

## BSI Technische Richtlinie 03125 Beweiswerterhaltung kryptographisch signierter Dokumente

### **Anlage TR-ESOR-E: Konkretisierung der Schnittstellen auf Basis des eCard-API-Frameworks**

Bezeichnung	Konkretisierung der Schnittstellen auf Basis des eCard-API-Frameworks
Kürzel	BSI TR-ESOR-E
Version	1.2
Datum	31.01.15

Bundesamt für Sicherheit in der Informationstechnik  
Postfach 20 03 63  
53133 Bonn  
Tel.: +49 228 99 9582-0  
E-Mail: [tresor@bsi.bund.de](mailto:tresor@bsi.bund.de)  
Internet: <https://www.bsi.bund.de>

© Bundesamt für Sicherheit in der Informationstechnik 2015

## Inhaltsverzeichnis

1. Einführung.....	5
2. Überblick.....	7
3. Funktionen der ArchiSafe-Schnittstelle (TR-ESOR-S.4).....	10
3.1 ArchiveSubmissionRequest und ArchiveSubmissionResponse.....	10
3.1.1 ArchiveSubmissionRequest.....	10
3.1.2 ArchiveSubmissionResponse.....	13
3.2 ArchiveUpdateRequest und ArchiveUpdateResponse.....	15
3.2.1 ArchiveUpdateRequest.....	15
3.2.2 ArchiveUpdateResponse.....	16
3.3 ArchiveRetrievalRequest und ArchiveRetrievalResponse.....	18
3.3.1 ArchiveRetrievalRequest.....	18
3.3.2 ArchiveRetrievalResponse.....	20
3.4 ArchiveEvidenceRequest und ArchiveEvidenceResponse.....	21
3.4.1 ArchiveEvidenceRequest.....	22
3.4.2 ArchiveEvidenceResponse.....	23
3.5 ArchiveDeletionRequest und ArchiveDeletionResponse.....	25
3.5.1 ArchiveDeletionRequest.....	25
3.5.2 ArchiveDeletionResponse.....	26
3.6 ArchiveDataRequest und ArchiveDataResponse.....	27
3.6.1 ArchiveDataRequest.....	28
3.6.2 ArchiveDataResponse.....	29
3.7 VerifyRequest und VerifyResponse.....	31
3.7.1 VerifyRequest.....	31
3.7.2 VerifyResponse.....	35
4. Funktionen der interne Schnittstellen.....	37
4.1 TR-ESOR-S.1 (ArchiSafe-Modul – Krypto-Modul).....	37
4.1.1 Prüfung von Signaturen, beweisrelevanten Daten, Beweisdaten und Archivdatenobjekten.....	37
4.1.2 Signaturerstellung.....	37
4.1.2.1 SignRequest.....	37
4.1.2.2 SignResponse.....	38
4.2 TR-ESOR-S.2 (ArchiSig-Modul – ECM-/Langzeitspeichersystem).....	39
4.2.1 Speichern eines Archivdatenobjektes.....	39
4.2.2 Ergänzen einer neuen Version eines Archivdatenobjektes.....	39
4.2.3 Auslesen von Archivdatenobjekten.....	39
4.3 TR-ESOR-S.3 (ArchiSig-Modul – Krypto-Modul).....	40
4.3.1 Anfordern eines (qualifizierten) Zeitstempels.....	40
4.3.1.1 TimestampRequest wird realisiert durch SignRequest.....	40
4.3.1.2 TimestampResponse wird realisiert durch SignResponse.....	41
4.3.2 Prüfen eines (qualifizierten) Zeitstempels.....	41
4.3.2.1 VerifyRequest.....	42
4.3.2.2 VerifyResponse.....	43

4.3.3 Berechnung eines Hashwertes.....	43
4.3.3.1 Hash.....	44
4.3.3.2 HashResponse.....	45
4.4 TR-ESOR-S.5 (ArchiSafe-Modul – ECM-Langzeitspeichersystem).....	45
4.4.1 Abfrage beweiswerterhaltend archivierter Daten.....	45
4.4.2 Löschen von Archivdatenobjekten.....	45
4.4.3 Abfrage diskreter Datenobjekte.....	46
4.5 TR-ESOR-S.6 (ArchiSafe-Modul – ArchiSig-Modul).....	46
4.5.1 Beweiswerterhaltende Archivierung elektronischer Daten.....	46
4.5.2 Ergänzen einer neuen Version eines Archivdatenobjektes.....	46
4.5.3 Rückgabe technischer Beweisdaten.....	46
5. Fehlercodes.....	47
6. Spezifikation einer Webservice-basierten Schnittstelle.....	49
6.1 Spezifikation der Aufruf- und Rückgabeparameter als XML-Schema.....	49
6.2 WSDL-Spezifikation der Schnittstelle TR-ESOR-S.4.....	54

## 1. Einführung

Ziel der Technischen Richtlinie „Beweiswerterhaltung kryptographisch signierter Dokumente“ ist die Spezifikation sicherheitstechnischer Anforderungen für den langfristigen Beweiswerterhalt von kryptographisch signierten elektronischen Dokumenten und Daten nebst zugehörigen elektronischen Verwaltungsdaten (Metadaten).

Eine für diese Zwecke definierte Middleware (TR-ESOR-Middleware) im Sinn dieser Richtlinie umfasst alle diejenigen Module (**M**) und Schnittstellen (**S**), die zur Sicherung und zum Erhalt der Authentizität und zum Nachweis der Integrität der aufbewahrten Dokumente und Daten eingesetzt werden.

Die im Hauptdokument dieser Technischen Richtlinie vorgestellte Referenzarchitektur besteht aus den nachfolgend beschriebenen funktionalen und logischen Einheiten:

- der Eingangs-Schnittstelle S.4 der TR-ESOR-Middleware, die dazu dient, die TR-ESOR-Middleware in die bestehende IT- und Infrastrukturlandschaft einzubetten;
- dem „ArchiSafe-Modul“ ([**TR-ESOR-M.1**]), welches den Informationsfluss in der Middleware regelt, die Sicherheitsanforderungen an die Schnittstellen zu den IT-Anwendungen umsetzt und für eine Entkopplung von Anwendungssystemen und ECM/Langzeitspeicher sorgt;
- dem „Krypto-Modul“ ([**TR-ESOR-M.2**]) nebst den zugehörigen Schnittstellen S.1 und S.3, das alle erforderlichen Funktionen zur Erstellung (optional) und Prüfung elektronischer Signaturen, zur Nachprüfung elektronischer Zertifikate und zum Einholen qualifizierter Zeitstempel für die Middleware zur Verfügung stellt. Darüber hinaus kann es Funktionen zur Ver- und Entschlüsselung von Daten und Dokumenten zur Verfügung stellen;
- dem „ArchiSig-Modul“ ([**TR-ESOR-M.3**]) mit der Schnittstelle S.6, das die erforderlichen Funktionen für die Beweiswerterhaltung der digital signierten Unterlagen bereitstellt;
- einem ECM/Langzeitspeicher mit den Schnittstellen S.2 und S.5, der die physische Archivierung/Aufbewahrung und auch das Speichern der beweiswerterhaltenden Zusatzdaten übernimmt.

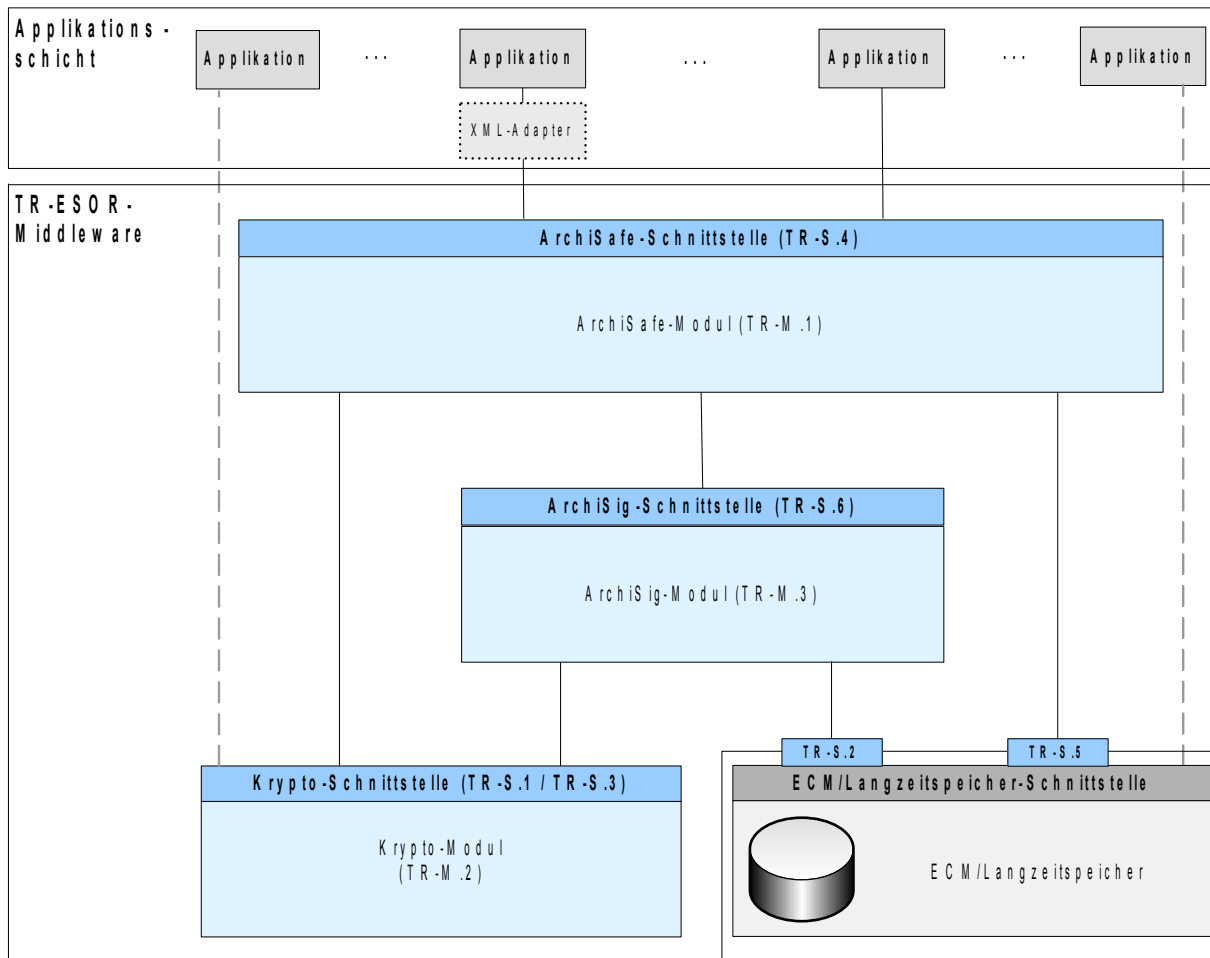
*Dieser ECM/Langzeitspeicher ist nicht mehr direkt Teil der Technischen Richtlinie, gleichwohl werden über die beiden Schnittstellen, die noch Teil der TR-ESOR-Middleware sind, Anforderungen daran gestellt.*

*Ebenso wenig ist die Applikationsschicht, die auch einen XML-Adapter enthalten kann, direkter Teil der Technischen Richtlinie, auch wenn dieser XML-Adapter als Teil einer Middleware implementiert werden kann.*

Die in Abbildung 1 dargestellte IT-Referenzarchitektur orientiert sich an der ArchiSafe<sup>1</sup> Referenzarchitektur und soll die logische (funktionale) Interoperabilität künftiger Produkte mit den Zielen und Anforderungen der Technischen Richtlinie ermöglichen und unterstützen.

---

<sup>1</sup> Siehe dazu <http://www.archisafe.de>



**Abbildung 1: Schematische Darstellung der IT-Referenzarchitektur**

Diese Technische Richtlinie ist modular aufgebaut und spezifiziert in einzelnen Anlagen zum Hauptdokument die funktionalen und sicherheitstechnischen Anforderungen an die erforderlichen IT-Komponenten und Schnittstellen der TR-ESOR-Middleware. Die Spezifikationen sind strikt plattform-, produkt-, und herstellerunabhängig.

Das vorliegende Dokument trägt die Bezeichnung „Anlage TR-ESOR-E“ und konkretisiert die in [TR-ESOR-S] eingeführten Schnittstellen auf Basis des in der BSI TR 03112 spezifizierten eCard-API-Frameworks.

## 2. Überblick

Wie in Abschnitt 9 des Hauptdokumentes näher erläutert, ist für den Nachweis der Konformität zur vorliegenden technischen Richtlinie ein dreistufiges Verfahren vorgesehen.

**(A2.0-1)** Demnach muss in *Konformitätsstufe 1* lediglich die funktionale und logische Konformität eines aus mindestens einem in dieser Richtlinie spezifizierten Modul bestehenden Produktes oder Systems mit den Anforderungen der Richtlinie nachgewiesen werden. Die Unterstützung der einzelnen in [TR-ESOR-S] und hier beschriebenen Schnittstellen ist somit optional.

**(A2.0-2)** Sofern bei der Realisierung eines aus mindestens einem in dieser Richtlinie spezifizierten Modul bestehenden Produktes oder Systems die technische Konformität und Interoperabilität der *Konformitätsstufe 2* nachgewiesen werden soll, muss diese auf Basis der in diesem Dokument beschriebenen Profilierung des eCard-API-Frameworks umgesetzt werden.

Hierbei müssen die folgenden im vorliegenden Dokument näher aufgeführten Funktionen mit den hier beschriebenen Parameterkonstellationen unterstützt werden:

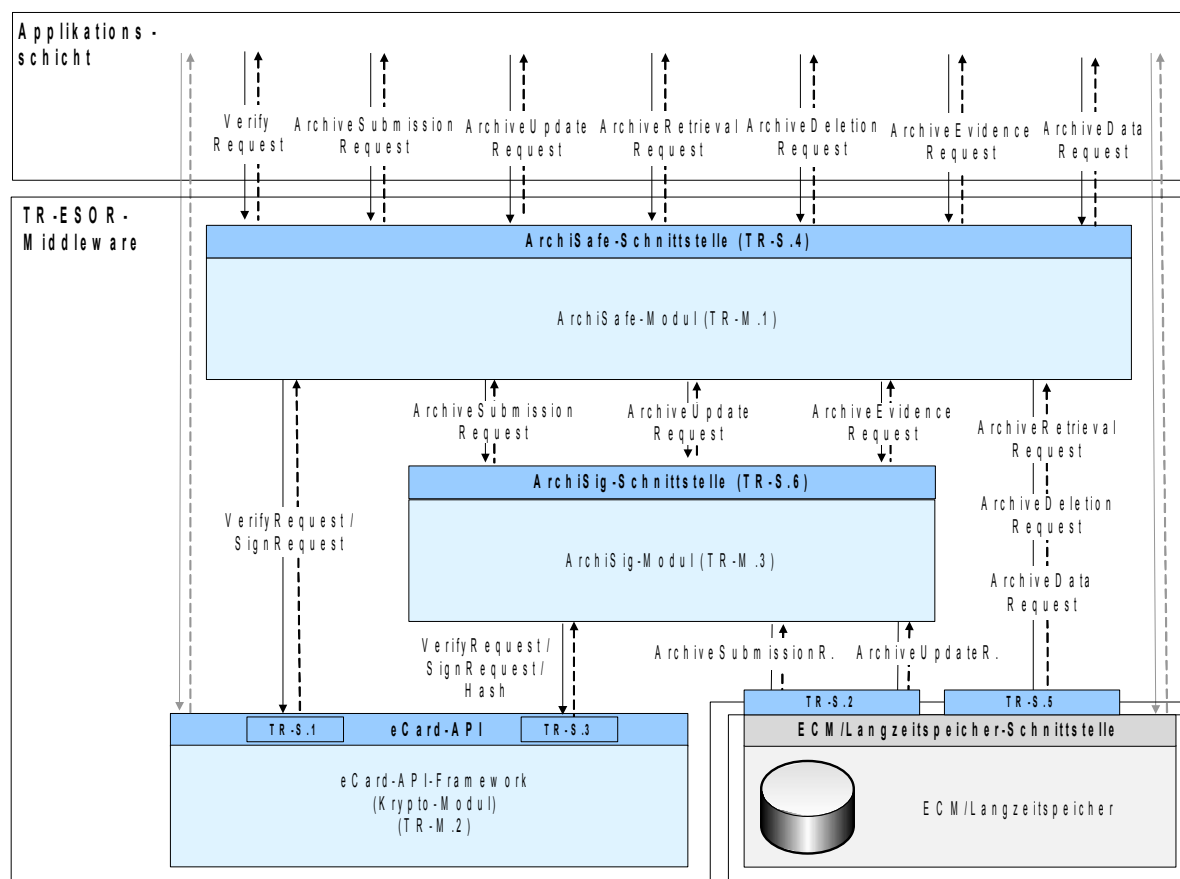
- ArchiveSubmissionRequest und ArchiveSubmissionResponse (siehe Abschnitt 3.1)
- ArchiveRetrievalRequest und ArchiveRetrievalResponse (siehe Abschnitt 3.3)
- ArchiveEvidenceRequest und ArchiveEvidenceResponse (siehe Abschnitt 3.4)
- ArchiveDeletionRequest und ArchiveDeletionResponse (siehe Abschnitt 3.5)

Darüber hinaus sollen die folgenden im vorliegenden Dokument näher aufgeführten Funktionen mit den hier beschriebenen Parameterkonstellationen unterstützt werden:

- ArchiveUpdateRequest und ArchiveUpdateResponse (siehe Abschnitt 3.2)
- ArchiveDataRequest und ArchiveDataResponse (siehe Abschnitt 3.6)
- VerifyRequest und VerifyResponse (siehe Abschnitt 3.7)

**(A2.0-3)** Sofern bei der Realisierung eines aus mindestens einem in dieser Richtlinie spezifizierten Modul bestehenden Produktes oder Systems die technische Konformität und Interoperabilität der *Konformitätsstufe 2* nachgewiesen werden soll, soll XAIP aus Anhang [TR-ESOR-F] als XML-Datenformat verwendet werden. Abweichungen im verwendeten XML-Datenformat sind zulässig, allerdings muss dann erläutert werden, wie eine gleichwertige Funktionalität realisiert wird. Insbesondere ist zu erläutern, wie eine Transformation in das XAIP Format aus Anhang F erfolgen kann.

