericContainerContainer/Content-Container/Message einzutragen, signiert, verschlüsselt usw.
Umsetzung Kommunikationsszenario	XTA-Methode: asynchron (sendMessage) OSCI-Kommunikationsszenario: OneWayActive (Store-Delivery)
Schutzprofil	
Schutzprofil I "Normal"	x
Schutzprofil II "hoch / mit Personenbezug"	
Schutzprofil III "hohe Verfügbarkeit"	
Schutzprofil IV "schnell & sicher"	
Kommunikationsszenario	asynchron
Infrastrukturprofil	
Angabe Verzeichnisdienst	DVDV
Transportstandard	OSCI 1.2

Eigenschaft	Beispielwert
Zertifikatsprüfung	V-PKI

B.3.2 Beispielhaftes Wesensprofil: XHD

Dieses Wesensprofil stellt einen Entwurf dar, der in der Erprobungsphase als Vorlage für Datenübermittlungen von Hoheitlichen Dokumenten (für den synchron durchzuführenden Sperrdienst) verwendet werden kann.

Tabelle B.2. Wesensprofil XHD (Sperrdienst)

Eigenschaft	Beispielwert
Transportauftrag	
MessageID	urn:uuid:1234
ID des Payload	ID der fachlichen Nachricht
Aktenzeichen	az1234
Adresse Leser	dbs:490030020000
Adresse Autor	pab:12345678_00
Bezeichnung Dienst Leser (nur DVDV)	http://www.bsi.de/trxhd/1.2/wsdl/ xhdBeh2eIDSperrdienstOSCI.wsdl
Nachrichtentyp	eIDSperrung
Anforderung Zeitstempel	false
Lieferzeitraum (von - bis)	
Kontaktdaten von Sender für Empfänger	Wird vom Sender gefüllt - Originators/Sender
Wesensprofil	
Bezeichnung des Intermediärs im VD	TESTAIntermediär/ InternetIntermediär
Bezeichnung Empfänger im VD	TESTAEmpfänger/ InternetEmpfänger
Bezeichnung Leser für Verschlüsselung im OSCI-Bereich im VD	
Bezeichnung Leser für Verschlüsselung der XÖV-Nachricht im VD	Inhaltsdaten-Verschlüsselungszertifikat
Niveau Signatur (Content-Container z.B. der OSCI-Nachricht)	F, SHA 256, RSA
Verschlüsselung (Content-Container z.B. der OSCI-Nachricht)	
Niveau Signatur Transportebene	optional
Verschlüsselung auf Transportebene	SHA256
Signatur der XÖV-Nachricht	nein
Verschlüsselung der XÖV-Nachricht	nein
Quittierung	keine Quittung
Transportstruktur OSCI (Übersetzung der XTA-Nachrichtenstruktur und Scope von Verschlüsselung/Signatur)	OSCI-Struktur: Ein Content im ersten ContentContainer, signiert + Es gibt nur einen CC bzw. EDC, der heißt 'XHD' XTA-Struktur: im GenericContainerContainer/Content-Container/Message einzutragen, signiert, usw.

Eigenschaft	Beispielwert
Umsetzung Kommunikationsszenario	XTA-Methode: synchron (sendMessageSync) OSCI-Kommunikationsszenario: RequestResponse (MediateDelivery)
Schutzprofil	
Schutzprofil I "Normal"	
Schutzprofil II "hoch / mit Personenbezug"	
Schutzprofil III "hohe Verfügbarkeit"	
Schutzprofil IV "schnell & sicher"	x
Kommunikationsszenario	synchron
Infrastrukturprofil	
Angabe Verzeichnisdienst	DVDV
Transportstandard	OSCI 1.2
Zertifikatsprüfung	V-PKI

B.4 Definition der drei-stufigen XTA-Konformität

Ziel der Definition einer XTA-Konformität ist es, kontrollierbare Bedingungen zwischen Kommunikations-Endpunkten zu schaffen, die von der öffentlichen Verwaltung leicht nachvollziehbar und überprüfbar sind.

Die XTA-Konformität bezeichnet eine Menge von prüfbaren Anforderungen bzgl. IT-Sicherheit und -Datenschutz an die am Transport beteiligten Komponenten. Diese werden durch XTA-Profile festgelegt.

