GOG-Archive Schnittstellenspezifikation Version 1.1 - Final

Stand: 19. November 2009

Verfasser:	Mag. Wolf Dieter Auer	**Abteilung:**	E-DC-AI	**Tel.:**	+43/ (0)1 / 711 23
Projekt:	GOG-Archive	**Titel:**	Schnittstellenspezifikation Version 1.1	**Version:**	1.1.0.014
Erstelldatum:	04.09.2009	**Änderungsdatum:**	30.11.2009	**Druckdatum:**	30.11.2009
Verfassende Stelle:	Bundesrechenzentrum GmbH.	**Empfangende Stelle:**	BMJ		

Inhaltsverzeichnis

1 Änderung vom 23.3.2009

1.1 Ergänzende technische Regeln

1) Regelung für die Verschlüsselung der Metadatenfeldern "JusticDeedType", "Subject" und "Description".
2) Definition eines Flags zur Kennzeichnung von Testrequests
3) Empfehlung für die Verschlüsselung von ValueTypes
4) Empfehlung des Einsatzes der "exclusiv c14n"-Canonisierung
5) Konvention für die Ermittlung des Sessionkeys
6) XADES Version http://uri.etsi.org/01903/v1.1.1/ ist zu verwenden
7) XMLDSig der Archivsignatur muss ein Element SigningTime aus dem XADES Namespace http://uri.etsi.org/01903/v1.1.1/ enthalten
8) Neue Client Fehlermeldungen

2 Änderung vom 23.1.2009

2.1 Anforderungen an PDF Dokumente

1) PDFs müssen im Arcobat PDF 1.4 Format vorliegen und zusätzlich dem ISO-Standard *ISO-19005-1 - Document management – Electronic document file format for long-term preservation – Part 1: Use of PDF 1.4 (PDF/A-1) - Level B Conformance* entsprechen.

3 Änderung vom 25.9.2008

3.1 Metadaten

1) Berechnung der Seitenanzahl:

   Die Anzahl der Seiten eines Dokumentes ist unabhängig vom Format der einzelnen Seiten zu übergeben.

   Anmerkung:
   Bei der Archivkoordinationssitzung vom 25.9.2008 wurde das Problem der Berechnung der Seitenanzahl ausführlich diskutiert:

   Da bei Gerichten der Ausdruck von A4-Seiten verrechnet wird, wird gewünscht, dass die GOGArchive die für den Ausdruck des Dokumentes auf A4-Seiten erforderliche Seitenzahl übergeben. Das bedeutet, dass Seiten größer A4 auf A4 umgerechnet werden müssen.

   Dies ist aufwändig, da das gesamte Dokument durchgescannt werden muss, um die einzelnen Seitenformate zu ermitteln. Bei cyberDOC und Archivium ist dies nur am Client möglich und würde den Archivierungsvorgang unzumutbar verlangsamen.

   Für die Zukunft wird angestrebt, dass bei Gericht keine Pläne in Originalgröße ausgedruckt werden und auf das zuständige Vermessungsamt verwiesen wird.

   Wie schon bisher wird die Anzahl der Seiten eines Dokumentes unabhängig vom Format der einzelnen Seiten übergeben.

4 Änderungen vom 16.9.2008

4.1 Eingebettete PDF-Signaturen

Die Forderung, dass eingebetteten Signaturen vom Typ "PDF-Amtssignatur" sein müssen, wird aus der Spezifikation genommen (Spez. 2.0.0)

4.2 Archivsignatur

Die Archivsignatur muss vom Typ XMLDSig mit den in **R0001** definierten Einschränkungen sein.

4.3 Einstellungsinformationen

Das Pflichtelement `gsig:InsertionSignature` enthält entweder

den Zeitstempel der Einstellung der Urkunde ins Archiv plus dem Zertifikat des Einstellers

oder eine Einstellungssignatur vom Typ XMLDSig mit den in **R0001** definierten Einschränkungen

4.4 Erstellung und Prüfung von Signaturen (R0001)

Die Archivsignatur und die Einstellungssignatur müssen grundsätzlich einer *Detached Signature* laut XMLDSIG [2] entsprechen.

Weiters muss die Archivsignatur den nachfolgend gelisteten Einschränkungen (Profilierung) genügen:

2) Die Archivsignatur muss im Element `dsig:SignedInfo` zumindest ein Element `dsig:Reference` enthalten. Dieses Element muss in seinem Attribut `URI` einen Wert enthalten, der folgendem Aufbau genügt: `file:<Dateiname>.<Extension>` (also z.B. file:Urkunde.pdf).

3) Die Archivsignatur darf im Element `dsig:SignedInfo` neben der Referenz auf die PDF-Urkunde weitere `dsig:Reference` Elemente enthalten, um z.B. Signaturattribute innerhalb des die XMLDSIG-Signatur repräsentierenden XML-Dokuments zu referenzieren. In einem solchen Fall muss die unter (1) erläuterte Referenz jedoch das erste `dsig:Reference` Element in `dsig:SignedInfo` sein.

4) Die Archivsignatur muss im Element `dsig:KeyInfo` genau ein Element `dsig:X509Data` enthalten. Dieses Element muss zumindest ein Element `dsig:X509Certificate` enthalten, das als Textinhalt das Signaturzertifikat des Archivs enthält. `dsig:X509Data` darf darüber hinaus weitere `dsig:X509Certificate` Elemente enthalten, um z.B. weitere Zertifikate für die Zertifikatskettenbildung zu transportieren. In einem solchen Fall muss das `dsig:X509Certificate` mit dem Signaturzertifikat jedoch das erste innerhalb von `dsig:X509Data` sein.

4.5 Metadaten

5) Größe der Urkunde in Bytes
Ist die Filegröße des Klartextes, binär codiert (nicht base64)

6) Geschäftszahl der Justiz wird nicht mehr übertragen
Element "gmsg:Identification" im getDeedRequest wird nicht mehr mitgeführt

4.6 Operation "SendIsReleased"

Die Operation "SendIsReleased" kann in dieser Form nicht realisiert werden.

4.7 Operation "CloseDeed"

Eine explizite Operation "closeDeed" wurde spezifiziert. Mit Hilfe der Operation muss der Client bekannt geben, dass er keine weiteren Blöcke beim Server anfordern wird (z.B. weil Urkunde zu groß ist, oder weil Download bereits abgeschlossen ist).

4.8 Umbenennung auf "SendErrorNotification"

Die Operation "SendNotification" wurde auf "SendErrorNotification" umbenannt.

5 Transport von Dokumenten

Es können sowohl Dokumente im PDF- als auch im TIFF-Format verarbeitet werden. Wobei folgende Konventionen einzuhalten sind

5.1 TIFF Dokumente

Die BRZ GmbH unterstützt das Tagged Image File Format der Version 6.0 vom 3. Juni 1992

Ausgenommen von der Unterstützung sind die Features "**planar images**" und "**extra samples**"

5.2 PDF Dokumente

7) Alle GOG Archive verpflichten sich PDF-Dokumente, beim Einstellen in das Archiv gegen PDF/A-1b zu validieren. Hierbei obliegt es dem Archivbetreiber, ob ein Validator oder ein PDF zu PDF/A-1b Konverter eingesetzt wird.