**(A2.0-4)** Sofern bei der Realisierung eines aus mindestens einem in dieser Richtlinie spezifizierten Modul bestehenden Produktes oder Systems die technische Konformität und Interoperabilität der *Konformitätsstufe 2* nachgewiesen werden soll, muss das Basis-ERS-Profil aus Anhang [TR-ESOR-ERS] als Evidence-Record-Format verwendet werden.



**Abbildung 2: Umsetzung der IT-Referenzarchitektur auf Basis des eCard-API-Frameworks**

Wie in Abbildung 2 dargestellt, werden bei der vollständigen Umsetzung der IT-Referenzarchitektur auf Basis des eCard-API-Frameworks

1. die Schnittstellen des Krypto-Moduls gemäß des eCard-API-Frameworks (Technische Richtlinie des BSI TR 03112) realisiert und
2. auch die Schnittstellen des ArchiSafe-, ArchiSig- und ECM/Langzeitspeichers nutzen die gleichen grundlegenden Schnittstellentypen (`dss:RequestBaseType` und `dss:ResponseBaseType`) aus [OASIS-DSS], die auch bei den Signatur- und Verschlüsselungsfunktionen aus [eCard-2] genutzt werden.

Die URI-Fehlercodes in den Rückgaben der nicht bereits in der Technischen Richtlinie des BSI TR 03112 definierten Funktionen haben das Präfix <http://www.bsi.bund.de/tr-esor/api/1.2>, welches um entsprechende Bezeichner ergänzt wird. Dieser Namensraum ist in den visualisierten XML-Strukturen am Kürzel „tr“ erkennbar.

Während in den ASN.1-Strukturen [TR-ESOR-S] für die Angabe des verwendeten Protokolls bzw. der verwendeten API-Version stets ein explizites Element `Version` enthalten ist, kann bei den als XML-Schema definierten Strukturen (vgl. Abschnitt 6.1) und den darauf aufbauenden Webservice-Schnittstellen (vgl. Abschnitt 6.2) darauf verzichtet werden, da die Version der Struktur implizit durch den oben genannten Namensraum spezifiziert ist.

Außerdem werden jeweils statt der generischen `Controls`-Elemente in der ASN.1-Struktur die entsprechenden `dss:OptionalInputs`- und `dss:OptionalOutputs`-Elemente aus [OASIS-DSS] genutzt, die bei Bedarf auch weitere Elemente enthalten können.

Falls die in diesem Dokument beschriebenen Schnittstellen und Funktionen asynchron genutzt werden sollen, kann dies unter Verwendung der hierfür vorgesehenen Mechanismen aus [OASIS-Async] realisiert werden.

In den folgenden Abschnitten findet sich eine XML-basierte Spezifikation der verschiedenen in [TR-ESOR-S] eingeführten Funktionen zur Beweiswerterhaltung kryptographisch signierter



Dokumente. Hierbei werden die Funktionen der ArchiSafe-Schnittstelle (TR-S.4) in Abschnitt 3 spezifiziert. In Abschnitt 4 findet sich eine Beschreibung der internen Schnittstellen der TR-ESOR-Middleware, die auf die vorherige Spezifikation der Funktionen in Abschnitt 3 Bezug nimmt. In Abschnitt 5 sind die verwendeten Fehlercodes zusammengefasst und näher erläutert und in Abschnitt 6 finden sich schließlich die normativen XML-Schema- und WSDL-Spezifikationen für die in Abschnitt 3 spezifizierte ArchiSafe-Schnittstelle (TR-S.4).

### 3. Funktionen der ArchiSafe-Schnittstelle (TR-ESOR-S.4)

In diesem Abschnitt findet sich eine XML-basierte Spezifikation der in [TR-ESOR-S] eingeführten Funktionen der TR-ESOR-Middleware an der ArchiSafe-Schnittstelle **TR-ESOR-S.4 (TR-S.4)**:

- `ArchiveSubmissionRequest` und `ArchiveSubmissionResponse` (siehe Abschnitt 3.1)
- `ArchiveUpdateRequest` und `ArchiveUpdateResponse` (siehe Abschnitt 3.2)
- `ArchiveRetrievalRequest` und `ArchiveRetrievalResponse` (siehe Abschnitt 3.3)
- `ArchiveEvidenceRequest` und `ArchiveEvidenceResponse` (siehe Abschnitt 3.4)
- `ArchiveDeletionRequest` und `ArchiveDeletionResponse` (siehe Abschnitt 3.5)
- `ArchiveDataRequest` und `ArchiveDataResponse` (siehe Abschnitt 3.6)
- `VerifyRequest` und `VerifyResponse` (siehe Abschnitt 3.7)

Die graphische Darstellung der Schnittstellen in diesem Kapitel wurde - analog zur Spezifikation des eCard-API-Frameworks (siehe z.B. [eCard-2]) - mit einem XML-Viewer erstellt und dient lediglich der Veranschaulichung der XML-Strukturen. Die normative Spezifikation der Schnittstellen ist durch das XML-Schema bzw. die darauf aufbauende WSDL-Spezifikation (siehe Abschnitt 6) gegeben.


#### 3.1 ArchiveSubmissionRequest und ArchiveSubmissionResponse

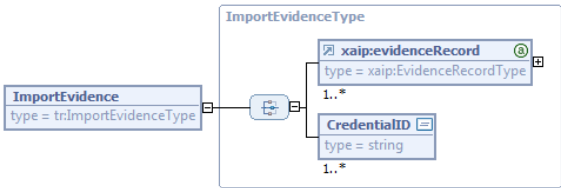
Mit der Funktion `ArchiveSubmissionRequest` wird dem aufgerufenen Modul ein Archivdatenobjekt (`xaip:XAIP`) zur Ablage übergeben und das aufrufende Modul erhält im Erfolgsfall in der `ArchiveSubmissionResponse` eine AOID zurück, mit der später wieder auf das archivierte Objekt oder die zugehörigen technischen Beweisdaten zugegriffen werden kann.

Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in den Schnittstellen TR-S.2 (vgl. Abschnitt 4.2) und TR-S.6 (vgl. Abschnitt 4.5) genutzt.

##### 3.1.1 ArchiveSubmissionRequest

Name	<b>ArchiveSubmissionRequest</b>	
Beschreibung	Mit der Funktion <code>ArchiveSubmissionRequest</code> wird dem aufgerufenen Modul ein Archivdatenobjekt übergeben.	
Aufruf	<p>Aufruf der <code>ArchiveSubmissionRequest</code>-Funktion</p>	
	Name	Beschreibung

	dss:OptionalInputs	<p>Ist für optionale Eingabelemente vorgesehen.</p> <p><b>(A3.1.1-1):</b> Gemäß der vorliegenden Spezifikation <u>sollen</u> folgende Elemente unterstützt werden:</p> <ul style="list-style-type: none"> <li>• AOID,</li> <li>• ReturnVerificationReport,</li> <li>• ImportEvidence.</li> </ul> <p>Dabei gilt:</p> <ul style="list-style-type: none"> <li>• AOID Durch die Übergabe eines AOID-Elementes kann die AOID von der aufrufenden Anwendung vergeben werden. Im Regelfall fehlt dieses Element und die AOID wird vom aufgerufenen Modul bereitgestellt.</li> </ul>  <ul style="list-style-type: none"> <li>• ReturnVerificationReport Durch die Übergabe eines ReturnVerificationReport-Elementes gemäß <b>[OASIS VR]</b> bzw. <b>[eCard-2]</b> kann ein ausführlicher Prüfbericht in Form eines VerificationReport-Elementes für die im XAIP-Element oder im unten genannten ImportEvidence-Element enthaltenen Signaturobjekte oder Beweisdaten angefordert werden. Bei einem übergebenen xaip:XAIP-Element wird im Details-Element des IndividualReport-Elementes des zurückgelieferten Prüfberichts (vgl. Abschnitt 3.3 in <b>[OASIS VR]</b>) ein XAIPReport-Element gemäß <b>[TR-ESOR-VR]</b> zurückgeliefert. Sofern kein xaip:XAIP sondern ein ArchiveData-Element und im ImportEvidence-Element (siehe unten) ein Evidence Record übergeben wird, wird für jeden übergebenen Evidence Record ein EvidenceRecordReport gemäß <b>[TR-ESOR-VR]</b> zurückgeliefert.</li> </ul>
--	--------------------	--

		<ul style="list-style-type: none"> <li>• ImportEvidence Mit der Übergabe des nachfolgend dargestellten ImportEvidence-Elementes kann der Import von einem oder mehreren zu einer bestimmten XAIP-Version bzw. zu den übergebenen Binärdaten gehörenden Evidence Records gemäß [RFC4998] oder [RFC6283]<sup>2</sup> angestoßen werden. Die Struktur des xaip:evidenceRecord-Elementes ist in [TR-ESOR-F] erläutert. Um Evidence Records für mehrere Versionen eines XAIPs importieren, zu können, <u>kann</u> dieses Element mehrmals auftreten. Das xaip:evidenceRecord-Element <u>muss</u> hier die Attribute AOID und VersionID enthalten. Sofern die zu importierenden Evidence Records bereits im XAIP enthalten sind, wird statt des Evidence Records hier die entsprechende CredentialID übergeben.</li> </ul>  <p><b>(A3.1.1-2):</b> Im Zuge des Imports von Evidence Records <u>müssen</u> diese von der TR-ESOR-Middleware vollständig geprüft werden. Dies umfasst die im entsprechenden ERS-Standard vorgesehenen Prüfungsschritte<sup>3</sup>, wobei die jeweiligen Zertifikate der Zeitstempel vollständig bis hin zu einer vertrauenswürdigen Wurzel geprüft werden <u>müssen</u>.</p>
	xaip:XAIP	Enthält ein XML-basiertes Archivdatenobjekt gemäß [TR-ESOR-F], das durch den Aufruf der beweiserhaltenden Archivierung zugeführt werden soll.
	ArchiveData	Enthält ein in einem beliebigen anderen Format vorliegendes Archivdatenobjekt. Der hierfür genutzte ArchiveDataType ist als anyType mit einem optionalen Type-Attribut definiert.

<sup>2</sup> [RFC4998] muss, [RFC6283] kann unterstützt werden.

<sup>3</sup> Siehe Abschnitt 3.3 in [RFC4998] und Abschnitt 2.3 in [RFC6283].

		<p>Durch das Type-Attribut <a href="http://www.bsi.bund.de/tr-esor/api/1.2/type/binaryData">http://www.bsi.bund.de/tr-esor/api/1.2/type/binaryData</a> wird klargestellt, dass im ArchiveData-Element ein Kindelement binaryData übergeben wird, das Base 64 codierte Nutzdaten und ein mimeType-Attribut enthält, die beim entsprechenden XAIP in ein dataObject-Element eingebettet werden.</p> <p>Weitere Übergabetypen <u>können</u> im Rahmen einer Profilierung der vorliegenden Spezifikation spezifiziert werden.</p>
--	--	---

### 3.1.2 ArchiveSubmissionResponse

<b>Name</b>	<b>ArchiveSubmissionResponse</b>							
<b>Beschreibung</b>	Als Antwort auf einen ArchiveSubmissionRequest wird ein entsprechendes ArchiveSubmissionResponse-Element zurückgeliefert, das im Erfolgsfall einen eindeutigen Identifikator des Archivdatenobjektes, die AOID, enthält.							
<b>Rückgabe</b>	<p>ArchiveSubmissionResponse ist die Antwort zum ArchiveSubmissionRequest-Aufruf</p> <table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 30%;">Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>dss:Result</td> <td>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in [eCard-1] und unten näher beschrieben.</td> </tr> <tr> <td>dss:OptionalOutputs</td> <td>                     Ist für optionale Ausgabeelemente vorgesehen.  <b>(A3.1.2-1):</b> Gemäß der vorliegenden Spezifikation kann das folgende Element auftreten:                     <ul style="list-style-type: none"> <li>VerificationReport gemäß [OASIS VR] bzw. [eCard-2] und [TR-ESOR-VR], der zurückgeliefert werden <u>muß</u>, sofern er explizit angefordert wurde oder bei der Prüfung der übergebenen Daten ein Fehler oder eine Warnung aufgetreten ist und deshalb als ResultMajor ein Fehlercode <a href="#">.../resultmajor#error</a> oder <a href="#">.../resultmajor#warning</a> zurückgeliefert wird.</li> </ul> </td> </tr> </tbody> </table>		Name	Beschreibung	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in [eCard-1] und unten näher beschrieben.	dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen. <b>(A3.1.2-1):</b> Gemäß der vorliegenden Spezifikation kann das folgende Element auftreten: <ul style="list-style-type: none"> <li>VerificationReport gemäß [OASIS VR] bzw. [eCard-2] und [TR-ESOR-VR], der zurückgeliefert werden <u>muß</u>, sofern er explizit angefordert wurde oder bei der Prüfung der übergebenen Daten ein Fehler oder eine Warnung aufgetreten ist und deshalb als ResultMajor ein Fehlercode <a href="#">.../resultmajor#error</a> oder <a href="#">.../resultmajor#warning</a> zurückgeliefert wird.</li> </ul>
Name	Beschreibung							
dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in [eCard-1] und unten näher beschrieben.							
dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen. <b>(A3.1.2-1):</b> Gemäß der vorliegenden Spezifikation kann das folgende Element auftreten: <ul style="list-style-type: none"> <li>VerificationReport gemäß [OASIS VR] bzw. [eCard-2] und [TR-ESOR-VR], der zurückgeliefert werden <u>muß</u>, sofern er explizit angefordert wurde oder bei der Prüfung der übergebenen Daten ein Fehler oder eine Warnung aufgetreten ist und deshalb als ResultMajor ein Fehlercode <a href="#">.../resultmajor#error</a> oder <a href="#">.../resultmajor#warning</a> zurückgeliefert wird.</li> </ul>							

Name	ArchiveSubmissionResponse	
	AOID	<u>Muss</u> , sofern die AOID vom aufgerufenen Modul erzeugt oder ergänzt wurde, vorhanden sein und für zukünftige Zugriffe auf das Archivdatenobjekt genutzt werden.
	<p>Statusinformationen und Fehler bei ArchiveSubmissionResponse (vgl. [eCard-1] Abschnitt 4.1 und 4.2).</p>	
Name	Fehlercode	
ResultMajor	<ul style="list-style-type: none"> <li>• <a href="#">/resultmajor#ok</a></li> <li>• <a href="#">/resultmajor#error</a></li> <li>• <a href="#">/resultmajor#warning</a></li> </ul>	
ResultMinor	<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/al/common#noPermission</a></li> <li>• <a href="#">/resultminor/al/common#internalError</a></li> <li>• <a href="#">/resultminor/al/common#parameterError</a></li> <li>• <a href="#">/resultminor/ar/lowSpaceWarning</a></li> <li>• <a href="#">/resultminor/ar/noSpaceError</a></li> <li>• <a href="#">/resultminor/ar/existingAOID</a></li> <li>• <a href="#">/resultminor/ar/notSupported</a></li> <li>• <a href="#">/resultminor/ar/unknownArchiveDataType</a></li> <li>• <a href="#">/resultminor/ar/XAIP_NOK</a></li> <li>• <a href="#">/resultminor/ar/XAIP_NOK_EXPIRED</a></li> <li>• <a href="#">/resultminor/ar/XAIP_NOK_SUBMTIME</a></li> <li>• <a href="#">/resultminor/ar/XAIP_NOK_SIG</a></li> <li>• <a href="#">/resultminor/ar/XAIP_NOK_ER</a></li> </ul>	

### 3.2 ArchiveUpdateRequest und ArchiveUpdateResponse

Mit der Funktion `ArchiveUpdateRequest` kann eine neue Version für ein bereits abgelegtes Archivdatenobjekt erzeugt werden. Hierbei werden die bereits abgelegten Daten nicht verändert, sondern es wird lediglich zusätzlich eine neue Version hinzugefügt.

Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in TR-S.2 (vgl. Abschnitt 4.2) und TR-S.6 (vgl. Abschnitt 4.5) genutzt.