Als ein Ergebis des Projektes XTA wurde ein Grobkonzept zur Definition einer mehrstufigen XTA-Konformität erstellt, das hier vorgestellt wird. Im Anschluss an das Projekt XTA soll im nächsten Schritt mit der Erprobung des Profilkonzeptes auch diese Definition der XTA-Konformität weiterentwickelt und um ein Überprüfungs- bzw. Zerfizierungsverfahren ergänzt werden.

Um einerseits das Ziel zu erreichen, kontrollierbare Bedingungen zu schaffen und andererseits die Aufwände für die einzelnen Komponenten angemessen auf die jeweiligen Einsatzbereiche zu beschränken, sind in der Definition der XTA-Konformität folgende drei Stufen vorgesehen:

XTA-WS-Spezifikationskonformität: Sie ist erfüllt, wenn die XTA-Webserviceschnittstelle gemäß jeweils aktueller Spezifikation umgesetzt ist. Die XTA-WS-Spezifikationskonformität ist weitgehend unabhängig von den anzuwendenden Profilen. Eine Ausnahme ist hierbei die Vorgaben zur synchronen Kommunikation, die den Profilen entnommen werden und die die Implementierung der entsprechenden Methoden nach sich zieht.

Die Überprüfung der XTA-WS-Spezifikationskonformität erfolgt im Wesentlichen mit Unterstützung von Testbeds.

XTA-Wesenskonformität: Sie setzt die XTA-WS-Spezifikationskonformität voraus. (Für Fachverfahren, die ein eigenes Transportverfahren integriert haben und daher die betroffenen Schnittstellen nicht bedienen müssen, gilt diese Voraussetzung nicht.)

Fach- und Transportverfahren müssen mindestens die jeweils beauftragten Wesensprofile unterstützen.

Von Transportverfahren wird zusätzlich gefordert, dass mindestens die jeweils zu den Wesensprofilen gehörenden Infrastrukturprofile bedient werden können.

Fachverfahren ohne integrierte Transportverfahren sind von den Vorgaben der zu verwendenden Infrastrukturprofile nicht direkt betroffen.

Für Fachverfahren mit integrierten Transportverfahren gilt, dass mindestens die jeweils die beauftragten Wesens- und Infrastrukturprofile bedient werden können müssen.

XTA-Betriebskonformität: Von Transport- und Fachverfahren wird gefordert, dass sie jeweils in Betriebsumgebungen betrieben werden, die für die aus den Profilen resultierenden Anforderungen geeignet sind.

Entsprechend wird die XTA-Konformität für die einzelnen Komponenten wie folgt definiert:

Ein **Fachverfahren (ohne integriertes Transportverfahren) ist XTA-konform**, wenn erfüllt ist:

- XTA-WS Spezifikationskonformität: Das Fachverfahren setzt die XTA-WS Schnittstelle um (Implementierung des XTA-Client).
- XTA-Wesenskonformität: Das Fachverfahren unterstützt mindestens die zu den abgedeckten Fachbereichen gehörenden Wesensprofilen.
- XTA-Betriebskonformität: Das Fachverfahren wird in einer den Wesensprofilen entsprechenden Betriebsumgebung betrieben.

Ein **Transportverfahren ist XTA-konform**, wenn erfüllt ist:

- XTA-WS Spezifikationskonformität: Das Transportverfahren setzt die XTA-WS Schnittstelle um. (Implementierung des XTA-Servers).
- XTA-Wesenskonformität: Das Transportverfahren unterstützt mindestens die Wesensprofile, für die es beauftragt ist, und die dazu gehörigen Infrastrukturprofile.
- XTA-Betriebskonformität: Das Transportverfahren wird in einer den Wesensprofilen entsprechenden Betriebsumgebung betrieben.

Ein **Fachverfahren mit integriertem Transportverfahren ist XTA-konform**, wenn erfüllt ist:

- XTA-Wesenskonformität: Das Fachverfahren mit integriertem Transportverfahren unterstützt mindestens die zu den abgedeckten Fachbereichen gehörenden Wesensprofile sowie die dazu gehörigen Infrastrukturprofile.
- Betriebskonformität: Das Fachverfahren mit integriertem Transportverfahren wird in einer den Wesensprofilen entsprechenden Betriebsumgebung betrieben.
- (Die Forderung einer XTA-WS-Spezifikationskonformität entfällt.)