8) Für die Validierung bzw. Konvertierung wird kein spezielles Tool vorgeschrieben, jedoch muss die vom Archivbetreiber eingesetzte Software den **Isartor PDF/A-1b Test** erfüllen.

9) Im Gegenzug vertraut die Justiz darauf, dass die in den Archiven erfolgreich eingestellten Dokumente PDF/A-1b entsprechen. D.h. beim Abholen der Dokumente durch die Justiz werden die Dokumente nicht nochmals validiert.

5.2.1 Isartor Test Suite

Das PDF/A Competence Center hat im August 2008 die Isartor Test Suite vorgestellt. Unter dem Motto "Validiere die Validatoren" wurden rund 200 Testfälle definiert. Zu jedem Testfall gibt es ein fehlerhaftes PDF, welches der Validator als fehlerhaft erkennen muss.

Wie bereits erwähnt steht das Testprogramm seit August 2008 den Toolherstellern zur Verfügung, um ihre PDF/A-1b Validatoren oder PDF zu PDF/A-1b Konverter entsprechend testen und anpassen zu können. Es ist damit zu rechnen, dass Ende 2009 eine breite Palette von Tools zur Verfügung steht, die dem Isartor PDF/A-1b Test vollständig erfüllt.

Die Suite kann unter http://www.pdfa.org/doku.php?id=pdfa:en:isartor_test_suite:download herunter geladen werden.

5.2.2 PDF/A Competence Center

Das PDF/A Competence Center (www.pdfa.org) ist ein Zusammenschluss weltweit führender Unternehmen und Fachleute im Bereich PDF-Technologie. Der Zweck des PDF/A Competence Centers ist die Förderung des Informations- und Erfahrungsaustausches auf dem Gebiet Langzeitarchivierung gemäß ISO 19005: PDF/A.

Österreich ist mit der digitalen Langzeitarchivierungsinitiative (dig:LA) des Bundeskanzleramts vertreten.

5.2.3 PDF/A – Ein Standard für die Langzeit-Archivierung

Am 28. September 2005 hat die Internationale Organisation für Standardisierungen (ISO) einem neuen Standard für die Regelung der Archivierung elektronischer Dokumente zugestimmt: *ISO-19005-1 - Document management – Electronic document file format for long-term preservation – Part 1: Use of PDF 1.4 (PDF/A-1).*

Dieser Standard ist das Ergebnis einer über dreijährigen Sitzungsarbeit von Vertretern aus weltweit ansässigen Unternehmungen und anderen Organisationen. PDF/A-1 wiederum ist weiter unterteilt in die Übereinstimmungsgraden PDF/A-1a und PDF/A-1b.

PDF/A-1a (Level A Conformance) bezeichnet die vollständige Übereinstimmung mit dem PDF/A Standard ISO 19005-1: Part 1.

PDF/A-1b (Level B Conformance) bezeichnet die Mindestanforderungen zur Übereinstimmung mit PDF/A. Die PDF/A-1b Anforderungen sollten für die visuelle Langzeit-Reproduktion genügen und gelten auch für die GOG Archive Schnittstelle.

5.2.4 Unterschied PDF und PDF/A

Das PDF-Format an sich garantiert keine Langzeit-Reproduzierbarkeit, nicht einmal das Prinzip WYSIWYG (*what you see is what you get*). Damit beides gewährleistet ist, mussten gewisse Einschränkungen und Erweiterungen in den Standard aufgenommen werden. Ferner, um bei einem breiten Publikum akzeptiert zu werden, musste PDF/A auf einer bereits existierenden PDF-Version aufbauen. Das ISO TC 171 hat Adobe's **PDF Referenz 1.4** (von Adobe implementiert in der Acrobat 5-Version) als **Grundlage des Standards** gewählt. Der ISO-Standard sagt aus, dass PDF/A „alle Anforderungen der PDF Referenz erfüllen muss, wie durch diesen Teil des ISO 19005 Standards ergänzt". Der Standard beschreibt also nur die Unterschiede zur Referenz. Um PDF/A vollständig zu verstehen, muss also auch die PDF Referenz 1.4 verstanden werden.

Bestimmte, in PDF 1.4 erlaubte Funktionalität, wie die Transparenz oder die Ton- und Videoreproduktion, sind aus PDF/A ausgeschlossen worden. Es gibt andererseits in PDF 1.4 optionale Konstrukte, welche in PDF/A vorhanden sein müssen. So müssen in PDF/A beispielsweise alle verwendeten Schriften eingebettet sein. Kurzum, PDF/A präzisiert im Wesentlichen spezifische Eigenschaften der PDF Referenz 1.4 und definiert ob sie obligatorisch, empfohlen, eingeschränkt oder verboten sind.

5.2.5 Hauptunterschied zwischen PDF/A-1a und PDF/A-1b

PDF/A-1b stellt sicher, dass Dokumente ohne externe Ressourcen visualisiert werden können. Im Unterschied dazu müssen bei PDF/A-1a zusätzlich alle Graphiken OCR gescannt werden und der Text und dessen Struktur zusätzlich in das PDF eingebettet werden. Damit kann der Inhalt des Dokuments mittels Transkription Blinden vorgelesen bzw. in Braille Schrift übertragen werden.

5.3 Container PDF Dokumente werden nicht unterstützt

Auf Grund organisatorischer und rechtlicher Gründe ist es dem Bundesministerium für Justiz nicht möglich Container-PDF Dokumente - wie sie vom BAIK-Archiv erstellt werden - zu unterstützen.

5.3.1 Was ist ein Container PDF Dokumente

Ein Container PDF ist vergleichbar einem Gerichtsakt in dem alle zu einem Geschäftsfall gehörenden Dokumente zusammengefasst sind. Ein Container PDF Dokument ist ein ganz normales PDF Dokument in das alle zu einem Geschäftsfall gehörenden Dokumente als **Anhang** eingebettet sind. Auf einem Deckblatt - versehen mit einer Amtssignatur - werden alle eingebetteten Dokumente aufgelistet.

5.3.2 Regelung für die Konvertierung von Container-PDF Dokumenten

10) Container-PDFs sind nicht zulässig

11) Jedes einzelne Dokument eines Container PDFs muss über die gegenständlich spezifizierte Schnittstelle im TIFF oder PDF Format abrufbar sein.

12) Für jedes einzelne Dokument des Container-PDFs (Urkunde, Anhänge) muss ein eindeutiger Identbegriff vergeben werden.

5.4 Berechnung der Seitenanzahl

Die Anzahl der Seiten eines Dokumentes ist weiterhin - unabhängig vom Format der einzelnen Seiten - zu bestimmen und in den Metadaten bereitzustellen.

5.5 Übertragung der Urkunden in Blöcken

Statt die Urkunde wie bisher als Attachment zu versenden, wird sie in einzelne Blöcke zerlegt und direkt im SOAP-Body base64-kodiert transportiert.

Diese Maßnahme hat folgende Vorteile:

13) Die Systeme laufen stabiler, da die Antwortzeiten für einzelnen Requests sehr kurz und konstant sind.