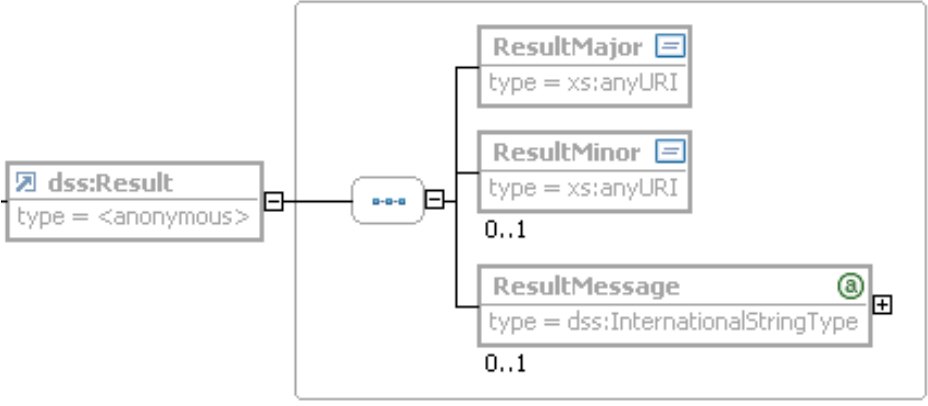
### 3.2.1 ArchiveUpdateRequest

<b>Name</b>	<b>ArchiveUpdateRequest</b>	
<b>Beschreibung</b>	Mit der Funktion <code>ArchiveUpdateRequest</code> wird eine neue Version für ein bereits abgelegtes Archivdatenobjekt erzeugt (vgl. [TR-ESOR-M.1]).	
	<p>Aufruf der <code>ArchiveUpdateRequest</code>-Funktion</p>	
	<b>Name</b>	<b>Beschreibung</b>
	<code>dss:OptionalInputs</code>	Ist für optionale Eingabeelemente vorgesehen. <b>(A3.2.1-1):</b> Gemäß der vorliegenden Spezifikation <u>sollen</u> hier die auf Seite 11 spezifizierten optionalen Eingabeelemente <code>AOID</code> , <code>ReturnVerificationReport</code> und <code>ImportEvidence</code> unterstützt werden.
	<code>xaip:DXAIP</code>	Enthält ein ergänzendes XML-basiertes Archivdatenobjekt (Delta-XAIP) gemäß [TR-ESOR-F], das ein neues <code>versionManifest</code> , die Vorgängerversion, Verweise auf unverändert aus dieser übernommene Objekte und die zu ergänzenden Elemente enthält, die in einer neuen Version eines bereits abgelegten Archivdatenobjektes ergänzt werden sollen.

### 3.2.2 ArchiveUpdateResponse

<b>Name</b>	<b>ArchiveUpdateResponse</b>									
<b>Beschreibung</b>	Als Antwort auf einen ArchiveUpdateRequest wird ein entsprechendes ArchiveUpdateResponse-Element zurückgeliefert, das im Erfolgsfall einen im Kontext einer AOID eindeutigen Identifikator der neuen Version des Archivdatenobjektes, die VersionID, enthält.									
<b>Rückgabe</b>	<div style="text-align: center;"> </div> <p>ArchiveUpdateResponse ist die Antwort zum ArchiveUpdateRequest-Aufruf</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%; text-align: left;">Name</th> <th style="width: 50%; text-align: left;">Beschreibung</th> </tr> </thead> <tbody> <tr> <td>dss:Result</td> <td>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in <b>[eCard-1]</b> und unten näher beschrieben.</td> </tr> <tr> <td>dss:OptionalOutputs</td> <td>Ist für optionale Ausgabeelemente vorgesehen. <b>(A3.2.2-1):</b> Gemäß der vorliegenden Spezifikation kann das folgende Element auftreten: <ul style="list-style-type: none"> <li>VerificationReport gemäß <b>[OASIS VR]</b> bzw. <b>[eCard-2]</b> und <b>[TR-ESOR-VR]</b>, der zurückgeliefert werden <u>muss</u>, sofern er explizit angefordert wurde oder bei der Prüfung der übergebenen Daten ein Fehler oder eine Warnung aufgetreten ist und deshalb als ResultMajor ein Fehlercode <a href="#">.../resultmajor#error</a> oder <a href="#">.../resultmajor#warning</a> zurückgeliefert wird.</li> </ul> </td> </tr> <tr> <td>VersionID</td> <td>Ist im Erfolgsfall vorhanden und enthält den bezüglich des über die AOID identifizierten Archivdatenobjektes eindeutigen Versions-Identifikator. Die VersionID <u>soll</u> in der Form v1, v2, ... vx gebildet werden.</td> </tr> </tbody> </table>		Name	Beschreibung	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in <b>[eCard-1]</b> und unten näher beschrieben.	dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen. <b>(A3.2.2-1):</b> Gemäß der vorliegenden Spezifikation kann das folgende Element auftreten: <ul style="list-style-type: none"> <li>VerificationReport gemäß <b>[OASIS VR]</b> bzw. <b>[eCard-2]</b> und <b>[TR-ESOR-VR]</b>, der zurückgeliefert werden <u>muss</u>, sofern er explizit angefordert wurde oder bei der Prüfung der übergebenen Daten ein Fehler oder eine Warnung aufgetreten ist und deshalb als ResultMajor ein Fehlercode <a href="#">.../resultmajor#error</a> oder <a href="#">.../resultmajor#warning</a> zurückgeliefert wird.</li> </ul>	VersionID	Ist im Erfolgsfall vorhanden und enthält den bezüglich des über die AOID identifizierten Archivdatenobjektes eindeutigen Versions-Identifikator. Die VersionID <u>soll</u> in der Form v1, v2, ... vx gebildet werden.
Name	Beschreibung									
dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in <b>[eCard-1]</b> und unten näher beschrieben.									
dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen. <b>(A3.2.2-1):</b> Gemäß der vorliegenden Spezifikation kann das folgende Element auftreten: <ul style="list-style-type: none"> <li>VerificationReport gemäß <b>[OASIS VR]</b> bzw. <b>[eCard-2]</b> und <b>[TR-ESOR-VR]</b>, der zurückgeliefert werden <u>muss</u>, sofern er explizit angefordert wurde oder bei der Prüfung der übergebenen Daten ein Fehler oder eine Warnung aufgetreten ist und deshalb als ResultMajor ein Fehlercode <a href="#">.../resultmajor#error</a> oder <a href="#">.../resultmajor#warning</a> zurückgeliefert wird.</li> </ul>									
VersionID	Ist im Erfolgsfall vorhanden und enthält den bezüglich des über die AOID identifizierten Archivdatenobjektes eindeutigen Versions-Identifikator. Die VersionID <u>soll</u> in der Form v1, v2, ... vx gebildet werden.									



Name	<b>ArchiveUpdateResponse</b>	
	 <p>Statusinformationen und Fehler bei ArchiveUpdateResponse (vgl. [eCard-1] Abschnitt 4.1 und 4.2).</p>	
	<b>Name</b>	<b>Fehlercode</b>
	ResultMajor	<ul style="list-style-type: none"> <li>• <a href="#">/resultmajor#ok</a></li> <li>• <a href="#">/resultmajor#error</a></li> <li>• <a href="#">/resultmajor#warning</a></li> </ul>
	ResultMinor	<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/al/common#noPermission</a></li> <li>• <a href="#">/resultminor/al/common#internalError</a></li> <li>• <a href="#">/resultminor/al/common#parameterError</a></li> <li>• <a href="#">/resultminor/ar/lowSpaceWarning</a></li> <li>• <a href="#">/resultminor/ar/noSpaceError</a></li> <li>• <a href="#">/resultminor/ar/existingPackageInfoWarning</a></li> <li>• <a href="#">/resultminor/ar/notSupported</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK_AOID</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK_EXPIRED</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK_SUBMTIME</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK_SIG</a></li> <li>• <a href="#">/resultminor/ar/XAIP_NOK_ER</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK_ID</a></li> <li>• <a href="#">/resultminor/ar/DXAIP_NOK_Version</a></li> </ul>

### 3.3 ArchiveRetrievalRequest und ArchiveRetrievalResponse

Mit der Funktion ArchiveRetrievalRequest kann das zu einer übergebenen AOID gehörende Archivdatenobjekt im XAIP-Format gemäß [TR-ESOR-F] über die TR-ESOR-Middleware aus dem ECM-/Langzeitspeichersystem ausgelesen werden.

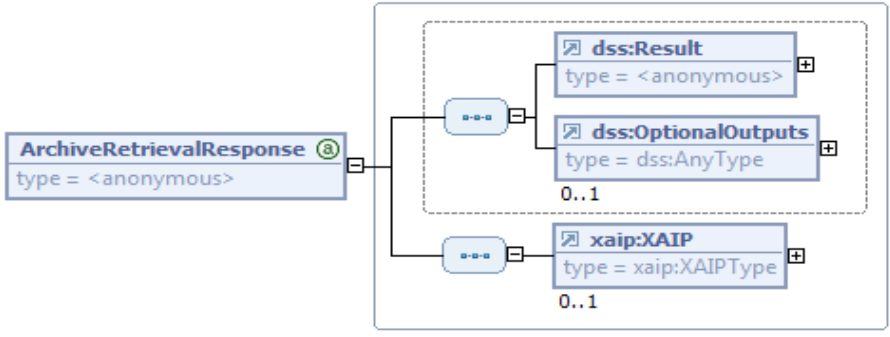
Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in den Schnittstellen S.2 (vgl. Abschnitt 4.2) und S.5 (vgl. Abschnitt 4.4) genutzt.

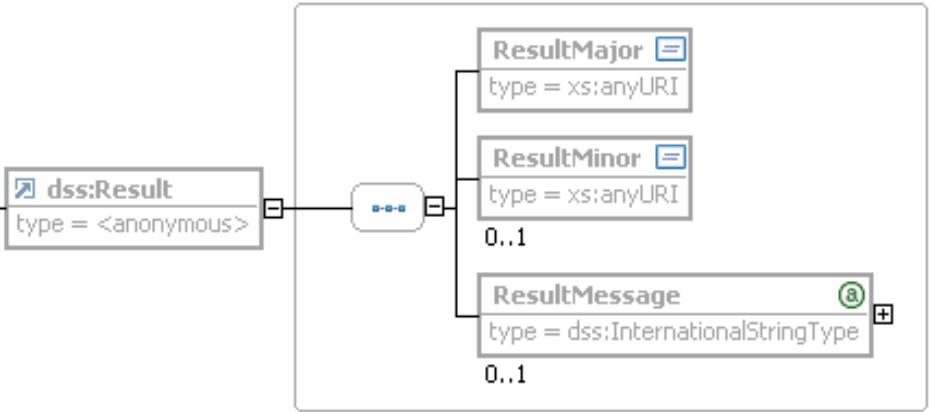
### 3.3.1 ArchiveRetrievalRequest

<b>Name</b>	<b>ArchiveRetrievalRequest</b>	
<b>Beschreibung</b>	Mit der Funktion ArchiveRetrievalRequest wird ein im Langzeitspeicher abgelegtes Archivdatenobjekt (xaip:XAIP) ausgelesen und zurückgeliefert.	
<b>Beschreibung</b>	<p>Aufruf der ArchiveRetrievalRequest-Funktion</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:OptionalInputs	<p>Ist für optionale Eingabeelemente vorgesehen.</p> <p><b>(A3.3.1-1):</b> Gemäß der vorliegenden Spezifikation <u>soll</u> das folgende optionale Eingabeelement unterstützt werden: IncludeERS.</p> <p>IncludeERS – gibt an, dass das zurückgelieferte XAIP den bzw. die entsprechenden Evidence Record im angegebenen Format (vgl. ERSFormat, Seite 22) enthalten sollen.</p> <p>Dieser bzw. diese Evidence Record(s) wird bzw.</p> <p>werden im dafür vorgesehenen xaip:credential/ xaip:EvidenceRecord Element zurückgeliefert.</p> <p><b>(A3.3.1-2):</b> Das VersionID-Attribut des xaip:EvidenceRecord Elementes muss auf die entsprechende Version verweisen.</p> <p>Sofern das versionManifest nicht kryptographisch geschützt ist, <u>muss</u> mit einem unprotectedObjectPointer Element im entsprechenden versionManifest auf die credentialID des xaip:credential-Elementes verwiesen werden. Umgekehrt <u>muss</u> auf die vom Evidence Record geschützten Datenobjekte im relatedObjects-Attribut des xaip:credential-Elementes verwiesen</p>

Name	ArchiveRetrievalRequest	
	AOID	Enthält den eindeutigen Identifikator des angeforderten Archivdatenobjektes.
	VersionID	<p><u>Kann</u> eine Folge von Versions-Identifikatoren enthalten, durch die angegeben wird welche Versionen des Archivdatenobjektes genau zurückgeliefert werden sollen.</p> <p>Sofern das VersionID-Element nicht angegeben ist, werden die zur letzten Version gehörigen Datenobjekte und Verwaltungsinformationen zurückgeliefert.</p> <p>Durch die Angabe von all werden alle existierenden Versionen eines Archivdatenobjektes zurückgeliefert.</p>

### 3.3.2 ArchiveRetrievalResponse

Name	ArchiveRetrievalResponse	
<b>Beschreibung</b>	Als Antwort auf einen ArchiveRetrievalRequest wird ein entsprechendes ArchiveRetrievalResponse-Element zurückgeliefert, welches im Erfolgsfall das angeforderte Archivdatenobjekt im xaip:Xaip-Format gemäß [TR-ESOR-F] enthält.	
<b>Rückgabe</b>	 <p>ArchiveRetrievalResponse ist die Antwort zum ArchiveRetrievalRequest-Aufruf</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:Result	<p>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in [eCard-1] und weiter unten näher beschrieben.</p> <p>Sofern nur ein Teil der angeforderten Versionen des Archivdatenobjektes zurückgeliefert werden konnte, wird dies durch den Fehlercode <a href="#">/resultminor/arl/requestOnlyPartlySuccessfulWarning</a> angezeigt.</p>

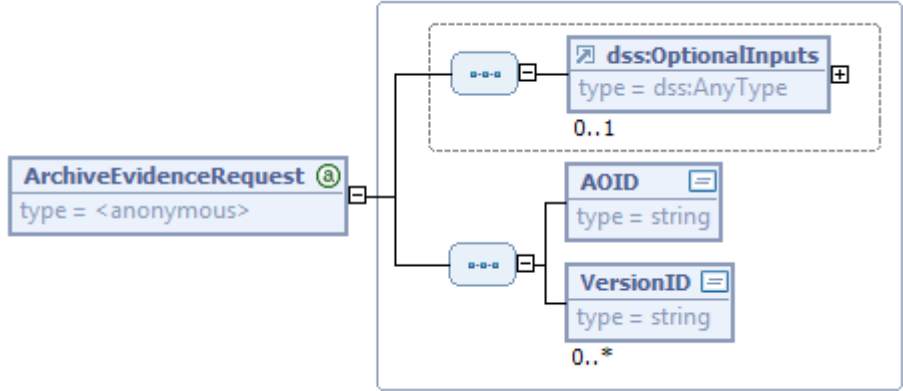
Name	ArchiveRetrievalResponse	
	dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen und <u>kann</u> beispielsweise entsprechende Steuerelemente ( <code>responseControls</code> ) enthalten, die im Rahmen einer Profilierung der vorliegenden Spezifikation definiert werden <u>sollen</u> .
	xaip:XAIP	Sofern kein Fehler aufgetreten ist, wird das angeforderte XML-basierte Archivdatenobjekt (XAIP) gemäß [TR-ESOR-F] zurückgeliefert.
	 <p data-bbox="496 1173 1382 1240">Statusinformationen und Fehler bei ArchiveRetrievalResponse (vgl. [eCard-1]).</p>	
	Name	Fehlercode
	ResultMajor	<ul style="list-style-type: none"> <li>• <a href="#">/resultmajor#ok</a></li> <li>• <a href="#">/resultmajor#error</a></li> <li>• <a href="#">/resultmajor#warning</a></li> </ul>
	ResultMinor	<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/al/common#noPermission</a></li> <li>• <a href="#">/resultminor/al/common#internalError</a></li> <li>• <a href="#">/resultminor/al/common#parameterError</a></li> <li>• <a href="#">/resultminor/ar/unknownAOID</a></li> <li>• <a href="#">/resultminor/ar/unsupported</a></li> <li>• <a href="#">/resultminor/ar/requestOnlyPartlySuccessfulWarning</a></li> <li>• <a href="#">/resultminor/ar/unknownVersionID</a></li> </ul>
	ResultMessage	Beim Auftreten der Fehlermeldung <code>.../unknownVersionID</code> <u>soll</u> die problematische <code>VersionID</code> hier zurückgeliefert werden.

### 3.4 ArchiveEvidenceRequest und ArchiveEvidenceResponse

Mit der Funktion `ArchiveEvidenceRequest` können die zugehörigen technischen Beweisdaten (Evidence Records gemäß [RFC4998] oder [RFC6283]<sup>4</sup>) für beweiswerterhaltend aufbewahrte und über AOID-Elemente adressierte Archivdatenobjekte (`xaip:XAIP`) zurückgeliefert werden.


Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in TR-S.6 (vgl. Abschnitt 4.5) genutzt.