Eine (vollständige oder teilweise) XTA-Konformität muss für die Verwaltung nachvollziehbar und überprüfbar sein. Hierfür ist es notwendig, für die am Transport beteiligten Komponenten jeweils angemessene Verfahren zu entwickeln, so dass eine definierte Qualität durchsetztbar wird.

C Asynchroner Empfang von Nachrichten (Zugriff mehrerer Leser auf ein Postfach)

Die hier beschriebene Variante wird in der aktuellen Version der XTA-WS nicht unterstützt. Sie berührt daher nicht die Anforderungen bzgl. der XTA-Konformität.

Bei einem asynchronen Empfang nimmt der Empfänger die Nachrichten entgegen und hält diese für den Leser für eine Abholung bereit. Der Leser kann die Nachrichten zu einem von ihm bestimmten Zeitpunkt abholen.

In dieser Variante des asynchronen Empfangs kann die Abholung parallel von mehreren Lesern erfolgen. Jeder Leser erhält einen Zugriff auf die Liste der abzuholenden Nachrichten, die der Empfänger verwaltet. Holen die Leser parallel Nachrichten ab, dann erhalten sie vom Empfänger jeweils disjunkte Mengen von Nachrichten.

In der folgenden Darstellung der Abholung von Nachrichten gibt es nur einen aktiven Leser. Für eine parallele Abholung durch mehrere Leser reserviert ein Leser in Schritt 1 einen Ressourcenhandle. Diesen reicht er an andere Leser weiter, die dann die folgenden Arbeitsschritte 2 und 3 parallel zu ihm durchführen.

1. Abholung der ersten Nachricht mit Kriterien

   Der Leser holt die erste Nachricht ab (Anhang A Rollenmodell D5.1). Hierbei kann er Selektionskriterien (gelesen, ungelesen; Zeitraum des Empfangs) angeben. Mit der ersten Nachricht bekommt der Leser eine Ressourcenkennung für einen „Iterator“, die er für die weitere Abholung von Nachrichten benötigt. (Wenn mehrere Leser auf die Nachrichten zugreifen wollen, reicht er diese Ressourcenkennung an andere Leser weiter.)

   - Abholung der ersten Nachricht (siehe Abschnitt 4.4.3.2 auf Seite 63)

2. Überprüfung der Kommunikation

   Der Leser überprüft, ob der Transport der Nachricht erfolgreich durchgeführt werden konnte (Anhang A Rollenmodell D7.1, D 8.1), z. B. ob die verwendeten Zertifikate gültig waren. Bei positivem Ergebnis kann er die Nachricht des Autors verarbeiten. Im Falle eines Misserfolgs muss er ggf. Eskalationsmaßnahmen ergreifen.

   XTA Funktionalitäten:

   - Abruf eines Transportprotokolls (siehe Abschnitt 4.4.1.3 auf Seite 45)

3. Liste der Nachrichten mit Iterator abholen

   Der Leser hat für das Abholen der Nachrichten vom Empfänger eine Ressourcenkennung erhalten. Unter Angabe dieser Kennung kann er die nächste Nachricht abholen (Anhang A Rollenmodell D5.1). Dabei sollte er die MessageID der zuletzt abgeholten Nachricht mit angeben. Dadurch quittiert er die erfolgreiche Abholung dem Empfänger. Liegt keine weitere Nachricht vor, dann liefert der Empfänger eine entsprechende Meldung zurück.

   XTA Funktionalitäten:

- Abholen einer weiteren Nachricht

  (siehe Abschnitt C.1.1 auf Seite 108)

4. Beenden der Abholung von Nachrichten

   Wenn der Leser alle Nachrichten abgeholt hat, soll er dies dem Empfänger mitteilen, indem er abschließend eine Quittung sendet. Diese Information unterstützt den Empfänger bei der Koordination des parallelen Zugriffs.

   XTA Funktionalität:

   - Quittieren der Abholung (siehe Abschnitt 4.4.3.3 auf Seite 65)

C.1 Methoden

Zur Umsetzung des parallelen Zugriffs auf ein Postfach werden zusätzlich folgende Methoden benötigt:

- getNextMessage
- getNextStatusList

C.1.1 Methode getNextMessage (Abholen einer nächsten Nachricht)

Der Leser kann für die Abholung von Nachrichten vom Empfänger einen sogenannten Iterator verwenden (siehe z. B. „Abholung der ersten Nachricht mit Kriterien“). Dieser Iterator wird durch eine Ressourcenkennung adressiert, die vom Empfänger vergeben wurde. Der Leser kann weitere Nachrichten unter Angabe dieser Ressourcenkennung erhalten. Zugleich kann er die MessageID der letzten erfolgreich abgeholten Nachricht angeben, wodurch der Leser dem Empfänger den Empfang der Nachricht quittiert. Der Empfänger liefert die nächste Nachricht, bis keine weitere Nachricht vorhanden ist.