14) Große Urkunden blockieren nicht die gesamte Verarbeitung, da auch andere Abfragen eine Chance bekommen parallel abgearbeitet zu werden.

15) Selbst Urkunden im Gigabytebereich können problemlos transportiert werden.

6 Konventionen für die XML-Verarbeitung

6.1 Archivsignatur

6.1.1 Canonisierung mit "exclusiv c14n"

Um Problemen bei der Prüfung von XMLDSig Signaturen vorzubeugen, empfehlen wir den Einsatz der "Exclusive XML Canonicalization" (Namespace http://www.w3.org/TR/xml-exc-c14n/).

Dessen ungeachtet wird die Canonicalization c14n (Namespace http://www.w3.org/TR/xml-c14n) weiterhin unterstützt.

6.1.2 XADES Namespace

Für die Archivsignaturen ist die XADES Version http://uri.etsi.org/01903/v1.1.1/ zu verwenden.

6.1.3 SigningTime

Die Archivsignatur im XMLDSig Format muss auf dem XPath
`/dsig:Signature/dsig:Object/etsi:QualifyingProperties/etsi:SignedProperties/etsi:SignedSignatureProperties/etsi:SigningTime`
den Zeitpunkt der Archivsignaturerstellung enthalten.

6.2 Property-Value Struktur

Die Property-Value Struktur dient dazu beliebige Daten in der Form <Feldname> = <Wert> transportieren zu können. Vereinfacht dargestelltes Beispiel: "Geburtsdatum" = "1957.03.16"

Mit Hilfe dieser Technik können im Nachhinein fachliche Änderungen an den Metadaten festgelegt werden, ohne dass die Schnittstellenspezifikation überarbeitet werden muss.

Dieses Konstrukt kommt erst zum Einsatz, wenn nach Finalisierung der Spezifikation zusätzliche Metadaten definiert werden müssen oder archivbetreiberspezifische Metadaten transportiert werden müssen.

6.3 Verschlüsselung von Properties

Wenn die einzelnen Values einer Property-Value Struktur verschlüsselt werden müssen, empfehlen wir vor dem Verschlüsseln die **Namespaces** aus dem XML-Content **zu entfernen**. Dies hat allein den Zweck bei etwaigen, zukünftigen Änderungen den verschlüsselten Inhalt leichter verarbeiten zu können.

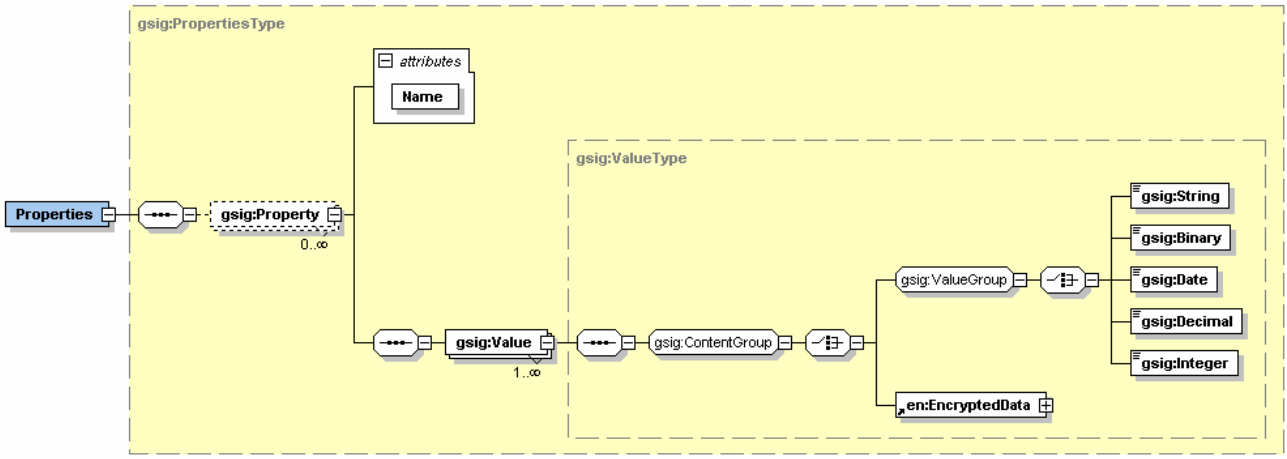

Abbildung 1: Struktur zur Abbildung der Liste der Property-Values

Die aktuelle Implementierung des Clients kann folgenden Inhalt korrekt als **gsig:ValueType** interpretieren:

Variante #1 - einfachste und empfohlen Variante:

```
<String>Das ist ein Test</String>
```

Variante #2:

```
<String xmlns ="http://brz.gv.at/GOGArchive/Signature/v1.0#">
    Das ist ein Test
</String>
```

Variante #3:

```
<gsig:String xmlns:gsig="http://brz.gv.at/GOGArchive/Signature/v1.0#">
    Das ist ein Test
</gsig:String>
```

Variante #4:

```
<?xml version="1.0" encoding="UTF-8"?>
<gsig:String xmlns:gsig="http://brz.gv.at/GOGArchive/Signature/v1.0#">
    Das ist ein Test
</gsig:String>
```

6.4 Verschlüsseln von Metadaten

Bei der Verschlüsselung der Metadatenfelder

- gsig:JusticeDeedType
- gsig:Subject
- gsig:Description

gilt die gleiche Regelung bei der Verschlüsselung der Properties. Zusätzlich darf die Angabe des Typs entfallen.

Beispielsweise kann für den Gegenstand der Urkunde (Justiz) (**gsig:JusticDeedType**) statt

```
<String> Gesellschaftsvertrag </String>
```

auch nur

Gesellschaftsvertrag

angeben werden.

6.5 Ermittlung des Sessionkeys

16) Der SessonKey wird auf dem XPath **`gmsg:GetDeedResponse/gsig:ArchiveSignature/gsig:DeedEncryption/en:EncryptedKey`** erwartet.

17) Der Verschlüsselungsalgorithums muss auf dem XPath **`gmsg:GetDeedResponse/gsig:ArchiveSignature/gsig:DeedEncryption/en:EncryptedKey/en:EncryptionMethod/@Algorithm`** angegeben sein.

6.6 Testrequest

Um Testrequests kennzeichnen zu können, wird ein Property mit dem Namen "**TestRequest**" vom Typ "**Integer**" definiert.

- Wertebereich: 1=TRUE, 0=FALSE
- Wenn es sich um keinen Testrequest handelt, kann die Angabe des Propertys entfallen.

Beispiel eines XML-Fragments, welches das Property "**TestRequest**" enthält:

```
<gsig:Property xmlns:gsig="http://brz.gv.at/GOGArchive/Signature/v1.0#" Name="TestRequest">
    <gsig:Value>
        <gsig:Integer>1</gsig:Integer>
    </gsig:Value>
</gsig:Property>
```

7 Workflow

Anmerkung: Für den im Folgenden vorgestellten Verarbeitungsprozess wird von der BRZ eine Referenzimplementierung in Java zur Verfügung gestellt. Die Referenzimplementierung zeigt lediglich wie Urkunden blockweise zum Client transferiert werden. Die erforderlichen kryptographischen Operationen werden nicht gezeigt.