#### 3.4.1 ArchiveEvidenceRequest

<b>Name</b>	<b>ArchiveEvidenceRequest</b>	
<b>Beschreibung</b>	Mit der Funktion <code>ArchiveEvidenceRequest</code> können für beweiswerterhaltend abgelegte Archivdatenobjekte technische Beweisdaten in Form von Evidence Records gemäß [RFC4998] oder [RFC6283] <sup>5</sup> angefordert werden.	
<b>Beschreibung</b>	 <p>AUFRUF DER <code>ArchiveEvidenceRequest</code>-FUNKTION</p>	
	<b>Name</b>	<b>Beschreibung</b>

<sup>4</sup> [RFC4998] muss, [RFC6283] kann unterstützt werden.

<sup>5</sup> [RFC4998] muss, [RFC6283] kann unterstützt werden.

Name	ArchiveEvidenceRequest	
	dss:OptionalInputs	<p>Ist für optionale Ausgabeelemente vorgesehen. <b>(A3.4.1-1)</b>: Gemäß der vorliegenden Spezifikation <u>soll</u> das folgende Element unterstützt werden:</p> <div data-bbox="1050 409 1238 483" style="border: 1px solid black; padding: 2px; width: fit-content; margin: 10px auto;"><b>ERSFormat</b>  type = anyURI</div> <p>Mit dem Element <code>tr:ERSFormat</code> vom Typ <code>anyURI</code> kann das gewünschte Format der zurückgelieferten Evidence Records angegeben werden, wobei folgende URIs vorgesehen sind:</p> <ul style="list-style-type: none"><li>• <a href="urn:ietf.rfc:4998">urn:ietf.rfc:4998</a> für ASN.1-basierte Evidence Records gemäß <b>[RFC4998]</b> oder</li><li>• <a href="urn:ietf.rfc:6283">urn:ietf.rfc:6283</a> für XML-basierte Evidence Records gemäß <b>[RFC6283]</b>.</li></ul> <p>Fehlt das <code>ERSFormat</code>-Element, so werden ASN.1-basierte Evidence Records gemäß <b>[RFC4998]</b> zurückgeliefert.</p>

Name	ArchiveEvidenceRequest	
	AOID	Ist der eindeutige Identifikator des angeforderten Archivdatenobjektes.
	VersionID	<p><u>Kann</u> mehrfach auftreten und angeben für welche Versionen eines über die AOID identifizierten Archivdatenobjektes Evidence Records zurückgeliefert werden sollen.</p> <p>Sofern das VersionID-Element nicht angegeben ist, wird der Beweisdatensatz für die aktuelle Version des XAIP zurückgeliefert.</p> <p>Durch die Angabe von all werden Evidence Records für alle existierenden Versionen eines Archivdatenobjektes zurückgeliefert.</p>

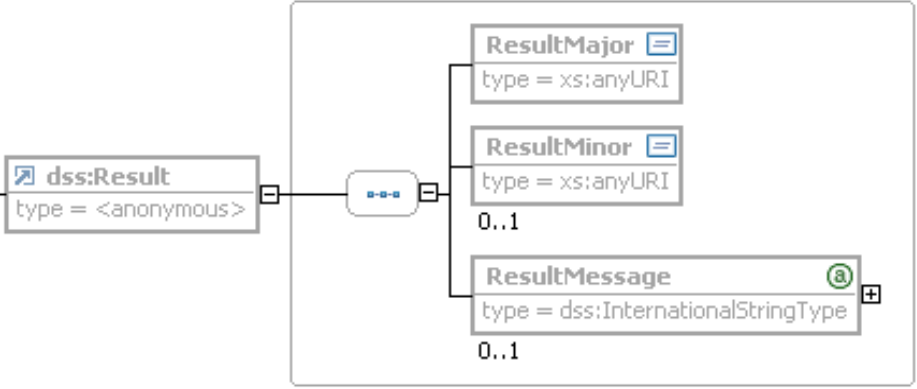
### 3.4.2 ArchiveEvidenceResponse

Name	ArchiveEvidenceResponse	
<b>Beschreibung</b>	Als Antwort auf einen ArchiveEvidenceRequest wird ein entsprechendes ArchiveEvidenceResponse-Element zurückgeliefert, das die angeforderten Beweisdaten enthält.	
<b>Rückgabe</b>	<p>ArchiveEvidenceResponse ist die Antwort zum ArchiveEvidenceRequest-Aufruf</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:Result	<p>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in Abschnitt 4.1.2 von [eCard-1] und unten näher beschrieben.</p> <p>Sofern nicht für alle mittels der übergebenen AOID adressierten Archivdatenobjekte entsprechende Beweisdaten (Evidence Records) zurückgeliefert werden konnten, wird dies durch die <a href="#">/resultminor/arl/requestOnlyPartlySuccessfulWarning</a> angezeigt.</p>

Name	ArchiveEvidenceResponse	
	dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen und <u>kann</u> beispielsweise entsprechende Steuerelemente (responseControls) enthalten, die im Rahmen einer Profilierung der vorliegenden Spezifikation definiert werden <u>sollen</u> .
	xaip:evidenceRecord	Sofern vom ArchiSig-Modul entsprechende Evidence Records <sup>6</sup> gemäß [RFC4998] bzw. [RFC6283] konstruiert werden können, werden diese hier zurückgeliefert. Die detaillierte Struktur dieses Elementes ist nachfolgend erläutert.
	<p>Das xaip:evidenceRecord-Element gemäß [TR-ESOR-F] ist vom Typ <b>xaip:EvidenceRecordType</b>, der als Erweiterung des <b>ec:EvidenceRecordType</b> aus [eCard-2] definiert ist und zusätzlich die Attribute AOID und VersionID, enthält, die in [TR-ESOR-F] näher erläutert sind.</p> <p><b>(A3.4.2-1):</b> Bei der hier beschriebenen Verwendung von xaip:evidenceRecord <u>müssen</u> die Attribute AOID und VersionID gesetzt sein.</p>	
	<b>Name</b>	<b>Beschreibung</b>
	xmlEvidenceRecord	Enthält einen XML-basierten Evidence Record gemäß [RFC6283].
	asn1EvidenceRecord	Enthält einen ASN.1-basierten Evidence Record gemäß [RFC4998].

<sup>6</sup> Sofern die TR-ESOR-Middleware mehrere redundante Hashbäume pflegt, werden hier mehrere Evidence Records zurückgeliefert.



<b>Name</b>	<b>ArchiveEvidenceResponse</b>	
	 <p>Statusinformationen und Fehler bei ArchiveEvidenceResponse (vgl. [eCard-1]).</p>	
	<b>Name</b>	<b>Fehlercode</b>
	ResultMajor	<ul style="list-style-type: none"> <li>• <a href="#">/resultmajor#ok</a></li> <li>• <a href="#">/resultmajor#error</a></li> <li>• <a href="#">/resultmajor#warning</a></li> </ul>
	ResultMinor	<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/al/common#noPermission</a></li> <li>• <a href="#">/resultminor/al/common#internalError</a></li> <li>• <a href="#">/resultminor/al/common#parameterError</a></li> <li>• <a href="#">/resultminor/ar1/notSupported<sup>7</sup></a></li> <li>• <a href="#">/resultminor/ar1/unknownAOID</a></li> <li>• <a href="#">/resultminor/ar1/unknownVersionID/</a></li> <li>• <a href="#">resultminor/ar1/requestOnlyPartlySuccessfulWarning</a></li> </ul>

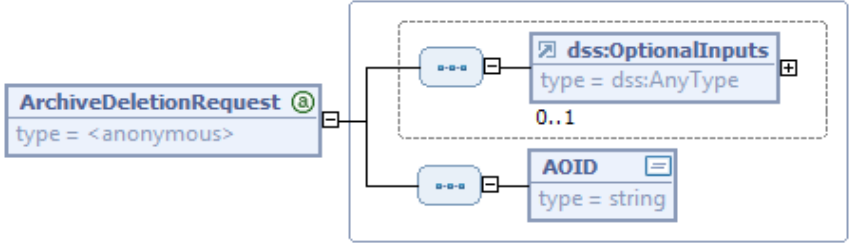
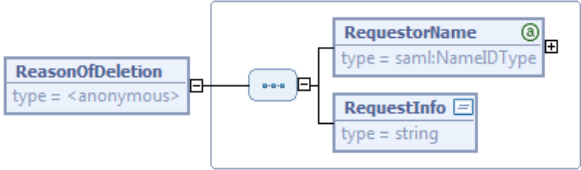
### 3.5 ArchiveDeletionRequest und ArchiveDeletionResponse

Mit der Funktion `ArchiveDeletionRequest` kann ein Archivdatenobjekt über die TR-ESOR-Middleware aus dem ECM-/Langzeitspeichersystem gelöscht werden.

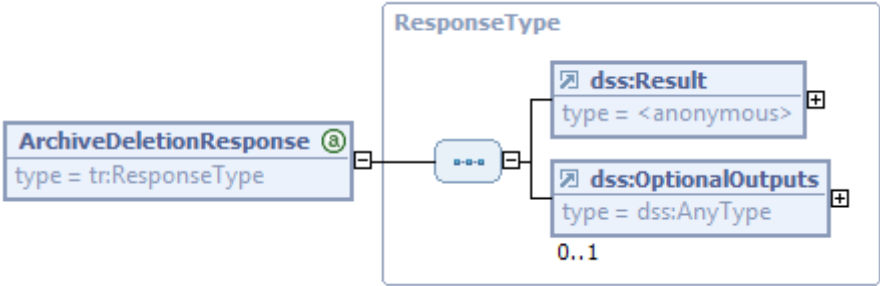
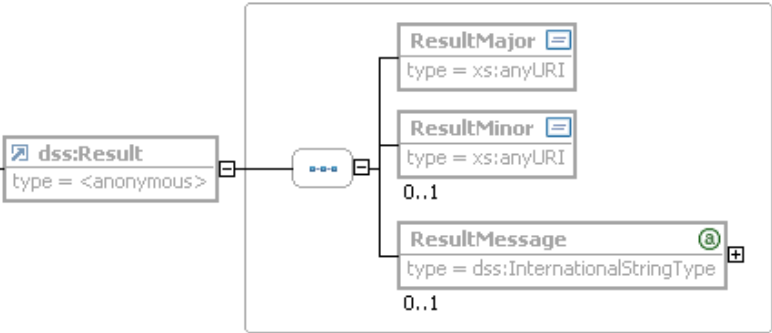
Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in der Schnittstelle TR-S.5 (vgl. Abschnitt 4.4) genutzt.

<sup>7</sup> Im `ResultMessage`-Element sollen nähere Informationen darüber zurückgeliefert werden, welche angeforderte Funktionalität nicht unterstützt wird.

## 3.5.1 ArchiveDeletionRequest

<b>Name</b>	<b>ArchiveDeletionRequest</b>	
<b>Beschreibung</b>	Mit der Funktion ArchiveDeletionRequest kann ein im Langzeitspeicher abgelegtes Archivdatenobjekt (xaip:XAIP) gelöscht werden.	
<b>Beschreibung</b>	<p>Aufruf der ArchiveDeletionRequest-Funktion</p>  <p>The diagram shows the structure of the ArchiveDeletionRequest. It consists of a main container 'ArchiveDeletionRequest' with type '&lt;anonymous&gt;' and a required flag (Ⓜ). This container has two inputs: a dashed box labeled 'dss:OptionalInputs' with type 'dss:AnyType' and cardinality '0..1', and an 'AOID' element with type 'string'. The 'dss:OptionalInputs' box contains three elements, each with a required flag (Ⓜ): 'RequestorName' (type 'saml:NameIDType'), 'RequestInfo' (type 'string'), and 'ReasonOfDeletion' (type '&lt;anonymous&gt;').</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:OptionalInputs	<p>Ist für optionale Eingabeelemente vorgesehen. Insbesondere bei einer vorzeitigen Löschung <u>mus</u>s das folgende Element ReasonOfDeletion genutzt und unterstützt werden:</p> <p><b>(A3.5.1-1):</b> Das ReasonOfDeletion-Element <u>mus</u>s vorhanden sein, sofern die Aufbewahrungsdauer der letzten Version noch nicht abgelaufen ist, und enthält neben dem Namen der aufrufenden Instanz auch eine Begründung für die Löschung.</p> <p><b>(A3.5.1-2):</b> Die gesamte Aktion einschließlich der Begründung <u>mus</u>s protokolliert werden und der übergebene RequestorName <u>sol</u>l mit den verwendeten Authentisierungsinformationen abgeglichen werden.</p>  <p>The diagram shows the structure of the ReasonOfDeletion element. It has type '&lt;anonymous&gt;' and a required flag (Ⓜ). It contains three elements, each with a required flag (Ⓜ): 'RequestorName' (type 'saml:NameIDType'), 'RequestInfo' (type 'string'), and 'ReasonOfDeletion' (type '&lt;anonymous&gt;').</p>
	AOID	Das AOID-Element gibt an, welches Archivdatenobjekt gelöscht werden soll.

### 3.5.2 ArchiveDeletionResponse

<b>Name</b>	<b>ArchiveDeletionResponse</b>	
<b>Beschreibung</b>	Als Antwort auf einen ArchiveDeletionRequest wird ein entsprechendes ArchiveDeletionResponse-Element zurückgeliefert, das Informationen über den Erfolg oder Misserfolg der Anfrage enthält.	
<b>Rückgabe</b>	 <p>ArchiveDeletionResponse ist die Antwort zum ArchiveDeletionRequest-Aufruf</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in <b>[eCard-1]</b> und unten näher beschrieben.
	dss:OptionalOutputs	Ist für optionale Ausgabeelemente vorgesehen und <u>kann</u> beispielsweise entsprechende Steuerelemente ( <code>responseControls</code> ) enthalten, die im Rahmen einer Profilierung der vorliegenden Spezifikation definiert werden <u>sollen</u> .
	 <p>Statusinformationen und Fehler bei ArchiveDeletionResponse (vgl. <b>[eCard-1]</b>).</p>	
	<b>Name</b>	<b>Fehlercode</b>
	ResultMajor	<ul style="list-style-type: none"> <li>• <a href="#">/resultmajor#ok</a></li> <li>• <a href="#">/resultmajor#error</a></li> </ul>
	ResultMinor	<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/al/common#noPermission</a></li> <li>• <a href="#">/resultminor/al/common#internalError</a></li> <li>• <a href="#">/resultminor/al/common#parameterError</a></li> </ul>

<b>Name</b>	<b>ArchiveDeletionResponse</b>	
		<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/arl/unknownAOID</a></li> <li>• <a href="#">/resultminor/arl/notSupported</a></li> <li>• <a href="#">/resultminor/arl/missingReasonOfDeletion</a></li> </ul>

### 3.6 ArchiveDataRequest und ArchiveDataResponse

Mit der Funktion `ArchiveDataRequest` können diskrete Datenelemente aus einem bereits abgelegten Archivdatenobjekt (`xaip:XAIP`) ausgelesen werden.

Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in TR-S.5 (vgl. Abschnitt 4.4) genutzt.