C.1.1.1 Operation getNextMessage

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MsgBoxGetNextRequest	osci:MsgBoxGetNextRequestType

Wesentliche Parameter:

- „osci:MsgBoxRequestId“: Angabe der Ressourcenkennung des Iterators.
- „wsa:MessageID“: Der Leser gibt die ID der zuletzt abgeholten Nachricht an. Dadurch quittiert er den erfolgreichen Empfang. (optionaler Parameter)

Output.

Soap Part	Name	Type
Header	MsgBoxResponse	osci:MsgBoxResponseType
Body	GenericContentContainer	xta:GenericContentContainer

Rückgabewerte:

- „xta:GenericContentContainer“: Dieses Objekt enthält die nächste Nachricht. Sie besteht aus der eigentlichen Nachricht und einer beliebigen Anzahl von Anhängen (Attachments). Die Nachricht sel-

ber kann in einem verschlüsselten Container hinterlegt werden. Zu der Nachricht kann ein Betreff (Subject) angegeben werden.

C.1.2 Methode getNextStatusList (Nächste Teilliste von MessageIDs und Metadaten holen)

Insbesondere bei umfangreichen Datenmengen oder bei parallelem Zugriff mehrerer Leser kann es sinnvoll sein, sich die Liste der MessageIDs und Metadaten für Nachrichten blockweise geben zu lassen. Der Vorgang des Abrufs einer Teilliste erfolgt, indem nur eine bestimmte Anzahl von MessageIDs und Metadaten erwartet und zurückgeliefert wird. Zusätzlich wird eine Ressourcenkennung („Handle“) für einen Iterator geliefert, mit der die weiteren Blöcke („Teillisten“) von MessageIDs mit der Methode getNextStatusList (siehe Abschnitt C.1.2 auf Seite 109) abgeholt werden können.

Die Verwendung der Methode getNextStatusList steht im direkten Zusammenhang mit der Methode getStatusList: Der Leser kann mit dieser Methode Teillisten von MessageIDs von Metadaten vom Empfänger abholen, nachdem er die erste dieser Teillisten mit der Methode getStatusList (siehe Abschnitt 4.4.3.1 auf Seite 62) erhalten hat:

C.1.2.1 Ergebnisse

- Liste der Ergebnisparameter. Im Ergebnis-Header (MsgBoxResponse) werden Zusatzinformationen zum Anfragevorgang und seiner Ergebnisliste geliefert.
- Der technische Fehler (SoapFault) <PermissionDeniedException> entsteht, wenn der Account gesperrt oder nicht vorhanden ist.
- Der technische Fehler (SoapFault) <XTAWSTechnicalProblemException> entsteht, wenn ein technischer Fehler im XTA-WS aufgetreten ist.

C.1.2.2 Operation getNextStatusList

Input.

Soap Part	Name	Type
Header	AuthorIdentifier	osci21:PartyType
Body	MsgBoxGetNextRequest	osci:MsgBoxGetNextRequestType

Wesentliche Parameter:

- "osci:MsgBoxRequestID“: Die Ressourcenkennung für den Zugriff auf den Iterator.

Output.

Soap Part	Name	Type
Header	MsgBoxResponse	osci:MsgBoxResponseType
Body	MsgStatusList	osci:MsgStatusListType

Rückgabewerte:

- "osci:MsgBoxRequestID“: Die Ressourcenkennung für die weiteren Zugriffe auf den Iterator.
- „osci:NoMessageAvailable“: Wurden zu den angegebenen Suchargumenten keine Daten gefunden, wird die Ursache angegeben.
- „osci:ItemsPending“: Anzahl der Nachrichten, auf die die Anfrage zutrifft.
- „osci:MsgStatusList“: Angabe der folgenden Informationen pro Nachricht:

- „wsa:MessageID: ID des Transportauftrags.
- „wsa:RelatesTo: Referenzen auf Objekte, z. B. Vorgängernachrichten
- „wsa:From“: Kennung des Autoren
- „osci:TypeOfBusinessScenario“: Angabe des Geschäftsprozesses
- „osci21:MsgSize“: Größe der Nachricht.
- „osci21:ObsoleteAfter“: Nach diesem Zeitpunkt wird die Nachricht gelöscht und darf also nicht mehr ausgeliefert werden.
- „osci21:Delivery“: Zeitpunkt des Eintreffens beim Empfänger, z. B. bei einem OSCI-Intermediär.
- „osci21:InitialFetch“: Zeitpunkt der erstmaligen Auslieferung zum Leser.

C.1.2.3 Beispielcode (Aufruf der Methode)

```
getNextStatusList(„osci:8dfrg8e7o485zfiuz84r7349)
```

D Beispielcode

Als zusätzliche Dokumentation werden die unterschiedlichen Versand- und Empfangsoptionen in Beispielcode (in einer an verbreitete Programmiersprachen angelehnte Pseudo-Sprache) dargestellt. Er soll den prinzipiellen Aufbau der Programme bzgl. der Nutzung der XTA-Funktionalitäten verdeutlichen.

D.1 Autor

Es werden folgende Variablen verwendet:

- $myAuthor: Die fachliche Kennung des Autors.
- $myMetadata: Die zu der zu versendenden Nachricht gehörende Metadaten
- $myMessage: Ein GenericContantContainer, der sowohl die zu sende Nachricht als auch die Antwort Nachricht enthält.
- $myMessageID: Die ID des zurückzurufenden Transportauftrags.
- $myLookupServiceRequest: Liste der zu prüfenden Empfänger
- $myLookupServiceResponse: Liste der geprüften Empfänger mit Prüfergebnis
- $myX509TokenContainer: Liste der zu Prüfenden Zertifikate

Ergänzend zu den XTA-WS-Methoden müssen für den Leser folgende Funktionen implementiert werden:

- extractState() : Aus dem TransportReport wird der Status extrahiert, der angibt, ob der Versand bereits durchgeführt wurde und ob er erfolgreich durchgeführt wurde.
- sendEscalationForMessage(): Hinweis auf ein Versand-Problem

D.1.1 Asynchroner Versand einer Nachricht

```
# Zuerst wird geprüft:
# 1. Gibt es einen aktiven Account beim Sender/Empfänger?
# 2. Funktioniert die Verbindung zum Sender/Empfänger?
if (not checkAccountActive($myAuthor)) {
   exit;
}

# Bietet der Empfänger den gewünschten Dienst an?
$myLookupServiceResponse =
lookupService($myAuthor, $myLookupServiceRequest);
```

```
# Erzeugen einer neuen MessageID
$aMessageID = ceateMessageID($myAuthor);

Try
{
  # Asynchroner Versand der Nachricht
  sendMessage($myMetadata, $myX509TokenContainer, $myMessage);

  # Wurde der Transport erfolgreich durchgeführt?
  # Hole Report bis Versandauftrag bearbeitet wurde. Eine
  # angemessene Wartezeit verhindert unangemessen viele
  # Anfragen beim Sender.

  repeat {
    # Eine Wartepause lässt dem Sender Zeit zum Senden.
    sleep(3600);
    # TransportReport holen
    $aReport = getTransportReport($aMessageId);
    # Den Status des Transportauftrags auslesen
    $aState = extractState($aReport);
  } until ($aState != „offen“);

  # Ist der Versand fehlgeschlagen, wird der
  # Verantwortliche informiert.
  if ($aState equal „rot“) {
    sendEscalationForMessage($aMessageID, $aReport);
  }
}
Catch
{
   # XTA Exception verarbeiten
   sendEscalationForMessage($myMessage);
}
```