7.1 Client

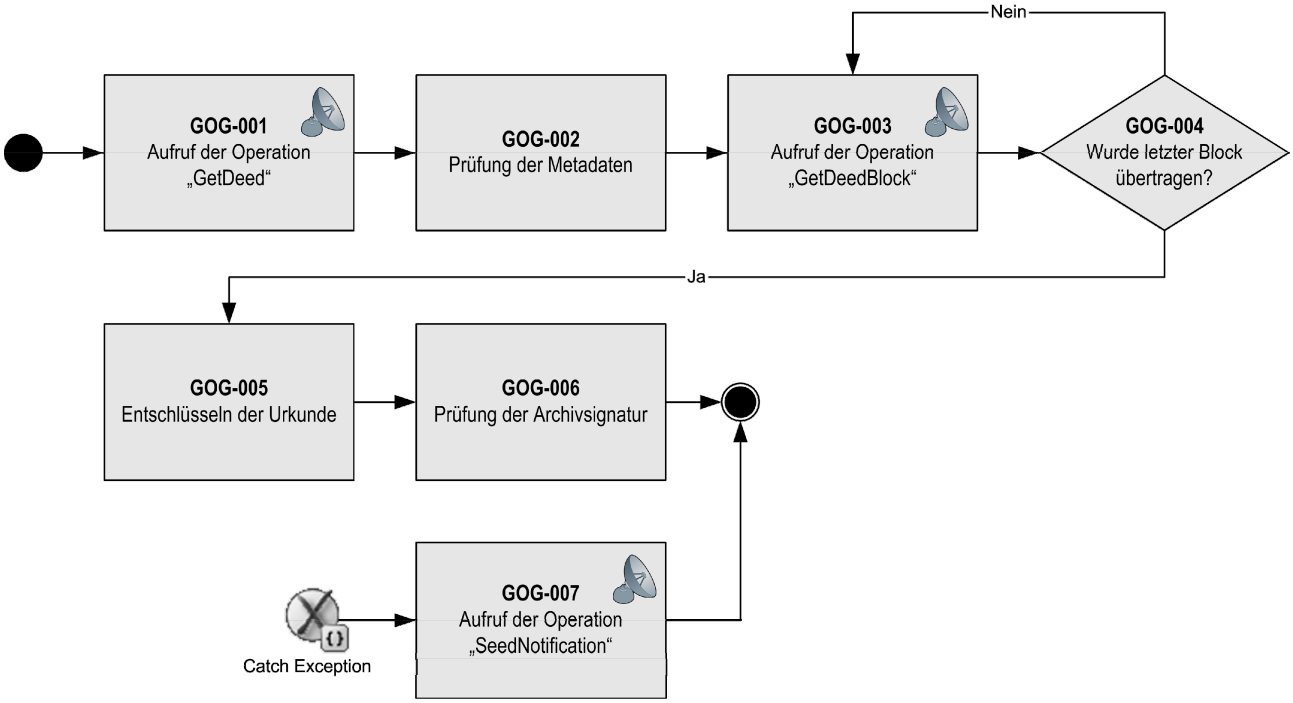

Abbildung 2: Ablauf der Urkundenabholung inklusive Ausnahmebehandung

7.1.1 Abholung der Urkunde

Aufruf der Operation "GetDeed" (GOG-001)
Der Client fordert die Urkunde mit der Webservicemethode "`getDeed()`" beim Server an.

Der Server holt die Urkunde aus dem Archiv, speichert sie temporär und
gibt dem Client einen Handle für die Urkunde zurück (vgl. Abschnitt Server (unten)).

Prüfung der Metadaten (GOG-002)
Der Client prüft die Metadaten auf Vollständigkeit

Aufruf der Operation "GetDeedBlock" (GOG-003)
Mit dem Handle - eine Referenz auf die abzuholende Urkunde auf Serverseite - wird die Urkunde blockweise abgeholt. Als Antwort auf den Aufruf der Operation erhält der Client einen ca. 64kByte großen Datenblock und die Nummer des nächsten abzuholenden Blocks.

Wurde letzter Block übertragen? (GOG-004)
Wenn die nächste Blocknummer gleich 0 ist, wurde die Urkunde vollständig übertragen und kann weiter verarbeitet werden.

Entschlüsselung der Urkunde (GOG-005)
Wenn in der XML Struktur am XPath
`/gsig:ArchiveSignatur/gsig:DeedEncryption/en:EncryptedKey` ein verschlüsselter SessionKey angegeben ist, ist die Urkunde zu entschlüsseln. Anderenfalls wurde die Urkunde im Klartext übertragen.

Prüfung der Archivsignatur (GOG-006)
Wenn in der XML-Struktur eine XMLDSig Signatur enthalten ist, wird diese verwendet, um die Archivsignatur zu pürfen.

Wenn in der XML-Struktur eine Transformationsvorschrift enthalten ist, muss die Archivsignatur aus dem PDF extrahiert und geprüft werden.

7.1.2 Ausnahmebehandlung

Aufruf der Operation "SendErrorNotification" (GOG-007)
Wenn ein definierter fachlicher Fehler auftritt, wird dieser mit Hilfe der Webservice Operation "SendErrorNotification" an den Archivbetreiber zurückgemeldet. Die Methode muss am Webserviceserver implementiert sein, es obliegt jedoch dem Archivbetreiber diese Information auszuwerten.

Typische Fehlermeldungen wären beispielsweise "*Archivsignatur ist ungültig*" oder "*TIFF konnte nicht in PDF konvertiert werden*".

7.2 Server

7.2.1 GetDeed

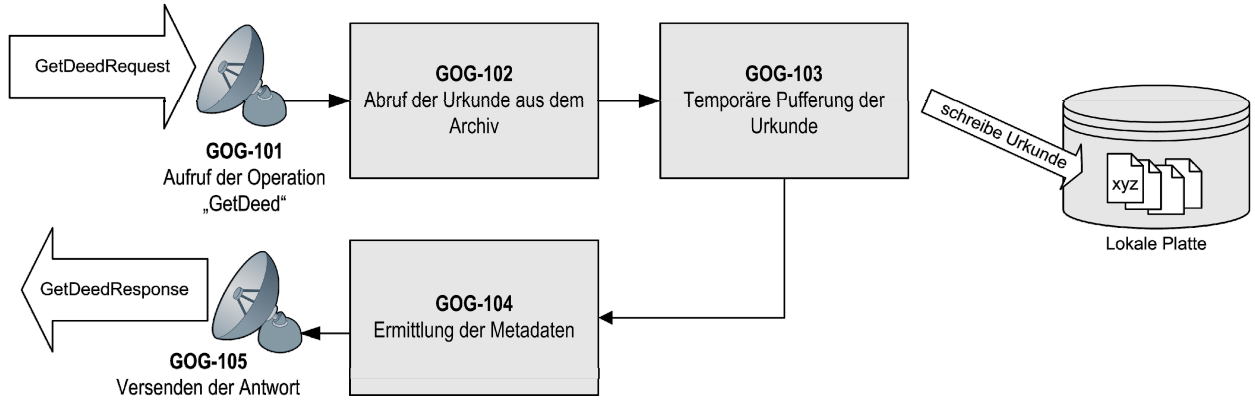

Abbildung 3: Verarbeitung einer Urkundenanforderung

Aufruf der Operation "GetDeed" (GOG-101)
Der Webservice Request wird vom Server entgegengenommen und die Syntax des Identbegriffs überprüft.