#### 3.6.1 ArchiveDataRequest

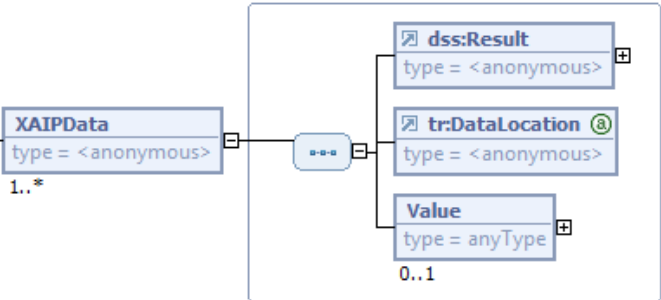
<b>Name</b>	<b>ArchiveDataRequest</b>	
<b>Beschreibung</b>	Mit der Funktion <code>ArchiveDataRequest</code> können diskrete Datenelemente aus einem im ECM-/Langzeitspeichersystem abgelegten, zumindest logisch im <code>xaip:XAIP</code> -Format gemäß [TR-ESOR-F] vorliegenden, Archivdatenobjekt ausgelesen werden.	
<b>Beschreibung</b>	<p>Aufruf der <code>ArchiveDataRequest</code>-Funktion</p>	
	<b>Name</b>	<b>Beschreibung</b>
	<code>dss:OptionalInputs</code>	Ist für optionale Eingabeelemente vorgesehen und <u>kann</u> beispielsweise Steuerelemente ( <code>requestControls</code> ) enthalten, die im Rahmen einer Profilierung der vorliegenden Spezifikation definiert werden <u>sollen</u> . Die vorliegende Spezifikation definiert keine solchen optionalen Eingabeelemente.
	<code>AOID</code>	Dieses Element enthält den Identifikator eines bestimmten Archivdatenobjektes.
	<code>tr:DataLocation</code>	Das <code>tr:DataLocation</code> -Element kann mehrmals auftreten und bestimmt die „Lokation“ der auszulesenden diskreten Datenelemente bezüglich eines zumindest logisch im <code>xaip:XAIP</code> -Format gemäß [TR-ESOR-F] vorliegenden Archivdatenobjektes. <sup>8</sup>

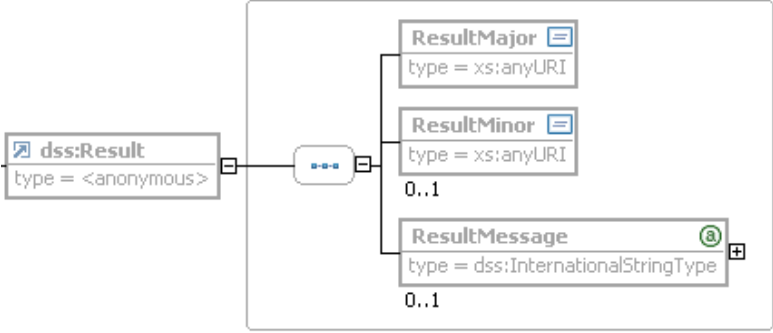
<b>Name</b>	<b>ArchiveDataRequest</b>
	<p>Das DataLocation-Element spezifiziert, welche Teile eines Archivobjektes zurückgeliefert werden sollen und ist folgendermaßen definiert:</p> <pre>&lt;element name="DataLocation"&gt;   &lt;complexType&gt;     &lt;complexContent&gt;       &lt;extension base="anyType"&gt;         &lt;attribute name="Type" type="anyURI"/&gt;       &lt;/extension&gt;     &lt;/complexContent&gt;   &lt;/complexType&gt; &lt;/element&gt;</pre> <p>Im Type-Attribut wird angegeben, welche Transformation für den Zugriff auf die gewünschten Daten angewandt werden soll, wobei die folgenden URIs vorgesehen sind:</p> <ul style="list-style-type: none"> <li>• <a href="http://www.w3.org/TR/2007/REC-xpath20-20070123/">http://www.w3.org/TR/2007/REC-xpath20-20070123/</a> für XPath,</li> <li>• <a href="http://www.w3.org/TR/2007/REC-xquery-20070123/">http://www.w3.org/TR/2007/REC-xquery-20070123/</a> für XQuery und</li> <li>• <a href="http://www.w3.org/TR/2003/REC-xptr-framework-20030325/">http://www.w3.org/TR/2003/REC-xptr-framework-20030325/</a> für XPointer</li> </ul>

### 3.6.2 ArchiveDataResponse

<b>Name</b>	<b>ArchiveDataResponse</b>	
<b>Beschreibung</b>	Als Antwort auf einen ArchiveDataRequest wird ein entsprechendes ArchiveDataResponse-Element zurückgeliefert, das die gewünschten Informationen enthält.	
<b>Rückgabe</b>	<p>ArchiveDataResponse ist die Antwort zum ArchiveDataRequest-Aufruf</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements ist in <b>[eCard-1]</b> und unten näher beschrieben. Sofern nur ein Teil der angefragten diskreten Datenobjekte zurückgeliefert werden konnte, wird dies durch den Fehlercode <a href="#">.../resultminor/arl/requestOnlyPartlySuccessfulWarning</a> angezeigt.

<sup>8</sup> Ausgehend von einem XML-basierten Archivdatenobjekt bieten sich für die diskrete Adressierung von XML-Datenelementen hier XPath (siehe unter: <http://www.w3.org/TR/2007/REC-xpath20-20070123/>), XQuery (siehe unter: <http://www.w3.org/TR/2007/REC-xquery-20070123/>) oder die XML Pointer Language XPointer (siehe unter: <http://www.w3.org/TR/2003/REC-xptr-framework-20030325/>) an.

Name	ArchiveDataResponse	
dss:OptionalOutputs		Ist für optionale Ausgabeelemente vorgesehen und <u>kann</u> beispielsweise entsprechende Steuerelemente (responseControls) enthalten, die im Rahmen einer Profilierung der vorliegenden Spezifikation definiert werden <u>sollen</u> .
XAIPData		Enthält im Erfolgsfall die gewünschten Daten und die „Lokation“, aus der diese aus der im ECM-/Langzeitspeichersystem zumindest logisch existierenden XAIP-Struktur ausgelesen wurden. Die detaillierte Struktur dieses Elementes ist nachfolgend dargestellt und erläutert.
		Das XAIPData-Element enthält im Erfolgsfall die gewünschten Daten.
Name	Beschreibung	
dss:Result		<p>Gibt an, ob die Anfrage erfolgreich durchgeführt werden konnte oder nicht.</p> <p>Als ResultMajor sind die beiden folgenden Werte möglich:</p> <ul style="list-style-type: none"> <li>• <a href="#">.../resultmajor#ok</a></li> <li>• <a href="#">.../resultmajor#error</a></li> </ul> <p>Als ResultMinor sind die folgenden Werte möglich:</p> <ul style="list-style-type: none"> <li>• <a href="#">.../resultminor/ar/unknownLocation</a></li> <li>• <a href="#">.../resultminor/al/common#parameterError</a></li> <li>• <a href="#">.../resultminor/al/common#internalError</a></li> </ul>
tr:DataLocation		Das DataLocation-Element spezifiziert, welche Teile eines Archivobjektes zurückgeliefert werden. Weitere Details zu diesem Element finden sich auf Seite 29.
Value		Enthält im Erfolgsfall die gewünschten Daten.

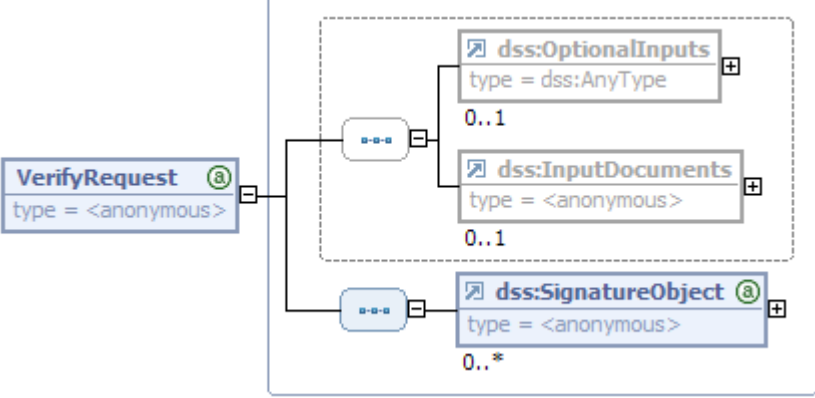
<b>Name</b>	<b>ArchiveDataResponse</b>	
	 <p>Statusinformationen und Fehler bei ArchiveDataResponse (vgl. [eCard-1]).</p>	
	<b>Name</b>	<b>Fehlercode</b>
	ResultMajor	<ul style="list-style-type: none"> <li>• <a href="#">/resultmajor#ok</a></li> <li>• <a href="#">/resultmajor#error</a></li> <li>• <a href="#">/resultmajor#warning</a></li> </ul>
	ResultMinor	<ul style="list-style-type: none"> <li>• <a href="#">/resultminor/al/common#noPermission</a></li> <li>• <a href="#">/resultminor/al/common#internalError</a></li> <li>• <a href="#">/resultminor/al/common#parameterError</a></li> <li>• <a href="#">/resultminor/ar/unknownAOID</a></li> <li>• <a href="#">/resultminor/ar/unsupported</a></li> <li>• <a href="#">/resultminor/ar/requestOnlyPartlySuccessfulWarning</a></li> </ul>

### 3.7 VerifyRequest und VerifyResponse

#### 3.7.1 VerifyRequest

Mit der Funktion `VerifyRequest` können XML-basierte Archivdatenobjekte (XAIP) samt der darin enthaltenen oder zusätzlich übergebenen beweisrelevanten Daten (Signaturen, Zeitstempel, Zertifikate, Sperrlisten, OCSP-Responses etc.) und Beweisdaten (Evidence Records) geprüft werden.

Wie in Abbildung 2 ersichtlich, wird diese Funktion neben der hier betrachteten Schnittstelle TR-S.4 auch in TR-S.1 (vgl. Abschnitt 4.1) genutzt.

<b>Name</b>	<b>VerifyRequest</b>	
<b>Beschreibung</b>	<p>Mit der Funktion <code>VerifyRequest</code> (vgl. Abschnitt 3.2.2 von <b>[eCard-2]</b>) werden XML-basierte Archivdatenobjekte (XAIP) bzw. ergänzendes XML-basiertes Archivdatenobjekt (Delta-XAIP) samt der darin enthaltenen oder zusätzlich übergebenen beweisrelevanten Daten (Signaturen, Zeitstempel, Zertifikate, Sperrlisten, OCSP-Responses etc.) und Beweisdaten (Evidence Records) geprüft.</p>	
<b>Aufrufparameter</b>	 <p>Aufruf der <code>VerifyRequest</code>-Funktion.</p>	
	<b>Name</b>	<b>Beschreibung</b>



Name	VerifyRequest
	<p>dss:OptionalInputs</p> <p>Entspricht dem requestControls-Element und kann zusätzliche Eingabeelemente enthalten.</p> <p><b>(A3.7.1-1):</b> Hierbei <u>sollen</u> insbesondere die in [eCard-2] definierten Elemente und Aufrufoptionen unterstützt werden.</p> <p>Dies umfasst insbesondere die folgenden Elemente:</p> <ul style="list-style-type: none"> <li>• VerifyUnderSignaturePolicy,</li> <li>• ReturnVerificationReport</li> </ul> <p>Es gilt im Einzelnen:</p> <ul style="list-style-type: none"> <li>• VerifyUnderSignaturePolicy</li> </ul> <p>Sofern in einem dss:Document/InlineXML-Kindelament von dss:InputDocuments ein XAIP-Element gemäß [TR-ESOR-F] enthalten ist, kann mit dem Element VerifyUnderSignaturePolicy und der im DefaultPolicy/SignaturePolicyIdentifier-Element angegebenen Signature-Policy <a href="http://www.bsi.bund.de/DE/tr-esor/sigpolicy/verify-timestamp">http://www.bsi.bund.de/DE/tr-esor/sigpolicy/verify-timestamp</a> die Prüfung und Ergänzung aller im übergebenen XAIP-Container enthaltenen Signaturen angefordert werden.</p> <p><b>(A3.7.1-2):</b> Hierbei <u>müssen</u> alle Signaturinformationen (Signaturen, Zeitstempel, Zertifikate, Sperrlisten, OCSP-Responses etc.) bis hin zu einer vertrauenswürdigen Wurzel geprüft werden.</p> <p>Die hierbei ermittelten Prüfinformationen (Zertifikate, Sperrlisten, OCSP-Responses) werden nach Möglichkeit als unsignierte Attribute bzw. Properties in den entsprechenden Signaturen bzw. in den Kind-Elementen certificateValues bzw. revocationValues des credential-Elementes abgelegt.</p>

Name	VerifyRequest	
		<p><b>(A3.7.1-3):</b> Sofern in der <code>credentialSection</code> des übergebenen XAIP-Containers ein oder mehrere <code>xaip:EvidenceRecord</code>-Elemente gemäß <b>[TR-ESOR-F]</b> enthalten sind, müssen diese entsprechend geprüft werden.</p> <ul style="list-style-type: none"> <li>• <code>ReturnVerificationReport</code></li> </ul> <p>Durch die Übergabe eines <code>ReturnVerificationReport</code>-Elementes gemäß <b>[OASIS VR]</b> bzw. <b>[eCard-2]</b> und <b>[TR-ESOR-VR]</b> kann ein ausführlicher Prüfbericht in Form eines <code>VerificationReport</code>-Elementes für die übergebenen Objekte (Signaturen, Zeitstempel, Zertifikate, Sperrinformationen, Evidence Records, XAIP mit den vorgenannten Daten) angefordert werden.</p>
	<code>dss:InputDocuments</code>	<p>Das <code>dss:InputDocuments</code>-Element enthält die zur Prüfung benötigten Dokumente, sofern diese nicht bereits im unten erläuterten <code>SignatureObject</code>-Element enthalten sind.</p> <p>Außerdem <u>kann</u> in einem <code>dss:Document/InlineXML</code>-Kindelement ein XAIP-Element gemäß <b>[TR-ESOR-F]</b> übergeben werden, so dass alle darin enthaltenen Signaturen in Verbindung mit der oben angegebenen <code>Signature-Policy</code> geprüft und ergänzt werden oder die Prüfung der darin enthaltenen Evidence Records angestoßen wird.</p>

<b>Name</b>	<b>VerifyRequest</b>	
	dss:SignatureObject	<p>In dss:SignatureObject-Elementen können grundsätzlich eigenständige Signaturen (detached signatures) zur Prüfung übergeben werden. Wenn Signaturen bereits im dss:InputDocuments enthalten sind, können die optionalen dss:SignatureObject-Elemente entfallen.</p> <p><b>(A3.7.1-4):</b> Als Kindelement von dss:SignatureObject/Other kann auch ein xaip:EvidenceRecord-Element übergeben werden, um die entsprechende Prüfung des Evidence Record anzustoßen. In diesem Fall <u>müssen</u> die Attribute AOID und VersionID vorhanden sein und das zugehörige XAIP-Element <u>muss</u> als Kindelement von dss:InputDocuments/dss:Document/InlineXML übergeben werden.</p> <p>Sofern das dss:SignatureObject-Element fehlt, <u>muss</u> genau ein dss:InputDocuments-Element vorhanden sein, das die zu prüfenden Signaturobjekte enthält.</p>

### 3.7.2 VerifyResponse

<b>Name</b>	<b>VerifyResponse</b>	
<b>Beschreibung</b>	Als Antwort auf einen VerifyRequest wird ein entsprechendes VerifyResponse-Element gemäß Abschnitt 3.2.2 von [eCard-2] zurückgeliefert.	
<b>Rückgabe</b>	<p>Rückgabe der VerifyRequest-Funktion</p>	
	<b>Name</b>	<b>Beschreibung</b>
	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und Abschnitt 3.2.2 von [eCard-2] beschrieben.

Name	VerifyResponse	
	dss:OptionalOutputs	<p>Sofern ein Fehler aufgetreten ist, enthält dieses Element entweder den Prüfbericht in Form eines <code>VerificationReport</code>-Elementes oder das um diese Prüfinformationen ergänzte Archivdatenobjekt in Form eines <code>xaip:XAIP</code>-Elements.</p> <p>Die grundsätzliche Struktur des Prüfberichtes ist in <b>[OASIS-VR]</b> näher beschrieben. In <b>[TR-ESOR-VR]</b> finden sich entsprechende Korrekturen für den <code>EvidenceRecordReport</code> sowie die Beschreibung des <code>XAIPReport</code>.</p> <p>Details zur Ablage dieser Prüfinformationen im XAIP-Container finden sich in <b>[TR-ESOR-F]</b>.</p>

## 4. Funktionen der interne Schnittstellen

In diesem Abschnitt werden die internen Schnittstellen der Referenzarchitektur TR-S.1 bis TR-S.3 und TR-S.5 bis TR-S.6 (vgl. Abbildung Abbildung 2) erläutert:

- TR-S.1: TR-ESOR-S.1 (ArchiSafe-Modul – Krypto-Modul) (siehe Abschnitt 4.1)
- TR-S.2: TR-ESOR-S.2 (ArchiSig-Modul – ECM-/Langzeitspeichersystem) (siehe Abschnitt 4.2)
- TR-S.3: TR-ESOR-S.3 (ArchiSig-Modul – Krypto-Modul) (siehe Abschnitt 4.3)
- TR-S.5: TR-ESOR-S.5 (ArchiSafe-Modul –ECM-/Langzeitspeichersystem) (siehe Abschnitt 4.4)
- TR-S.6: TR-ESOR-S.6 (ArchiSafe-Modul – ArchiSig-Modul) (siehe Abschnitt 4.5)

### 4.1 TR-ESOR-S.1 (ArchiSafe-Modul – Krypto-Modul)

Dieser Abschnitt beschreibt, wie die in [TR-ESOR-S] skizzierte Schnittstelle TR-S.1 auf Basis des eCard-API-Frameworks (BSI TR 03112) umgesetzt werden kann.

Die in [TR-ESOR-S] definierte Schnittstelle TR-S.1 umfasst zwei wesentliche Funktionen:

- Prüfung von Signaturen, beweisrelevanten Daten, Beweisdaten und Archivdatenobjekten (`VerifyRequest` / `VerifyResponse`)
- Signaturerzeugung (optional) (`SignRequest` / `SignResponse`)

#### 4.1.1 Prüfung von Signaturen, beweisrelevanten Daten, Beweisdaten und Archivdatenobjekten

Für die Prüfung von Signaturen, beweisrelevanten Daten (Zertifikaten, Zertifikatstatusinformationen, Zeitstempeln, etc.), Beweisdaten (Evidence Records) und Archivdatenobjekten (XAIPs) sind in [TR-ESOR-S] die Schnittstellensignaturen `VerifyRequest` und `VerifyResponse` als ASN.1-Strukturen definiert. Dies entspricht den gleichnamigen XML-Strukturen in [OASIS-DSS] und [eCard-2] mit den entsprechenden Korrekturen und Ergänzungen aus [TR-ESOR-VR], wie in Abschnitt 3.7 erläutert.

#### 4.1.2 Signaturerstellung

Für die Signaturerstellung sind in [TR-ESOR-S] die ASN.1-Strukturen `SignRequest` und `SignResponse` definiert. Dies entspricht den gleichnamigen XML-Strukturen in [OASIS-DSS] und [eCard-2].