D.1.2 Synchroner Versand einer Nachricht

```
# Zuerst wird geprüft:
# 1. Gibt es einen aktiven Account beim Sender/Empfänger?
# 2. Funktioniert die Verbindung zum Sender/Empfänger?
if (not checkAccountActive($myAuthor)) {
   exit;
}

# Bietet der Empfänger den gewünschten Dienst an?
$myLookupServiceResponse =
lookupService($myAuthor, $myLookupServiceRequest);

# Erzeugen einer neuen MessageID
$aMessageID = ceateMessageID($myAuthor);

Try
{
```

```
  # Synchroner Versand der Nachricht. Die Argumente werden
  # als „ref“ übergeben, d. h. sie können geändert werden.
  # Hier wird die Antwort zurückgegeben.
  sendMessageSync(ref $myMetadata, ref $myX509TokenContainer,
                  ref $myMessage);

  # TransportReport holen
  $aReport = getTransportReport($myAuthor, $aMessageID);
  # Den Status des Transportauftrags auslesen
  $aState = extractState($myAuthor, $aReport);

  # Ist der Versand fehlgeschlagen, wird der
  # Verantwortliche informiert.
  if ($aState equal „rot“) {
    sendEscalationForMessage($aMessageID, $aReport);
  }
}
Catch
{
   # XTA Exception verarbeiten
   sendEscalationForMessage($myMessage);
```

D.1.3 Rückruf einer Nachricht

```
# Zuerst wird geprüft:
# 1. Gibt es einen aktiven Account beim Sender/Empfänger?
# 2. Funktioniert die Verbindung zum Sender/Empfänger?
if (not checkAccountActive($myAuthor)) {
   exit;
}

# Status der Nachricht im TransportReport prüfen.
$aReport = getTransportReport($myAuthor, $myMessageID);
# Den Status des Transportauftrags auslesen
$aState = extractStatus($myAuthor, $aReport);

# Rückruf der Nachricht – falls die noch nicht verschickt
# wurde.
If ($aState == „offen“) {
   cancelMessage($myAuthor ,$myMessageID);
}
```

D.2 Leser

In den Beispielen werden folgende Variablen verwendet:

- $myFilter: Filter zur Beschreibung der Attribute der abzuholenden Nachrichten.
- $myFrom: Die abzuholenden Nachrichten sollen nach diesem Zeitpunkt eingegangen sein.
- $myTo: Die abzuholenden Nachrichten sollen vor diesem Zeitpunkt eingegangen sein.

Ergänzend zu den XTA-WS-Methoden müssen für den Leser folgende Funktionen implementiert werden:

- setPeriodOfTime: Mit dieser Funktion wird der Zeitraum gesetzt, in dem die abzuholenden Nachrichten angekommen sein müssen.
- setState(): Hiermit wird der Status gesetzt, den die abzuholenden Nachrichten haben sollen.
- setMaxCount(): Hiermit wird im Filter die maximale Anzahl der abzuholenden Einträge gesetzt.
- extractMessage(): Die Funktion liest aus der Rückgabe von getMessage() die enthaltene Nachricht aus.
- extractMessageIDs(): Diese Funktion liest aus der Rückgabe der WS-Methode getStatusList() die enthaltenen MessageIDs aus.
- extractMetadatas(): Diese Funktion liest aus der Rückgabe der WS-Methode getStatusList() die enthaltenen Metadaten aus.
- extractState() : Aus dem TransportReport wird der Status extrahiert der angibt, ob der Versand bereits durchgeführt wurde und wenn ja, ob er erfolgreich durchgeführt wurde.
- extractHandle(): Diese Funktion liest aus den Rückgaben der WS-Methoden getMessage() und getStatusList() die enthaltenen Ressourcenkennung (Handle) aus.
- sendEscalationForMessage(): Hiermit wird ein Problem mit dem Transport eskaliert.
- messageProcessing(): Mit dieser Funktion wird die empfangene Nachricht fachlich verarbeitet.
- metadataProcessing(): Mit dieser Funktion werden die Metadaten einer empfangenen Nachricht verarbeitet.
- popMetadata(): Die Funktion entfernt das erste Metadatenelement aus einer Liste und liefert es als Ergebnis.
- hasNext(): Die Funktion prüft ob noch weitere Einträge vorhanden sind.
- notEmpty(): Die Funktion prüft ob es sich nicht um eine leere Liste handelt.
- startWebService(): Hiermit wird der WebService des Lesers gestartet, der auf eintreffende Nachrichten wartet. Beim Eintreffen einer Nachricht beendet sich der WebService und liefert die empfangene Nachricht als Ergebnis.