Wenn der Identbegriff ungültig ist, wird eine Fehlermeldung zurückgegeben.

Abruf der Urkunde aus dem Archiv (GOG-102)
Die Urkunde wird aus dem Archiv ausgelesen und mit der Archivsignatur versehen.

In der ersten Ausbaustufe des Webservices muss bei PDF-Dokumenten neben der eingebetteten Archivsignatur zusätzlich eine zweite Archivsignatur erstellt werden, deren Hashwert über das gesamte PDF-Dokument berechnet wird.

Wenn es keine Urkunde zum angegebenen Identbegriff gibt, wird eine Fehlermeldung zurückgegeben.

Temporäre Pufferung der Urkunde (GOG-103)
Die Urkunde wird auf der lokalen Platte des Webserviceservers für die weitere Verarbeitung gespeichert. Eine Referenz auf das File wird in einer Queue im Hauptspeicher gehalten.

Designhinweis:

Wenn das Webservice-Server lastverteilt auf mehreren Maschinen aufgesetzt wird, muss die Referenz auf die Urkunde (Element `gmsg:Handle`) zusätzlich eine Routing-Information enthalten. Diese ist notwendig, um Aufrufe der Operation `GetDeedBlock` an jene Webservice-Server-Instanz bzw. Maschine zu routen auf der die Urkunde lokal gepuffert ist.

Ermittlung der Metadaten (GOG-104)
Das Dokument egal ob TIFF oder PDF muss geparst werden, um die Seitenanzahl korrekt ermitteln zu können (siehe **R2005** unten).

Speziell bei PDFs muss die eingebettete Beurkundungssignatur extrahiert werden, um das Zertifikat und der Zeitstemplung der Beurkundung auszulesen.

Versenden der Antwort (GOG-105)
Die Metadaten plus die Referenz auf die lokal gepufferte Urkunde (`gmsg:Handle`) werden an den Client zurückgesandt.

7.2.2 GetDeedBlock

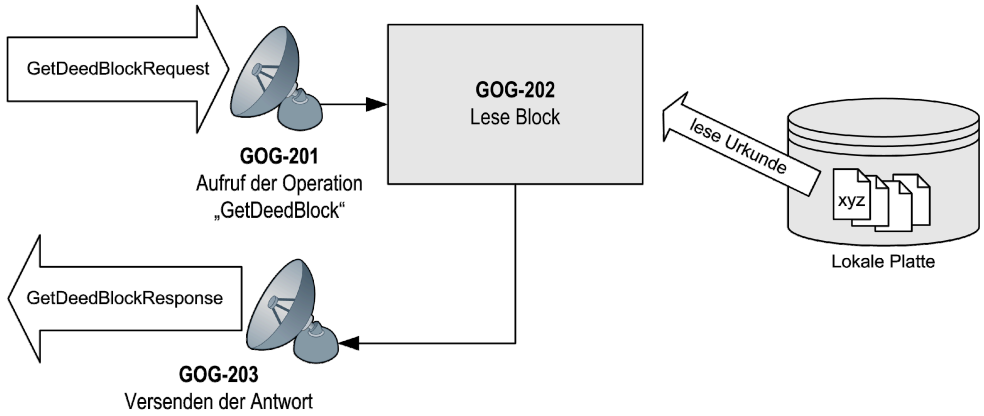

Abbildung 4: Verarbeitung eines Blockrequests

Aufruf der Operation "GetDeedBlock" (GOG-201)
Wird der Aufruf der Operation "GetDeedBlock" empfangen, kontrolliert der Server, ob der Handle noch gültig ist.

Lese Block (GOG-202)
Mit Hilfe des Handles wird die File-Referenz der Urkunde ermittelt und der angeforderte Block gelesen.

Wenn der letzte Block gelesen wurde, kann die Urkunde aus dem Filesystem gelöscht werden.

Versenden der Antwort (GOG-203)
Der Block plus die nächste Blocknummer werden an den Client zurückgesandt.

Wenn der letzte Block versendet wird, muss der Wert der nächsten Blocknummer gleich 0 sein.

8 Webservicedefinition & Datenstrukturen

8.1 Namespaces

Präfix	Namespace
gws	http://www.brz.gv.at/GOGArchive/Webservice/v1.0/20080604#
gmsg	http://www.brz.gv.at/GOGArchive/Message/v1.0/20080604#
gsig	http://www.brz.gv.at/GOGArchive/Signature/v1.0/20080604#
ds	http://www.w3.org/2000/09/xmldsig#
en	http://www.w3.org/2001/04/xmlenc#
etsi	http://uri.etsi.org/01903/v1.1.1#
xs	http://www.w3.org/2001/XMLSchema

8.2 GOGArchiveWebservice.wsdl

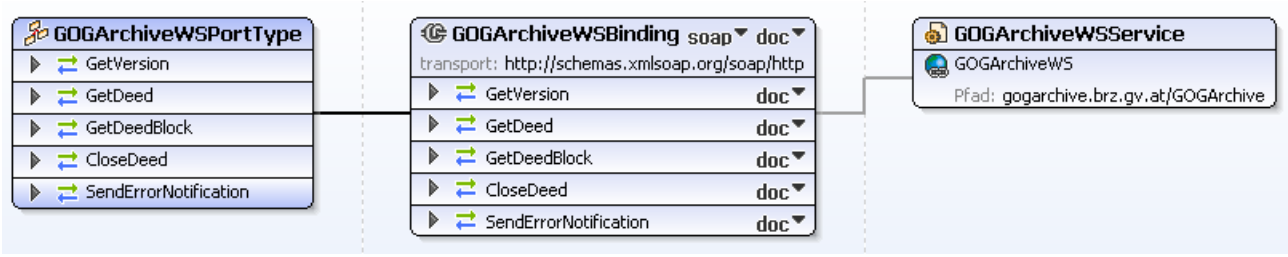

Abbildung 5: Überblick über die Webservice Definition

8.2.1 Operation "GetVersion"

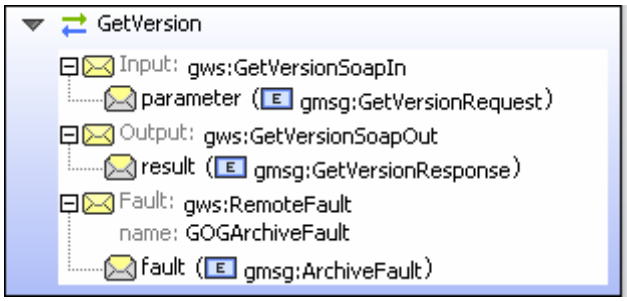

Abbildung 6

Die Operation "GetVersion" liefert die Versionsnummer des Webservices zurück. Die Operation ist rein grundsätzlich für erste Verbindungstests des Webservices gedacht, um mit wenig Aufwand einen ersten Request absetzen zu können.