##### 4.1.2.1 SignRequest

Ein `SignRequest` im Kontext der Schnittstelle S.1 übergibt ein Archivdatenobjekt (XAIP-Dokument) an das Krypto-Modul zur Erzeugung einer elektronischen Signatur.

<b>Name</b>	<b>SignRequest</b>						
<b>Beschreibung</b>	Mit der Funktion <code>SignRequest</code> aus [eCard-2] kann das übergebene Archivdatenobjekt mit einer (qualifizierten) elektronischen Signatur versehen werden.						
<b>Beschreibung</b>	<p>Aufruf der <code>SignRequest</code>-Funktion</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td><code>dss:OptionalInputs</code></td> <td>Entspricht dem <code>requestControls</code>-Element aus [TR-ESOR-S] und kann eines oder mehrere der in [eCard-2] definierten optionalen Eingabeelemente enthalten.</td> </tr> <tr> <td><code>dss:InputDocuments</code></td> <td>Enthält die zu signierenden Dokumente oder Datenstrukturen. Weitere Informationen hierzu finden sich in [OASIS-DSS] und [eCard-2].</td> </tr> </tbody> </table>	Name	Beschreibung	<code>dss:OptionalInputs</code>	Entspricht dem <code>requestControls</code> -Element aus [TR-ESOR-S] und kann eines oder mehrere der in [eCard-2] definierten optionalen Eingabeelemente enthalten.	<code>dss:InputDocuments</code>	Enthält die zu signierenden Dokumente oder Datenstrukturen. Weitere Informationen hierzu finden sich in [OASIS-DSS] und [eCard-2].
Name	Beschreibung						
<code>dss:OptionalInputs</code>	Entspricht dem <code>requestControls</code> -Element aus [TR-ESOR-S] und kann eines oder mehrere der in [eCard-2] definierten optionalen Eingabeelemente enthalten.						
<code>dss:InputDocuments</code>	Enthält die zu signierenden Dokumente oder Datenstrukturen. Weitere Informationen hierzu finden sich in [OASIS-DSS] und [eCard-2].						

#### 4.1.2.2 SignResponse

<b>Name</b>	<b>SignResponse</b>				
<b>Beschreibung</b>	Als Antwort auf einen <code>SignRequest</code> wird vom Krypto-Modul ein entsprechendes <code>SignResponse</code> -Element gemäß Abschnitt 3.2.1 von [eCard-2] zurückgeliefert.				
<b>Rückgabe</b>	<p><code>SignResponse</code> ist die Antwort zum <code>SignRequest</code>-Aufruf</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td><code>dss:Result</code></td> <td>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.2.1 von [eCard-2] beschrieben.</td> </tr> </tbody> </table>	Name	Beschreibung	<code>dss:Result</code>	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.2.1 von [eCard-2] beschrieben.
Name	Beschreibung				
<code>dss:Result</code>	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.2.1 von [eCard-2] beschrieben.				

Name	SignResponse	
	dss:OptionalOutputs	<u>Kann</u> ein DocumentWithSignature-Element enthalten, in denen z.B. ein XAIP-Element mit der eingebetteten Signatur enthalten ist. Details finden sich in Abschnitt 3.2.1 von [eCard-2].
	dss:SignatureObject	<u>Kann</u> eine erzeugte Signatur in Form eines dss:SignatureObject-Elementes enthalten. Details finden sich in Abschnitt 3.2.1 von [eCard-2]. Sofern die erstellte Signatur bereits im o.g. DocumentWithSignature-Element vorhanden ist, wird kein dss:SignatureObject-Element zurückgeliefert.

## 4.2 TR-ESOR-S.2 (ArchiSig-Modul – ECM-/Langzeitspeichersystem)

Dieser Abschnitt beschreibt, wie die in [TR-ESOR-S] skizzierte Schnittstelle TR-S.2 auf Basis der auch dem eCard-API-Frameworks (BSI TR 03112) zu Grunde liegenden Basistypen aus [OASIS-DSS] umgesetzt werden kann.

Diese Schnittstelle umfasst drei wesentliche Funktionen:

- Speichern eines Archivdatenobjektes (ArchiveSubmissionRequest / ArchiveSubmissionResponse)
- Ergänzen einer neuen Version eines Archivdatenobjektes (ArchiveUpdateRequest / ArchiveUpdateResponse)
- Auslesen eines Archivdatenobjektes (ArchiveRetrievalRequest / ArchiveRetrievalResponse)

### 4.2.1 Speichern eines Archivdatenobjektes

Für das Speichern eines Archivdatenobjektes sind in [TR-ESOR-S] die Schnittstellensignaturen ArchiveSubmissionRequest und ArchiveSubmissionResponse als ASN.1-Strukturen definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.1 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (dss:RequestBaseType und dss:ResponseBaseType) aus [OASIS-DSS] spezifiziert wurden.

### 4.2.2 Ergänzen einer neuen Version eines Archivdatenobjektes

Für das Ergänzen einer neuen Version eines Archivdatenobjektes sind in [TR-ESOR-S] die Schnittstellensignaturen ArchiveUpdateRequest und ArchiveUpdateResponse als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.2 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (dss:RequestBaseType und dss:ResponseBaseType) aus [OASIS-DSS] spezifiziert wurden.

### 4.2.3 Auslesen von Archivdatenobjekten

Für das Auslesen von Archivdatenobjekten sind in [TR-ESOR-S] die Schnittstellensignaturen ArchiveRetrievalRequest und ArchiveRetrievalResponse als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.3 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (dss:RequestBaseType und dss:ResponseBaseType) aus [OASIS-DSS] spezifiziert wurden.

### 4.3 TR-ESOR-S.3 (ArchiSig-Modul – Krypto-Modul)

Dieser Abschnitt beschreibt, wie die in [TR-ESOR-S] skizzierte Schnittstelle TR-S.3 auf Basis des eCard-API-Frameworks (BSI TR 03112) umgesetzt werden kann.

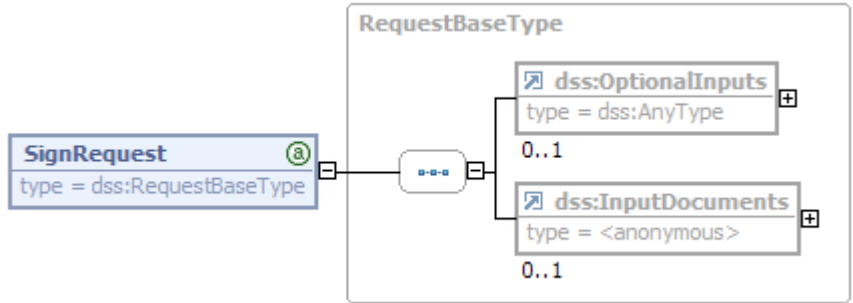
Die in [TR-ESOR-S] definierte Schnittstelle TR-S.3 umfasst drei wesentliche Funktionen:

- Anfordern eines (qualifizierten) Zeitstempels (TimestampRequest / TimestampResponse)
- Prüfen eines (qualifizierten) Zeitstempels (VerifyRequest / VerifyResponse)
- Berechnung eines Hashwertes (Hash / HashResponse)

#### 4.3.1 Anfordern eines (qualifizierten) Zeitstempels

Zum Anfordern eines (qualifizierten) Zeitstempels sind in TR-S.3 die auf [RFC3161] zurückgehenden ASN.1-Strukturen TimestampRequest / TimestampResponse vorgesehen. Dies entspricht der [OASIS-DSS]-basierten Funktion SignRequest / SignResponse aus [eCard-2].

##### 4.3.1.1 TimestampRequest wird realisiert durch SignRequest

<b>Name</b>	<b>SignRequest</b>						
<b>Beschreibung</b>	Ein SignRequest im Kontext der Schnittstelle S.3 übergibt einen Hashwert, zu dem ein (qualifizierter) Zeitstempel erstellt werden soll, an das Krypto-Modul.						
<b>Beschreibung</b>	 <p>Aufruf der SignRequest-Funktion</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>dss:OptionalInputs</td> <td>Enthält genau ein Element SignatureType mit der URI <a href="urn:ietf:rfc:3161">urn:ietf:rfc:3161</a>, durch die klargestellt wird, dass ein Zeitstempel gemäß [RFC3161] erzeugt werden soll.</td> </tr> <tr> <td>dss:InputDocuments</td> <td><b>(A4.3.1.1-1):</b> Während das Element dss:InputDocuments in [OASIS-DSS] und [eCard-2] optional ist, <u>muss</u> es hier vorhanden sein und genau ein dss:Document-Element in der DocumentHash-Ausprägung enthalten. Dieses Element enthält den Hashwert, aus dem ein (qualifizierter) Zeitstempel erzeugt werden soll.</td> </tr> </tbody> </table>	Name	Beschreibung	dss:OptionalInputs	Enthält genau ein Element SignatureType mit der URI <a href="urn:ietf:rfc:3161">urn:ietf:rfc:3161</a> , durch die klargestellt wird, dass ein Zeitstempel gemäß [RFC3161] erzeugt werden soll.	dss:InputDocuments	<b>(A4.3.1.1-1):</b> Während das Element dss:InputDocuments in [OASIS-DSS] und [eCard-2] optional ist, <u>muss</u> es hier vorhanden sein und genau ein dss:Document-Element in der DocumentHash-Ausprägung enthalten. Dieses Element enthält den Hashwert, aus dem ein (qualifizierter) Zeitstempel erzeugt werden soll.
Name	Beschreibung						
dss:OptionalInputs	Enthält genau ein Element SignatureType mit der URI <a href="urn:ietf:rfc:3161">urn:ietf:rfc:3161</a> , durch die klargestellt wird, dass ein Zeitstempel gemäß [RFC3161] erzeugt werden soll.						
dss:InputDocuments	<b>(A4.3.1.1-1):</b> Während das Element dss:InputDocuments in [OASIS-DSS] und [eCard-2] optional ist, <u>muss</u> es hier vorhanden sein und genau ein dss:Document-Element in der DocumentHash-Ausprägung enthalten. Dieses Element enthält den Hashwert, aus dem ein (qualifizierter) Zeitstempel erzeugt werden soll.						



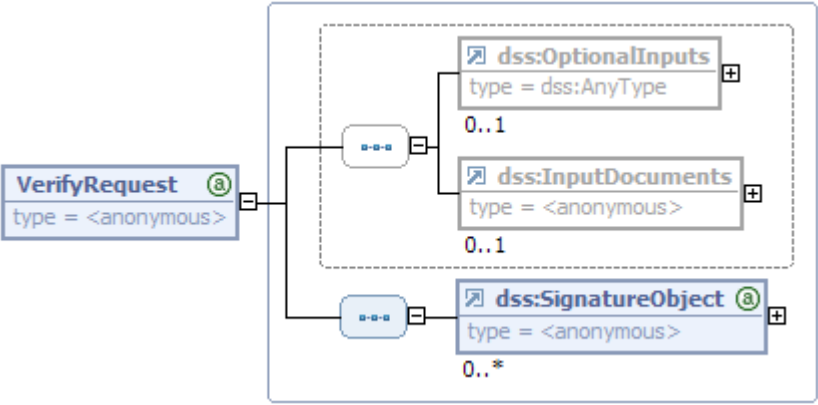
**4.3.1.2 TimestampResponse wird realisiert durch SignResponse**

<b>Name</b>	<b>SignResponse</b>									
<b>Beschreibung</b>	Als Antwort auf einen <code>SignRequest</code> wird vom Krypto-Modul ein entsprechendes <code>SignResponse</code> -Element gemäß Abschnitt 3.2.1 von <b>[eCard-2]</b> zurückgeliefert. Im Kontext der Schnittstelle S.3 wird hier ein (qualifizierter) Zeitstempel zurückgeliefert.									
<b>Rückgabe</b>	<p>SignResponse ist die Antwort zum SignRequest-Aufruf</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td><code>dss:Result</code></td> <td>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von <b>[eCard-1]</b> und in Abschnitt 3.2.1 von <b>[eCard-2]</b> beschrieben.</td> </tr> <tr> <td><code>dss:OptionalOutputs</code></td> <td>Das optionale Element <code>dss:OptionalOutputs</code> ist nicht vorhanden.</td> </tr> <tr> <td><code>dss:SignatureObject</code></td> <td>Enthält – sofern kein Fehler aufgetreten ist – genau ein <code>dss:SignatureObject</code>-Element, das ein <code>dss:Timestamp</code>-Element enthält, in dem der Zeitstempel in Form eines <code>RFC3161TimeStampToken</code>-Elementes enthalten ist.</td> </tr> </tbody> </table>		Name	Beschreibung	<code>dss:Result</code>	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von <b>[eCard-1]</b> und in Abschnitt 3.2.1 von <b>[eCard-2]</b> beschrieben.	<code>dss:OptionalOutputs</code>	Das optionale Element <code>dss:OptionalOutputs</code> ist nicht vorhanden.	<code>dss:SignatureObject</code>	Enthält – sofern kein Fehler aufgetreten ist – genau ein <code>dss:SignatureObject</code> -Element, das ein <code>dss:Timestamp</code> -Element enthält, in dem der Zeitstempel in Form eines <code>RFC3161TimeStampToken</code> -Elementes enthalten ist.
Name	Beschreibung									
<code>dss:Result</code>	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von <b>[eCard-1]</b> und in Abschnitt 3.2.1 von <b>[eCard-2]</b> beschrieben.									
<code>dss:OptionalOutputs</code>	Das optionale Element <code>dss:OptionalOutputs</code> ist nicht vorhanden.									
<code>dss:SignatureObject</code>	Enthält – sofern kein Fehler aufgetreten ist – genau ein <code>dss:SignatureObject</code> -Element, das ein <code>dss:Timestamp</code> -Element enthält, in dem der Zeitstempel in Form eines <code>RFC3161TimeStampToken</code> -Elementes enthalten ist.									

**4.3.2 Prüfen eines (qualifizierten) Zeitstempels**

Zum Prüfen eines (qualifizierten) Zeitstempels sind in TR-S.3 die ASN.1-Strukturen `VerifyRequest` / `VerifyResponse` vorgesehen. Dies entspricht den gleichnamigen Funktionen aus **[OASIS-DSS]** und **[eCard-2]**.

## 4.3.2.1 VerifyRequest

<b>Name</b>	<b>VerifyRequest</b>	
<b>Beschreibung</b>	Ein <code>VerifyRequest</code> im Kontext der Schnittstelle S.3 übergibt einen (qualifizierten) Zeitstempel an das Krypto-Modul zur Verifikation der darin enthaltenen (qualifizierten) elektronischen Signatur. Außerdem werden die für die Prüfung genutzten Zertifikate und Sperrinformationen in den zurück gelieferten Zeitstempel eingefügt. Entsprechende Empfehlungen für die Ablage dieser Informationen finden sich in [TR-ESOR-F].	
<b>Aufrufparameter</b>	 <p>Aufruf der <code>VerifyRequest</code>-Funktion.</p>	
	<b>Name</b>	<b>Beschreibung</b>
	<code>dss:OptionalInputs</code>	<p>Entspricht dem <code>requestControls</code>-Element aus [TR-ESOR-S] und kann optionale Eingabeelemente enthalten.</p> <p><b>(A4.3.2.1-1):</b> Gemäß der vorliegenden Spezifikation <u>muß</u> das optionale Eingabeelement <code>ReturnUpdatedSignature</code> aus Abschnitt 4.5.8 von [OASIS-DSS] unterstützt werden, bei dem mit dem <code>Type</code>-Attribut <a href="http://www.bsi.bund.de/tr-esor/api/1.2">http://www.bsi.bund.de/tr-esor/api/1.2</a> klargestellt wird, dass alle bei der Prüfung verwendeten Zertifikate und Sperrinformationen wie in [TR-ESOR-F] spezifiziert in den Zeitstempel eingefügt werden <u>müssen</u>.</p> <p><b>(A4.3.2.1-2):</b> Darüber hinaus <u>soll</u> das optionale Eingabeelement <code>ReturnVerificationReport</code> unterstützt werden, so dass für den entsprechenden Zeitstempel ein Prüfbericht gemäß [OASIS-VR] zurückgeliefert werden kann.</p>
	<code>dss:InputDocuments</code>	Das optionale Element <code>dss:InputDocuments</code> <u>soll nicht</u> vorhanden sein und wird ignoriert.

<b>Name</b>	<b>VerifyRequest</b>	
	dss:SignatureObject	Es ist genau ein dss:SignatureObject-Element in der dss:TimeStamp/ RFC3161TimeStampToken Ausprägung vorhanden, das den zu prüfenden Zeitstempel enthält.