D.2.1 Asynchroner Empfang von Nachrichten

```
# Definiere gewünschte Nachrichten über Angabe eines
# Zeitintervalls und ob neue, alte oder alle Nachrichten
# abgeholt werden sollen.
setPeriodOfTime($myFilter, $myFrom, $myTo);
setState($myFilter, „NeueNachrichten“);

# Liste der MessageIDs holen
$result = getStatusList($myFilter);
$listOfMessageIDs = extractMessageIDs($result);

# Für jede einzelne ID die Nachricht abholen
foreach $anId ($listOfMessageIDs) {

   # Report für die Nachricht holen und Status extrahieren
   $aReport = getTransportReport($anId);

   # Status extrahieren
   $aState = extractState($aReport);

   # War beim Transport alles ok, dann darf die Nachricht
   # verarbeitet werden.
   If ( $aState equal „grün“ ) {
```

```
      # getMessage() liefert die Nachricht und
      # Zusatzinformationen. Hiervon brauchen wir
      # die Ressourcenkennung (Neudeutsch: Handle)
      $result = getMessage($anId);
      $aMessage = extractMessage($result);
      $aGetMessageHandle = extractHandle($result);

      # Ressourcen freigeben
      close($aGetMessageHandle);

      # Alles ist ok. Die Nachricht darf verarbeitet
      # werden.
      messageProcessing($aMessage);
   } else {
      sendEscalationForMessage($anId, $aReport);
   }
}
```

D.2.2 Asynchroner Empfang von Nachrichten – (Zugriff mehrerer Leser)

```
# Definiere gewünschte Nachrichten über Angabe eines
# Zeitintervalls und ob neue, alte oder alle Nachrichten
# abgeholt werden sollen.
setPeriodOfTime($myFilter, $myFrom, $myTo);
setState($myFilter, „NeueNachrichten“);

# Ressourcenkennung für den Zugriff auf die Liste
# der gewünschten Nachrichten holen.
$result = getMessage($myFilter);
$aMessage = extractMessage($result);
$aGetMessageHandle = extractHandle($result);

# Solange Nachrichten abholen bis keine mehr da
# sind.
while ( hasNext($aGetMessageHandle) ){
   # Abholen der nächsten Nachricht.
   $result = getNextMessage($aGetMessageHandle);

   # Die eingebettete Nachricht auslesen.
   $aMessage = extractMessage($result);

   # Die MessageID holen
   $anId = extractMessageIDs($result);

   # Report für die Nachricht holen und Status extrahieren
   $aReport = getTransportReport($anId);

   # Status extrahieren
   $aState = extractState($aReport);

   # War beim Transport nicht alles ok, dann
```

```
   # wird die Nachricht eskaliert.
   If ( $aState equal „rot“ ) {
      sendEscalationForMessage($anId, $aReport);
   } else {
      # Alles ist ok. Die Nachricht darf verarbeitet
      # werden.
      messageProcessing($aMessage);
   }
}

# Ressourcen freigeben
close($aGetMessageHandle);
```

D.2.3 Asynchroner Empfang der Metadaten

```
# Definiere gewünschte Metadaten über Angabe eines
# Zeitintervalls und ob neue, alte oder alle Nachrichten
# abgeholt werden sollen. Es sollen jeweils maximal 100
# MessageIDs und Metadaten geholt werden.
setPeriodOfTime($myFilter, $myFrom, $myTo);
setState($myFilter, „NeueNachrichten“);
setMaxCount($myFilter, 100);

# Liste der MessageIDs und Metadaten holen
$result = getStatusList($myFilter);
$listOfMessageIDs = extractMessageIDs($result);
$listOfMetadatas = extractMetadatas($result);
$aGetStatusListHandle = extractHandle($result);

while ( notEmpty($listOfMessageIDs) ) {

   # Überprüfung des Transports und Verarbeitung der
   # einzelnen Metadaten.
   foreach $anId ($listOfMessageIDs) {

      # Report für die Nachricht holen und Status extrahieren
      $aReport = getTransportReport($anId);

      # Status extrahieren
      $aState = extractState($aReport);

      # War beim Transport nicht alles ok, dann
      # wird die Nachricht eskaliert.
      If ( $aState equal „rot“ ) {
         sendEscalationForMessage($anId, $aReport);
      } else {
         # Alles ist ok. Die Metadaten dürfen verarbeitet
         # werden.
         $aMetadata = popMetadata($listOfMetadatas);
         metadataProcessing($aMetadata);
      }
   }
```

```
   # Die nächsten MessageIDs holen
   $listOfMessageIDs = getNextStatusList($aGetStatusListHandle);
}
# Ressourcen freigeben
close($aGetStatusListHandle);
```