Input	Output	Fault
gmsg:GetVersionRequest	gmsg:GetVersionResponse	ArchiveFault

8.2.2 Operation "GetDeed"

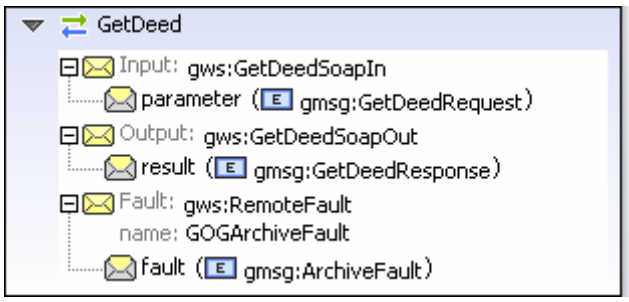

Abbildung 7

Dient der Abfrage einer Urkunde aus einem GOG-Archiv.

Input	Output	Fault
gmsg:GetDeedRequest	gmsg:GetDeedResponse	ArchiveFault

8.2.3 Operation "GetDeedBlock"

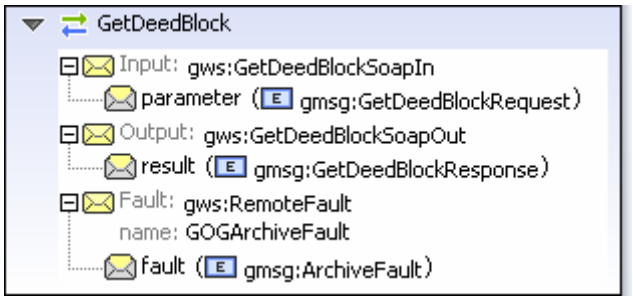

Abbildung 8

Mit Hilfe der Operation "GetDeedBlock" wird die Urkunde Stück für Stück base64-codiert zum Client übertragen.

Input	Output	Fault
gmsg:GetDeedBlockRequest	gmsg:GetDeedBlockResponse	ArchiveFault

8.2.4 Operation "SendErrorNotification"

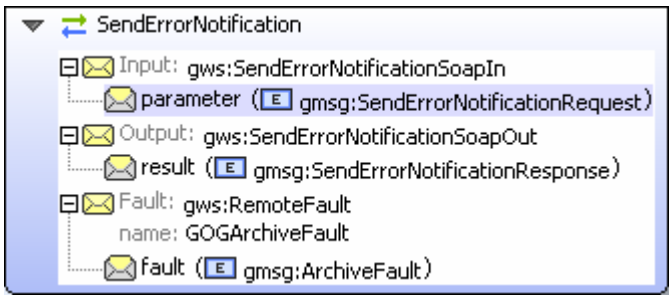

Abbildung 9

Die Operation "SendErrorNotification" wird dazu verwendet fachliche Fehler, die bei der Nachverarbeitung der Urkunde erkannt werden, an den GOG Archivbetreiber zurückzumelden.

Beispiele hierfür wären "*Urkunde entspricht nicht PDF/A-1b*" oder "*Archivsignatur konnte nicht geprüft werden*".

Input	Output	Fault
gmsg:SendErrorNotificationRequest	gmsg:SendErrorNotificationResponse	ArchiveFault

8.2.5 Operation "CloseDeed"

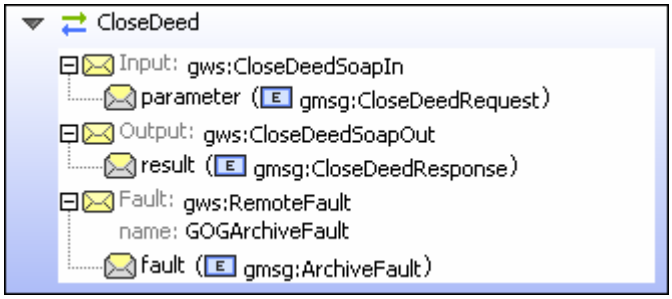

Abbildung 10

Mit Hilfe der Operation "CloseDeed" muss der Client bekannt geben, dass er keine weiteren Blöcke beim Server anfordern wird (z.B. weil Urkunde zu groß ist, oder weil Download bereits abgeschlossen ist).

Input	Output	Fault
gmsg:CloseDeedRequest	gmsg: CloseDeedResponse	ArchiveFault

8.3 GOGArchiveMessage.xsd

8.3.1 gmsg:GetVersionRequest

Abbildung 11

Der Request ist leer und enthält keine Parameter. Diese Konstruktion dient allein dazu den Webservice Generator zufrieden zu stellen.

8.3.2 gmsg:GetVersionResponse

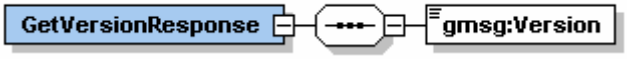

Abbildung 12

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Version	gmsg:VersionNumberTape	✓	Das Versionselement erwartet eine Versionsnummer entsprechend der Regular Expression "[0-9]{1,2}\.[0-9]{1,2}\.[0-9]{1,2}" z.B. "1.0.0", "1.10.61", etc..

8.3.3 gmsg:GetDeedRequest

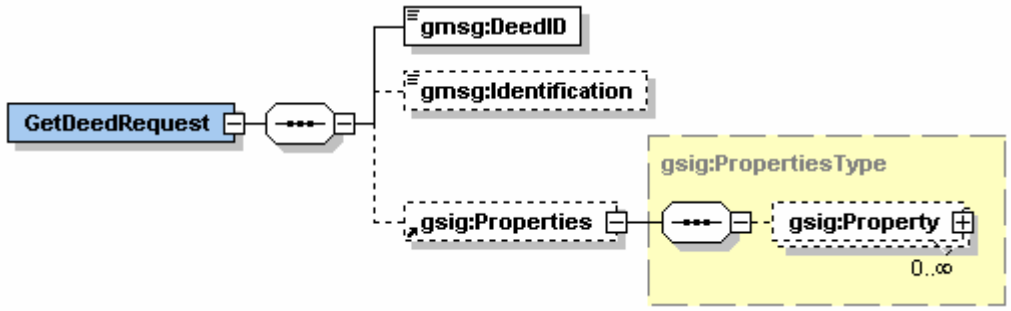

Abbildung 13

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:DeedID	string	✓	Der Urkundenidentbegriff ist die ID der Urkunde, wie sie vom Archivbetreiber festgelegt wurde
gmsg:Identification	string		Geschäftszahl der Justiz
gsig:Properties	gsig:PropertiesType		siehe GOGArchiveSignature.xsd