#### 4.3.2.2 VerifyResponse

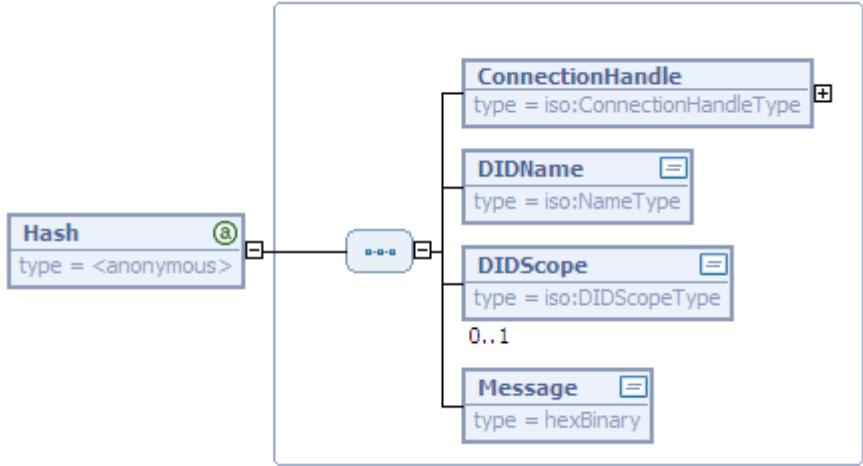
<b>Name</b>	<b>VerifyResponse</b>							
<b>Beschreibung</b>	Als Antwort auf einen VerifyRequest wird vom Krypto-Modul ein entsprechendes VerifyResponse-Element gemäß Abschnitt 3.2.2 von [eCard-2] zurückgeliefert.							
<b>Rückgabe</b>	<p>Rückgabe der VerifyRequest-Funktion</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>dss:Result</td> <td>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.2.2 von [eCard-2] beschrieben.</td> </tr> <tr> <td>dss:OptionalOutputs</td> <td>Sofern nicht ein Fehler aufgetreten ist, <u>muss</u> ein UpdatedSignature-Element vorhanden sein, das ein dss:SignatureObject-Element in der dss:TimeStamp/ RFC3161TimeStampToken-Ausprägung enthält, in dem sich der um die bei der Prüfung genutzten Zertifikate und Sperrinformationen ergänzte Zeitstempel befindet. Darüber hinaus <u>kann</u> ein VerificationReport-Element gemäß [OASIS VR] vorhanden sein, das im IndividualReport/Details-Element ein IndividualTimeStampReport-Element enthält.</td> </tr> </tbody> </table>		Name	Beschreibung	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.2.2 von [eCard-2] beschrieben.	dss:OptionalOutputs	Sofern nicht ein Fehler aufgetreten ist, <u>muss</u> ein UpdatedSignature-Element vorhanden sein, das ein dss:SignatureObject-Element in der dss:TimeStamp/ RFC3161TimeStampToken-Ausprägung enthält, in dem sich der um die bei der Prüfung genutzten Zertifikate und Sperrinformationen ergänzte Zeitstempel befindet. Darüber hinaus <u>kann</u> ein VerificationReport-Element gemäß [OASIS VR] vorhanden sein, das im IndividualReport/Details-Element ein IndividualTimeStampReport-Element enthält.
Name	Beschreibung							
dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.2.2 von [eCard-2] beschrieben.							
dss:OptionalOutputs	Sofern nicht ein Fehler aufgetreten ist, <u>muss</u> ein UpdatedSignature-Element vorhanden sein, das ein dss:SignatureObject-Element in der dss:TimeStamp/ RFC3161TimeStampToken-Ausprägung enthält, in dem sich der um die bei der Prüfung genutzten Zertifikate und Sperrinformationen ergänzte Zeitstempel befindet. Darüber hinaus <u>kann</u> ein VerificationReport-Element gemäß [OASIS VR] vorhanden sein, das im IndividualReport/Details-Element ein IndividualTimeStampReport-Element enthält.							

#### 4.3.3 Berechnung eines Hashwertes

Zur Berechnung eines Hashwertes sind in TR-S.3 die Schnittstellensignaturen HashRequest / HashResponse als ASN.1-Strukturen vorgesehen. Dies entspricht der Funktion Hash /

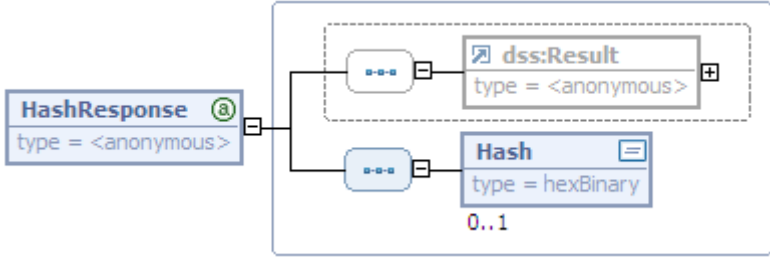
HashResponse aus [eCard-4] in Verbindung mit dem Generic Cryptography-Protokoll aus [eCard-7].

#### 4.3.3.1 Hash

Name	Hash										
<b>Beschreibung</b>	Bei einem Hash-Aufruf im Kontext der Schnittstelle S.3 wird für die übergebenen Daten ein Hashwert berechnet.										
<b>Aufrufparameter</b>	 <p>Aufruf der Funktion Hash.</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>ConnectionHandle</td> <td>Das ConnectionHandle-Element (vgl. [eCard-4], Abschnitt 3.1.3) gibt bei Bedarf an, auf welchem Hardwaremodul oder entfernten eCard-API-Framework die Berechnung des Hashwertes erfolgen soll. Sofern die Berechnung des Hashwertes durch das lokale Software-Modul erfolgen soll, <u>soll</u> das ConnectionHandle-Element leer sein.</td> </tr> <tr> <td>DIDName<sup>9</sup></td> <td>Dieser Parameter spezifiziert den zu verwendenden Hashalgorithmus, wobei derzeit folgende Algorithmen unterstützt werden <u>müssen</u>: <ul style="list-style-type: none"> <li>• SHA-256</li> <li>• SHA-384</li> <li>• SHA-512</li> </ul> </td> </tr> <tr> <td>DIDScope</td> <td>Löst im ISO/IEC 24727-3 Standard Mehrdeutigkeiten zwischen lokalen und globalen DIDs mit gleichem Namen auf. Dieser Parameter wird hier nicht verwendet und sofern vorhanden ignoriert.</td> </tr> <tr> <td>Message</td> <td>Enthält die Nachricht (bzw. einen Teil derselben, siehe [eCard-7]), aus der ein Hashwert berechnet werden soll.</td> </tr> </tbody> </table>	Name	Beschreibung	ConnectionHandle	Das ConnectionHandle-Element (vgl. [eCard-4], Abschnitt 3.1.3) gibt bei Bedarf an, auf welchem Hardwaremodul oder entfernten eCard-API-Framework die Berechnung des Hashwertes erfolgen soll. Sofern die Berechnung des Hashwertes durch das lokale Software-Modul erfolgen soll, <u>soll</u> das ConnectionHandle-Element leer sein.	DIDName <sup>9</sup>	Dieser Parameter spezifiziert den zu verwendenden Hashalgorithmus, wobei derzeit folgende Algorithmen unterstützt werden <u>müssen</u> : <ul style="list-style-type: none"> <li>• SHA-256</li> <li>• SHA-384</li> <li>• SHA-512</li> </ul>	DIDScope	Löst im ISO/IEC 24727-3 Standard Mehrdeutigkeiten zwischen lokalen und globalen DIDs mit gleichem Namen auf. Dieser Parameter wird hier nicht verwendet und sofern vorhanden ignoriert.	Message	Enthält die Nachricht (bzw. einen Teil derselben, siehe [eCard-7]), aus der ein Hashwert berechnet werden soll.
Name	Beschreibung										
ConnectionHandle	Das ConnectionHandle-Element (vgl. [eCard-4], Abschnitt 3.1.3) gibt bei Bedarf an, auf welchem Hardwaremodul oder entfernten eCard-API-Framework die Berechnung des Hashwertes erfolgen soll. Sofern die Berechnung des Hashwertes durch das lokale Software-Modul erfolgen soll, <u>soll</u> das ConnectionHandle-Element leer sein.										
DIDName <sup>9</sup>	Dieser Parameter spezifiziert den zu verwendenden Hashalgorithmus, wobei derzeit folgende Algorithmen unterstützt werden <u>müssen</u> : <ul style="list-style-type: none"> <li>• SHA-256</li> <li>• SHA-384</li> <li>• SHA-512</li> </ul>										
DIDScope	Löst im ISO/IEC 24727-3 Standard Mehrdeutigkeiten zwischen lokalen und globalen DIDs mit gleichem Namen auf. Dieser Parameter wird hier nicht verwendet und sofern vorhanden ignoriert.										
Message	Enthält die Nachricht (bzw. einen Teil derselben, siehe [eCard-7]), aus der ein Hashwert berechnet werden soll.										

<sup>9</sup> Eine in ISO/IEC 24727 näher beschriebene Differential Identity ermöglicht die Ausführung von kryptographischen Operationen. Der DIDName ist der logische Name, der für den Zugriff auf dieses kryptographische Objekt genutzt wird.

### 4.3.3.2 HashResponse

<b>Name</b>	<b>HashResponse</b>							
<b>Beschreibung</b>	Als Antwort auf einen Hash-Aufruf wird vom Krypto-Modul ein entsprechendes HashResponse-Element gemäß Abschnitt 3.5.4 von [eCard-4] zurückgeliefert.							
<b>Rückgabe</b>	 <p>Rückgabe der Funktion Hash.</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Beschreibung</th> </tr> </thead> <tbody> <tr> <td>dss:Result</td> <td>Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.5.4 von [eCard-4] beschrieben.</td> </tr> <tr> <td>Hash</td> <td>Enthält den Hashwert, sofern ein solcher berechnet werden konnte.</td> </tr> </tbody> </table>		Name	Beschreibung	dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.5.4 von [eCard-4] beschrieben.	Hash	Enthält den Hashwert, sofern ein solcher berechnet werden konnte.
Name	Beschreibung							
dss:Result	Enthält die Statusinformationen und die Fehler zu einer durchgeführten Aktion. Die Struktur dieses Elements und die möglichen Fehlercodes sind in Abschnitt 4.1.2 von [eCard-1] und in Abschnitt 3.5.4 von [eCard-4] beschrieben.							
Hash	Enthält den Hashwert, sofern ein solcher berechnet werden konnte.							

## 4.4 TR-ESOR-S.5 (ArchiSafe-Modul – ECM-Langzeitspeichersystem)

Dieser Abschnitt beschreibt, wie die in TR-S.5 skizzierte Schnittstelle auf Basis der auch dem eCard-API-Framework (BSI TR 03112) zu Grunde liegenden Basistypen aus [OASIS-DSS] umgesetzt werden kann.

Die in TR-S.5 definierte Schnittstelle umfasst die folgenden Funktionen:

- Abfrage beweiswerterhaltend archivierter Daten (ArchiveRetrievalRequest / -Response)
- Löschen von Archivdatenobjekten (ArchiveDeletionRequest / -Response)
- Abfrage diskreter Datenobjekte (ArchiveDataRequest / -Response)

### 4.4.1 Abfrage beweiswerterhaltend archivierter Daten

Für die Abfrage beweiswerterhaltend archivierter Daten sind in [TR-ESOR-S] die Schnittstellensignaturen ArchiveRetrievalRequest und ArchiveRetrievalResponse als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.3 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (dss:RequestBaseType und dss:ResponseBaseType) aus [OASIS-DSS] spezifiziert wurden.

### 4.4.2 Löschen von Archivdatenobjekten

Für das Löschen von Archivdatenobjekten sind in [TR-ESOR-S] die Schnittstellensignaturen ArchiveDeletionRequest und ArchiveDeletionResponse als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.5 näher beschriebenen XML-Strukturen, die analog

zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (`dss:RequestBaseType` und `dss:ResponseBaseType`) aus [OASIS-DSS] spezifiziert wurden.

#### 4.4.3 Abfrage diskreter Datenobjekte

Für die Abfrage diskreter Datenobjekte sind in [TR-ESOR-S] die Schnittstellensignaturen `ArchiveDataRequest` und `ArchiveDataResponse` als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.5 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (`dss:RequestBaseType` und `dss:ResponseBaseType`) aus [OASIS-DSS] spezifiziert wurden.

### 4.5 TR-ESOR-S.6 (ArchiSafe-Modul – ArchiSig-Modul)

Dieser Abschnitt beschreibt, wie die in [TR-ESOR-S] skizzierte Schnittstelle TR-S.6 auf Basis der auch dem eCard-API-Frameworks (BSI TR 03112) zu Grunde liegenden Basistypen aus [OASIS-DSS] umgesetzt werden kann.

Die in [TR-ESOR-S] definierte Schnittstelle TR-S.6 umfasst die folgenden Funktionen:

- Beweiswerterhaltende Archivierung elektronischer Daten (`ArchiveSubmissionRequest` / `ArchiveSubmissionResponse`)
- Ergänzen einer neuen Version eines Archivdatenobjektes (`ArchiveUpdateRequest` / `ArchiveUpdateResponse`)
- Rückgabe technischer Beweisdaten (`ArchiveEvidenceRequest` / `ArchiveEvidenceResponse`)

#### 4.5.1 Beweiswerterhaltende Archivierung elektronischer Daten

Für die beweiswerterhaltende Archivierung elektronischer Daten sind in [TR-ESOR-S] die Schnittstellensignaturen `ArchiveSubmissionRequest` und `ArchiveSubmissionResponse` als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.1 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (`dss:RequestBaseType` und `dss:ResponseBaseType`) aus [OASIS-DSS] spezifiziert wurden.

#### 4.5.2 Ergänzen einer neuen Version eines Archivdatenobjektes

Für das Ergänzen einer neuen Version eines Archivdatenobjektes sind in [TR-ESOR-S] die Schnittstellensignaturen `ArchiveUpdateRequest` und `ArchiveUpdateResponse` als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.2 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (`dss:RequestBaseType` und `dss:ResponseBaseType`) aus [OASIS-DSS] spezifiziert wurden.

#### 4.5.3 Rückgabe technischer Beweisdaten

Für die Rückgabe technischer Beweisdaten sind in [TR-ESOR-S] die Schnittstellensignaturen `ArchiveEvidenceRequest` und `ArchiveEvidenceResponse` als ASN.1-Struktur definiert. Dies entspricht den gleichnamigen und in Abschnitt 3.4 näher beschriebenen XML-Strukturen, die analog zu [eCard-2] unter Verwendung der grundlegenden Schnittstellentypen (`dss:RequestBaseType` und `dss:ResponseBaseType`) aus [OASIS-DSS] spezifiziert wurden.

## 5. Fehlercodes

Die vorliegende Spezifikation nutzt die folgenden generellen Fehlercodes aus [eCard-1]:

- [../resultmajor#ok](#)
- [../resultmajor#error](#)
- [../resultmajor#warning](#)
- [../resultminor/ar/common#noPermission](#)
- [../resultminor/ar/common#internalError](#)
- [../resultminor/ar/common#parameterError](#)

Darüber hinaus werden zusätzlich die folgenden Fehlercodes definiert:

Fehlercode	Beschreibung
<a href="#">../resultminor/ar/DXAIP_NOK</a>	Die Syntax des beim ArchiveUpdateRequest übergebenen DXAIP-Elements ist nicht korrekt.
<a href="#">../resultminor/ar/DXAIP_NOK_AOID</a>	Die AOID in dem beim ArchiveUpdateRequest übergebenen Delta-XAIP ist nicht bekannt.
<a href="#">../resultminor/ar/DXAIP_NOK_EXPIRED</a>	Das beim ArchiveUpdateRequest übergebene Delta-XAIP-Element kann nicht abgelegt werden, da die Aufbewahrungsfrist abgelaufen ist.
<a href="#">../resultminor/ar/DXAIP_NOK_SUBMTIME</a>	Die beim ArchiveUpdateRequest im übergebenen Delta-XAIP-Element angegebene submissionTime ist nicht korrekt, da sie in der Zukunft liegt.
<a href="#">../resultminor/ar/DXAIP_NOK_SIG</a>	Das beim ArchiveUpdateRequest übergebene Delta-XAIP-Element enthält zumindest eine ungültige Signatur.
<a href="#">../resultminor/ar/DXAIP_NOK_ER</a>	Das beim ArchiveUpdateRequest übergebene Delta-XAIP-Element enthält zumindest einen ungültigen Evidence Record.
<a href="#">../resultminor/ar/DXAIP_NOK_ID</a>	Die beim ArchiveUpdateRequest in einem placeHolder-Element übergebene XML-ID ist im bereits abgelegten XAIP-Element nicht vorhanden.
<a href="#">../resultminor/ar/DXAIP_NOK_Version</a>	Die beim ArchiveUpdateRequest im prevVersion-Element übergebene Version ist nicht die aktuellste Version.
<a href="#">../resultminor/ar/existingAOID</a>	Die im Rahmen des ArchiveSubmissionRequest übergebene AOID existiert bereits.
<a href="#">../resultminor/ar/existingPackageInfoWarning</a>	Bei der ArchiveUpdateRequest-Funktion wird ein DXAIP-Element übergeben, das ein packageInfo-Element enthält. Da im vorher existierenden XAIP bereits das packageInfo-Element belegt war, wird das übergebene packageInfo-Element ignoriert und eine entsprechende Warnung zurückgeliefert.