D.2.4 Synchroner Empfang von Nachrichten

```
# Der Empfang der Nachricht soll immer laufen.
while ( true ) {
   # Den Empfang für die Methode sendMessageSync starten.
   # Er wird beim Empfang einer Nachricht beendet. Die
   # Nachricht wird als Rückgabewert geliefert.
   $aMessage = startWebService(„sendMessageSync“);

   # Die MessageID holen
   $anId = extractId($aMessage);

   # Report für die Nachricht holen und Status extrahieren
   $aReport = getTransportReport($anId);

   # Status extrahieren
   $aState = extractState($aReport);

   # War beim Transport nicht alles ok, dann
   # wird die Nachricht eskaliert.
   if ( $aState equal „rot“ ) {
      sendEscalationForMessage($anId, $aReport);
   } else {
      # Alles ist ok. Die Nachricht darf verarbeitet werden.
      messageProcessing($aMessage);
   }
}
```

E Schlüsseltabellen

Name	# Einträge	Tabellendetails	Code-Datentyp
Record	Anzahl nicht bekannt, da Codeliste extern gepflegt wird	Seite 119	Seite 68
Report	Anzahl nicht bekannt, da Codeliste extern gepflegt wird	Seite 120	Seite 68

E.1 Details

E.1.1 Schlüsseltabelle Record

Codeliste	Record (urn:de:kosit:xta:codelist:Record)
Herausgeber	KoSIT
Beschreibung	Diese Tabelle enthält die Arten von Meldungen im Protokoll (TransportReport), also Arten von Fehler-, Warn- und Informationseinträgen. Die Schlüsseltabelle wird im XRepository (www.xrepository.de) im XML-Format OASIS Genericode über die ID urn:de:kosit:xta:codelist:Record bereitgestellt.
Schlüssel	Wert

E.1.2 Schlüsseltabelle Report

Codeliste	**Report (urn:de:kosit:xta:codelist:Report)**
Herausgeber	**KoSIT**
Beschreibung	**Diese Tabelle enthält die möglichen Arten von Reports, die in eigenen Formaten in dem Protokoll (TransportReport) eingefügt werden können.** **Die Schlüsseltabelle wird im XRepository (www.xrepository.de) im XML-Format OASIS Genericode über die ID urn:de:kosit:xta:codelist:Report bereitgestellt.**
Schlüssel	**Wert**

F Eingebundene externe Modelle

Folgende externe Modelle werden in dieser Spezifikation verwendet:

F.1 OSCI-Transport-V2.01

Der Standard selbst, die dazugehörige Spezifikation und weitergehende Informationen und Materialien sind unter http://www.xoev.de/de/download#OSCI-Transport2 verfügbar.

Präfix: osci21

Version: 2.01

Namensraum: http://www.osci.eu/ws/2013/02/transport

SchemaLocation-Basis: http://www.osci.eu/ws/2013/02/transport/

Folgende Schema-Dateien werden verwendet:

- OSCI21_MessageMetaData.xsd

F.2 WS-Addressing

Präfix: ws-addressing

Version:

Namensraum: http://www.w3.org/2005/08/addressing

SchemaLocation-Basis: http://www.w3.org/2005/08/addressing/

Folgende Schema-Dateien werden verwendet:

- ws-addr.xsd

F.3 XML-Encryption

Präfix: xml-encryption

Version: 1.0

Namensraum: http://www.w3.org/2001/04/xmlenc#

SchemaLocation-Basis: http://xoev.de/xta/2013/07/

Folgende Schema-Dateien werden verwendet:

- xenc-schema.xsd

F.4 XML-Signature

Präfix: xmldsig-core

Version: 0.1

Namensraum: http://www.w3.org/2000/09/xmldsig#

SchemaLocation-Basis: http://www.w3.org/TR/xmldsig-core/

Folgende Schema-Dateien werden verwendet:

- xmldsig-core-schema.xsd

F.5 XÖV-Basisdatentypen-V1.1

Präfix: XOEV-Basisdatentypen

Version: 1.1

Namensraum: http://xoev.de/schemata/basisdatentypen/1_1

SchemaLocation-Basis: http://xoev.de/schemata/basisdatentypen/1_1/

Folgende Schema-Dateien werden verwendet:

- xoev-basisdatentypen.xsd