8.3.4 gmsg:GetDeedResponse

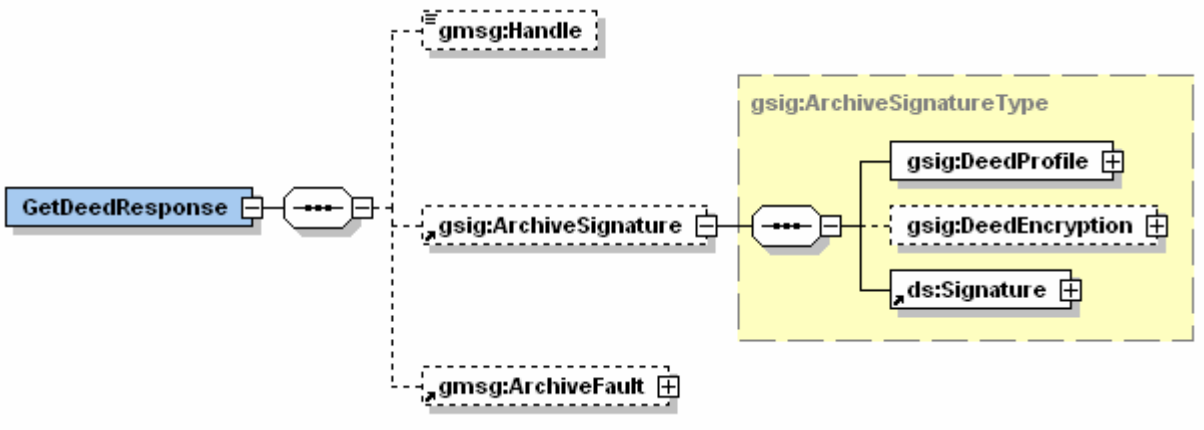

Abbildung 14

Regeln für den Aufbau des gmsg:GetDeedResponse (**R3001**)

- Wenn der Request erfolgreich verarbeitet werden konnte, muss sowohl ein Element `gmsg:Handle` als auch ein Element `gsig:ArchiveSignature` zurückgegeben werden.
- Wenn ein fachlicher Fehler zurückgemeldet wird, darf nur ein Element `gmsg:ArchiveFault` zurückgegeben werden.

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Handle	string		gmsg:Handle ist eine serverseitige Referenz auf die Urkunde, welche für den Download gepuffert wird.
gsig:ArchiveSignature	gsig:ArchiveSignatureType		Siehe GOGArchiveSignature.xsd
gmsg:ArchiveFault	Element		Fachliche Fehlermeldung (R3003)

8.3.5 gmsg:GetDeedBlockRequest

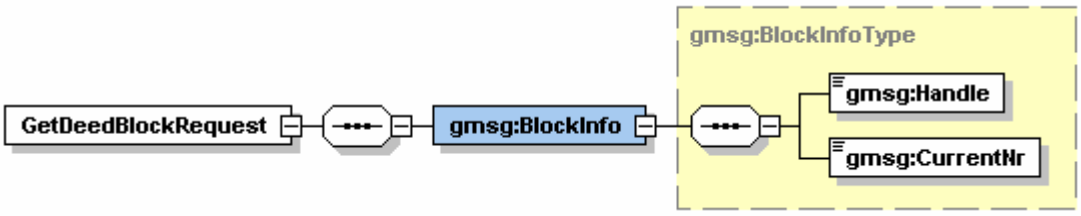

Abbildung 15

gmsg:BlockInfo

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Handle	string	✓	gmsg:Handle ist eine serverseitige Referenz auf die Urkunde, welche für den Download gepuffert wird.
gsig:CurrentNr	long	✓	Nummer des aktuellen Blocks, der beim Server angefordert wird.

8.3.6 gmsg:GetDeedBlockResponse

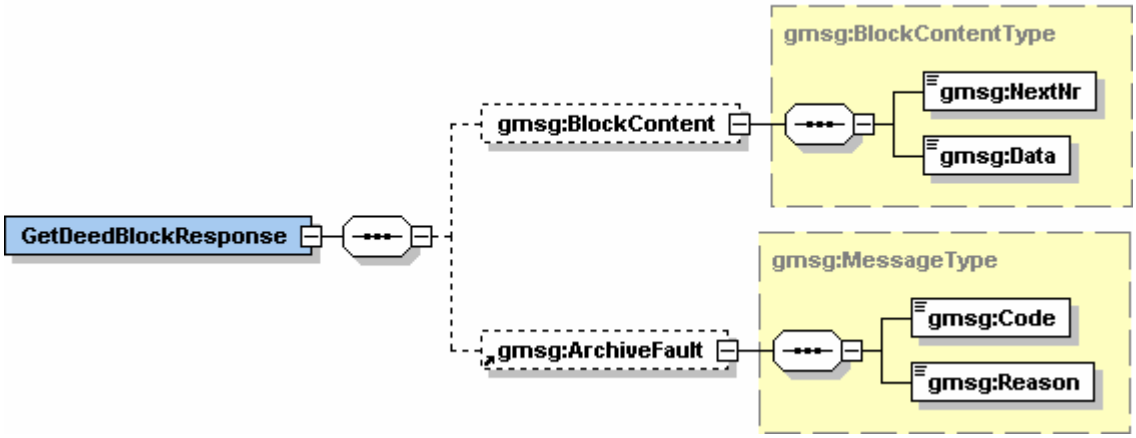

Abbildung 16

Regeln für den Aufbau des gmsg:GetDeedBlockResponse (**R3002**)

- Wenn der Request erfolgreich verarbeitet werden konnte, muss ein Element `gsig:BlockContent` zurückgegeben werden.
- Wenn ein fachlicher Fehler aufgetreten ist, muss ein Element `gmsg:ArchiveFault` zurückgegeben werden.

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:BlockContent	Element		Enthält einen ca. 64kByte großen Block der Urkunde
gmsg:ArchiveFault	Element		Fachliche Fehlermeldung

gmsg:BlockContent

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Handle	string	✓	Referenz auf die Urkunde auf Serverseite
gmsg:Data	base64	✓	Beinhaltet einen maximal 64kByte großen Block der Urkunde

8.3.7 gmsg:ArchiveFault

Anmerkung: Das Element gmsg:ArchiveFault wird sowohl für Webservice RemoteExceptions als auch fachliche Fehler verwendet.

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Code	string	✓	Der String muss einen der vordefinierten Fehlercodes enthalten. Es dürfen nur Fehlercodes der Klassen F100, S100 und S200 verwendet werden (R3003)
gmsg:Reason	string	✓	Nähere Details zum aufgetretenen Fehler

8.3.8 gmsg:SendErrorNotificationRequest

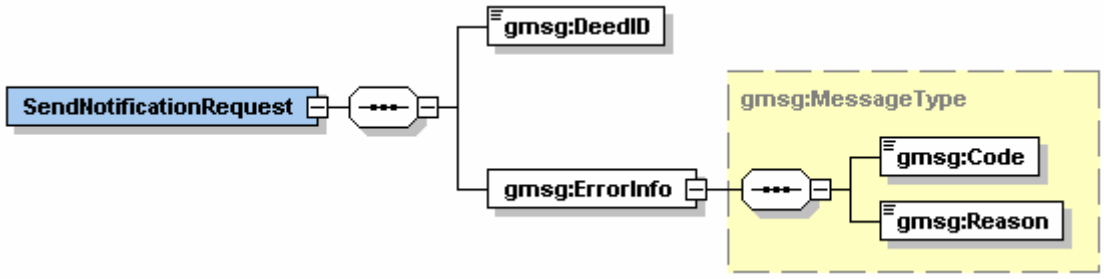

Abbildung 17

Dient der Rückmeldung eines bei der Nachverarbeitung aufgetretenen fachlichen Fehlers.