Fehlercode	Beschreibung
<a href="#">../resultminor/arl/lowSpaceWarning</a>	Diese Warnung gibt an, dass der verfügbare Speicherplatz einen kritischen Wert unterschritten hat.
<a href="#">../resultminor/arl/missingReasonOfDeletion</a>	Da beim ArchiveDeletionRequest kein ReasonOfDeletion-Element übergeben wurde, muss der Löschvorgang abgewiesen werden.
<a href="#">../resultminor/arl/noSpaceError</a>	Diese Fehlermeldung gibt an, dass kein Speicherplatz verfügbar war und deshalb das Archivdatenobjekt nicht abgelegt werden konnte.
<a href="#">../resultminor/arl/notSupported</a>	Diese Fehlermeldung gibt an, dass eine angeforderte Funktion, ein angefordertes Format oder ein übergebener optionaler Eingabeparameter nicht unterstützt wird.
<a href="#">../resultminor/arl/requestOnlyPartlySuccessfulWarning</a>	Diese Warnung gibt an, dass nicht alle angeforderten Daten zurückgeliefert werden konnten.
<a href="#">../resultminor/arl/unknownArchiveDataType</a>	Es wird ein binäres Datenobjekt mit einem nicht unterstützten Datenformat übergeben.
<a href="#">../resultminor/arl/unknownLocation</a>	Die im ArchiveDataRequest angegebene DataLocation ist nicht vorhanden bzw. fehlerhaft.
<a href="#">../resultminor/arl/unknownAOID</a>	Die übergebene AOID existiert nicht.
<a href="#">../resultminor/arl/unknownVersionID</a>	Die übergebene VersionID ist im entsprechenden XAIP nicht bekannt.
<a href="#">../resultminor/arl/XAIP_NOK</a>	Die Syntax des übergebenen XAIP-Elements ist nicht korrekt.
<a href="#">../resultminor/arl/XAIP_NOK_ER</a>	Das übergebene XAIP-Element enthält zumindest einen ungültigen Evidence Record.
<a href="#">../resultminor/arl/XAIP_NOK_EXPIRED</a>	Das übergebene XAIP-Element kann nicht abgelegt werden, da die Aufbewahrungsfrist abgelaufen ist.
<a href="#">../resultminor/arl/XAIP_NOK_SIG</a>	Das übergebene XAIP-Element enthält zumindest eine ungültige Signatur.
<a href="#">../resultminor/arl/XAIP_NOK_SUBMTIME</a>	Die im übergebenen XAIP-Element angegebene submissionTime ist nicht korrekt, da sie in der Zukunft liegt.



## 6. Spezifikation einer Webservice-basierten Schnittstelle

Die Spezifikation der Webservice-basierten Schnittstelle besteht aus zwei Bestandteilen: Zunächst werden die Aufruf- und Rückgabeparameter als XML-Schema [XSD] spezifiziert (vgl. Abschnitt 6.1). Darauf aufbauend wird in einem zweiten Schritt eine Webservice-Spezifikation gemäß [WSDL] entwickelt.

Abschnitt 6.2 enthält die Webservice-Spezifikation der Schnittstelle TR-S.4 (vgl. Abschnitt 3). Die internen Schnittstellen der TR-ESOR-Middleware können bei Bedarf leicht daraus abgeleitet werden, indem nur die benötigte Teilmenge der Funktionen genutzt wird.

Für den Nachweis der Konformitätsstufe 2 müssen die für das oder die Module relevanten Webservice-basierten Schnittstellen gemäß Abschnitt 6.2 unterstützt werden. Darüber hinaus können weitere Schnittstellen, wie z.B. eine sprachgebundene Java- oder C-Schnittstelle gemäß [eCard-1] unter Verwendung der in Abschnitt 6.1 spezifizierten XML-Strukturen unterstützt werden.

### 6.1 Spezifikation der Aufruf- und Rückgabeparameter als XML-Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:tr="http://www.bsi.bund.de/tr-esor/api/1.2"
  xmlns:xaip="http://www.bsi.bund.de/tr-esor/xaip/1.2"
  xmlns:ers="urn:ietf:params:xml:ns:ers"
  xmlns:ec="http://www.bsi.bund.de/ecard/api/1.1"
  xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  targetNamespace="http://www.bsi.bund.de/tr-esor/api/1.2"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <!-- ===== -->
  <!-- Version 1.2 vom 31.01.2015 -->
  <!-- ===== -->

  <import namespace="http://www.bsi.bund.de/tr-esor/xaip/1.2"
    schemaLocation="http://www.bsi.bund.de/SharedDocs/Download/tr-esor/xaip/1_2/tr-esor-xaip-v1_2_xsd" />

  <import namespace="urn:oasis:names:tc:dss:1.0:core:schema"
    schemaLocation="http://ws.openecard.org/schema/oasis-dss-core-schema-v1.0-os.xsd" />

  <import namespace="urn:ietf:params:xml:ns:ers"
    schemaLocation="http://ws.openecard.org/schema/xml-ers-rfc6283.xsd" />

  <import namespace="http://www.bsi.bund.de/ecard/api/1.1"
    schemaLocation="http://ws.openecard.org/schema/eCard.xsd" />

  <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
```

```

        schemaLocation="http://ws.openecard.org/schema/saml-schema-assertion-2.0.xsd" />

<!-- ===== -->
<!--      Uebergreifende Definitionen      -->
<!-- ===== -->

<complexType name="RequestType">
  <complexContent>
    <restriction base="dss:RequestBaseType">
      <sequence>
        <element ref="dss:OptionalInputs"
maxOccurs="1"
                                minOccurs="0" />
      </sequence>
    </restriction>
  </complexContent>
</complexType>

<complexType name="ResponseType">
  <complexContent>
    <restriction base="dss:ResponseBaseType">
      <sequence>
        <element ref="dss:Result" />
        <element ref="dss:OptionalOutputs"
maxOccurs="1"
                                minOccurs="0" />
      </sequence>
    </restriction>
  </complexContent>
</complexType>

<element name="AOID" type="string"/>

<!-- ===== -->
<!--      ArchiveSubmissionRequest      -->
<!-- ===== -->

<complexType name="ArchiveDataType">
  <complexContent>
    <extension base="anyType">
      <attribute name="Type" type="anyURI" />
    </extension>
  </complexContent>
</complexType>

<element name="ImportEvidence" type="tr:ImportEvidenceType"/>

```

```

    <complexType name="ImportEvidenceType">
      <choice>
        <element ref="xaip:evidenceRecord" maxOccurs="unbounded"
minOccurs="1" />
        <element name="CredentialID" type="string"
maxOccurs="unbounded" minOccurs="1" />
      </choice>
    </complexType>

    <element name="ArchiveSubmissionRequest">
      <complexType>
        <complexContent>
          <extension base="tr:RequestType">
            <choice>
              <element ref="xaip:XAIP"></element>
              <element name="ArchiveData"
type="tr:ArchiveDataType"></element>
            </choice>
          </extension>
        </complexContent>
      </complexType>
    </element>

    <element name="ArchiveSubmissionResponse">
      <complexType>
        <complexContent>
          <extension base="tr:ResponseType">
            <sequence>
              <element name="AOID" type="string"
maxOccurs="1"
minOccurs="0">
              </element>
            </sequence>
          </extension>
        </complexContent>
      </complexType>
    </element>

    <!-- ===== -->
    <!-- ArchiveUpdateRequest -->
    <!-- ===== -->

    <element name="ArchiveUpdateRequest">

```

```
<complexType>
  <complexContent>
    <extension base="tr:RequestType">
      <sequence>
        <element ref="xaip:DXAIP"></element>
      </sequence>
    </extension>
  </complexContent>
</complexType>
</element>

<element name="ArchiveUpdateResponse">
  <complexType>
    <complexContent>
      <extension base="tr:ResponseType">
        <sequence>
          <element name="VersionID" type="string"
maxOccurs="1" minOccurs="0"></element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</element>

<!-- ===== -->
<!--   ArchiveRetrievalRequest   -->
<!-- ===== -->

<element name="ArchiveRetrievalRequest">
  <complexType>
    <complexContent>
      <extension base="tr:RequestType">
        <sequence>
          <element name="AOID" type="string" />
          <element name="VersionID" type="string"
maxOccurs="unbounded" minOccurs="0"></element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</element>

<element name="IncludeERS" type="anyURI" />

<element name="ArchiveRetrievalResponse">
```

```

    <complexType>
      <complexContent>
        <extension base="tr:ResponseType">
          <sequence>
            <element ref="xaip:XAIP"
maxOccurs="1" minOccurs="0"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
  </element>

  <!-- ===== -->
  <!--   ArchiveEvidenceRequest   -->
  <!-- ===== -->

  <element name="ArchiveEvidenceRequest">
    <complexType>
      <complexContent>
        <extension base="tr:RequestType">
          <sequence>
            <element name="AOID"
type="string"></element>
            <element name="VersionID" type="string"
maxOccurs="unbounded" minOccurs="0"></element>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
  </element>

  <element name="ERSFormat" type="anyURI" />

  <element name="ArchiveEvidenceResponse">
    <complexType>
      <complexContent>
        <extension base="tr:ResponseType">
          <sequence>
            <element ref="xaip:evidenceRecord"
maxOccurs="unbounded"
minOccurs="0">
          </element>
          </sequence>
        </extension>
      </complexContent>
    </complexType>

```

```

</element>

<!-- ===== -->
<!--   ArchiveDeletionRequest   -->
<!-- ===== -->

<element name="ArchiveDeletionRequest">
  <complexType>
    <complexContent>
      <extension base="tr:RequestType">
        <sequence>
          <element name="AOID"
type="string"></element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</element>

<element name="ReasonOfDeletion">
  <complexType>
    <sequence>
      <element name="RequestorName"
type="saml:NameIDType" />
      <element name="RequestInfo" type="string" />
    </sequence>
  </complexType>
</element>

<element name="ArchiveDeletionResponse" type="tr:ResponseType"/>

<!-- ===== -->
<!--   ArchiveDataRequest   -->
<!-- ===== -->

<element name="ArchiveDataRequest">
  <complexType>
    <complexContent>
      <extension base="tr:RequestType">
        <sequence>
          <element name="AOID"
type="string"></element>
          <element ref="tr:DataLocation"
maxOccurs="unbounded"
minOccurs="1" />
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</element>

```

```

        </extension>
      </complexContent>
    </complexType>
  </element>

  <element name="DataLocation">
    <complexType>
      <complexContent>
        <extension base="anyType">
          <attribute name="Type" type="anyURI" />
        </extension>
      </complexContent>
    </complexType>
  </element>

  <element name="ArchiveDataResponse">
    <complexType>
      <complexContent>
        <extension base="tr:ResponseType">
          <sequence>
            <element name="XAIPData"
              maxOccurs="unbounded"
              minOccurs="1">
              <complexType>
                <sequence>
                  <element
                    ref="dss:Result" maxOccurs="1" minOccurs="1" />
                  <element
                    ref="tr:DataLocation" />
                  <element name="Value"
                    type="anyType" maxOccurs="1" minOccurs="0" />
                </sequence>
              </complexType>
            </element>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
  </element>
</schema>

```

## 6.2 WSDL-Spezifikation der Schnittstelle TR-ESOR-S.4

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://www.bsi.bund.de/tr-esor/api/1.2"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"

```

```
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
xmlns:tr="http://www.bsi.bund.de/tr-esor/api/1.2"
>

<!--=====-->
<!-- Version 1.2 of 31.01.2015
-->
<!--=====-->

<!-- ===== -->
<!-- Definition of types -->
<!-- (only include XSDs) -->
<!-- ===== -->

<wsdl:types>
  <xsd:schema
targetNamespace="http://www.bsi.bund.de/tr-esor/api/1.2"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xaip="http://www.bsi.bund.de/tr-esor/xaip/1.2"
  xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
  elementFormDefault="qualified">
    <xsd:include
schemaLocation="http://www.bsi.bund.de/SharedDocs/Download/tr-esor/api/1_2/
tr-esor-interfaces-v1_2_xsd" />
    </xsd:schema>
  </wsdl:types>

  <!-- ===== -->
  <!-- Definition of messages -->
  <!-- ===== -->

  <!-- ArchiveSubmissionRequest -->

  <wsdl:message name="ArchiveSubmissionRequest">
    <wsdl:part name="parameters"
element="tr:ArchiveSubmissionRequest" />
  </wsdl:message>
  <wsdl:message name="ArchiveSubmissionResponse">
    <wsdl:part name="parameters"
element="tr:ArchiveSubmissionResponse"/>
  </wsdl:message>

  <!-- ArchiveUpdateRequest -->

  <wsdl:message name="ArchiveUpdateRequest">
```



```
        <wsdl:part name="parameters" element="tr:ArchiveUpdateRequest"
/>
    </wsdl:message>
    <wsdl:message name="ArchiveUpdateResponse">
        <wsdl:part name="parameters"
element="tr:ArchiveUpdateResponse"/>
    </wsdl:message>

<!-- ArchiveRetrievalRequest -->

    <wsdl:message name="ArchiveRetrievalRequest">
        <wsdl:part name="parameters"
element="tr:ArchiveRetrievalRequest" />
    </wsdl:message>
    <wsdl:message name="ArchiveRetrievalResponse">
        <wsdl:part name="parameters"
element="tr:ArchiveRetrievalResponse" />
    </wsdl:message>

<!-- ArchiveEvidenceRequest -->

    <wsdl:message name="ArchiveEvidenceRequest">
        <wsdl:part name="parameters"
element="tr:ArchiveEvidenceRequest" />
    </wsdl:message>
    <wsdl:message name="ArchiveEvidenceResponse">
        <wsdl:part name="parameters"
element="tr:ArchiveEvidenceResponse" />
    </wsdl:message>

<!-- ArchiveDeletionRequest -->

    <wsdl:message name="ArchiveDeletionRequest">
        <wsdl:part name="parameters"
element="tr:ArchiveDeletionRequest" />
    </wsdl:message>
    <wsdl:message name="ArchiveDeletionResponse">
        <wsdl:part name="parameters"
element="tr:ArchiveDeletionResponse" />
    </wsdl:message>

<!-- ArchiveDataRequest -->

    <wsdl:message name="ArchiveDataRequest">
        <wsdl:part name="parameters" element="tr:ArchiveDataRequest" />
    </wsdl:message>
```

```
<wsdl:message name="ArchiveDataResponse">
  <wsdl:part name="parameters" element="tr:ArchiveDataResponse"
/>
</wsdl:message>

<!-- VerifyRequest -->

<wsdl:message name="VerifyRequest">
  <wsdl:part name="parameters" element="dss:VerifyRequest" />
</wsdl:message>
<wsdl:message name="VerifyResponse">
  <wsdl:part name="parameters" element="dss:VerifyResponse"/>
</wsdl:message>

<!-- ===== -->
<!-- Definition of portType -->
<!-- ===== -->

<wsdl:portType name="S4">
  <wsdl:operation name="ArchiveSubmission">
    <wsdl:input message="tr:ArchiveSubmissionRequest" />
    <wsdl:output message="tr:ArchiveSubmissionResponse" />
  </wsdl:operation>
  <wsdl:operation name="ArchiveUpdate">
    <wsdl:input message="tr:ArchiveUpdateRequest" />
    <wsdl:output message="tr:ArchiveUpdateResponse" />
  </wsdl:operation>
  <wsdl:operation name="ArchiveRetrieval">
    <wsdl:input message="tr:ArchiveRetrievalRequest" />
    <wsdl:output message="tr:ArchiveRetrievalResponse" />
  </wsdl:operation>
  <wsdl:operation name="ArchiveEvidence">
    <wsdl:input message="tr:ArchiveEvidenceRequest" />
    <wsdl:output message="tr:ArchiveEvidenceResponse" />
  </wsdl:operation>
  <wsdl:operation name="ArchiveDeletion">
    <wsdl:input message="tr:ArchiveDeletionRequest" />
    <wsdl:output message="tr:ArchiveDeletionResponse" />
  </wsdl:operation>
  <wsdl:operation name="ArchiveData">
    <wsdl:input message="tr:ArchiveDataRequest" />
    <wsdl:output message="tr:ArchiveDataResponse" />
  </wsdl:operation>
  <wsdl:operation name="Verify">
    <wsdl:input message="tr:VerifyRequest" />
    <wsdl:output message="tr:VerifyResponse" />
  </wsdl:operation>
</wsdl:portType>
```

```

        </wsdl:operation>

</wsdl:portType>

<!-- ===== -->
<!-- Definition of Binding -->
<!-- ===== -->

<wsdl:binding name="S4" type="tr:S4">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="ArchiveSubmission">
    <soap:operation
      soapAction="http://www.bsi.bund.de/tr-esor/ArchiveS
ubmission" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ArchiveUpdate">
    <soap:operation
      soapAction="http://www.bsi.bund.de/tr-esor/ArchiveU
pdate" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ArchiveRetrieval">
    <soap:operation
      soapAction="http://www.bsi.bund.de/tr-esor/ArchiveR
etrieval" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ArchiveEvidence">
    <soap:operation
      soapAction="http://www.bsi.bund.de/tr-esor/ArchiveE
vidence" />
    <wsdl:input>

```

```

        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ArchiveDeletion">
    <soap:operation
        soapAction="http://www.bsi.bund.de/tr-esor/ArchiveD
eletion" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ArchiveData">
    <soap:operation
        soapAction="http://www.bsi.bund.de/tr-esor/ArchiveD
ata" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Verify">
    <soap:operation
        soapAction="http://www.bsi.bund.de/tr-esor/Verify"
/>
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>

<!-- Definition of Support-Service -->

<wsdl:service name="S4">
    <wsdl:port name="S4" binding="tr:S4">
        <soap:address location="http://127.0.0.1:18080" />
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```