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:DeedID	string	✓	Urkundenidentbegriff, damit der Archivbetreiber den bei der Nachverarbeitung aufgetreten Fehler einer Urkunde zuordnen zu können.
gmsg:ErrorInfo	Element	✓	Fachliche Fehlermeldung

gmsg:ErrorInfo

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Code	string	✓	Der String muss einen der vordefinierten Fehlercodes enthalten. Es dürfen nur Fehlercodes der Klassen F100, S100 und S200 verwendet werden (R3004)
gmsg:Reason	string	✓	Nähere Details zum aufgetretenen fachlichen Fehler

8.3.9 gmsg:SendErrorNotificationResponse

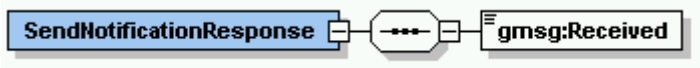

Abbildung 18

Name	Type	Pflicht	Bedeutung
Elemente			
gmsg:Received	boolean	✓	Reines Hilfskonstrukt, damit jedenfalls eine auswertbare Antwort an den Client zurück gemeldet wird. Kann in der Regel immer `true` sein.

8.3.10 gmsg:CloseDeedRequest

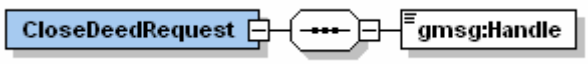

Abbildung 19

Dient der Rückmeldung eines bei der Nachverarbeitung aufgetretenen fachlichen Fehlers.

Name	Type	Pflicht	Bedeutung
Elemente			

gmsg:Handle	string	✓	gmsg:Handle ist eine serverseitige Referenz auf die Urkunde, welche für den Download gepuffert wird.

8.3.11 gmsg:CloseDeedResponse

Abbildung 20:

Der Response ist leer und enthält keine Parameter. Die Konstruktion dient allein dazu den Webservice Generator zufrieden zu stellen.

8.4 GOGArchiveSignature.xsd

8.4.1 gsig:ArchiveSignature

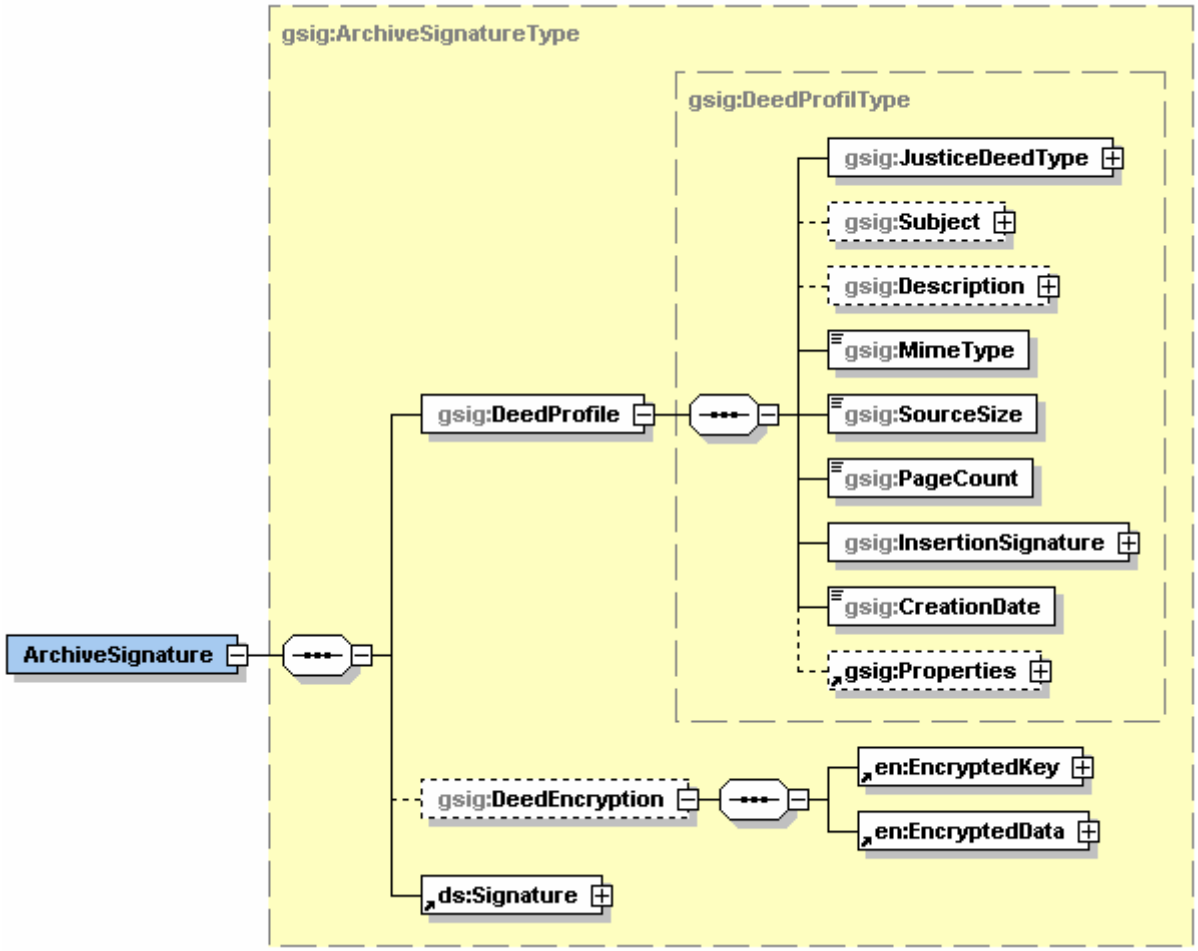

Abbildung 21: gsig:ArchiveSignature

Name	Type	Pflicht	Bedeutung
Elemente			
gsig:DeedProfile	Element	✓	Enthält Metadaten zur Urkunde

Name	Type	Pflicht	Bedeutung
gsig:DeedEncryption			Kann verschlüsselten Sessionkey und Referenz auf die verschlüsselte Urkunde enthalten. Urkunde und Metadaten müssen nicht zwingend verschlüsselt werden (R0003).
ds:Signatur	Element	✓	Enthält Archivsignatur im XML DSig Format

gsig:DeedEncryption

Name	Type	Pflicht	Bedeutung
Elemente			
en:EncryptedKey	Element	✓	Verschlüsselten Sessionkey wird sowohl für Urkunde als auch für folgende Metadaten verwendet: • gsig:Subject • gsig:Description • gsig:Properties
en:EncryptedData	Element	✓	Enthält Referenz auf Urkunde Die Referenz ist eine URI die auf ein File im lokalen Filesystem verweist

8.4.2 gsig:DeedProfile (Metadaten)

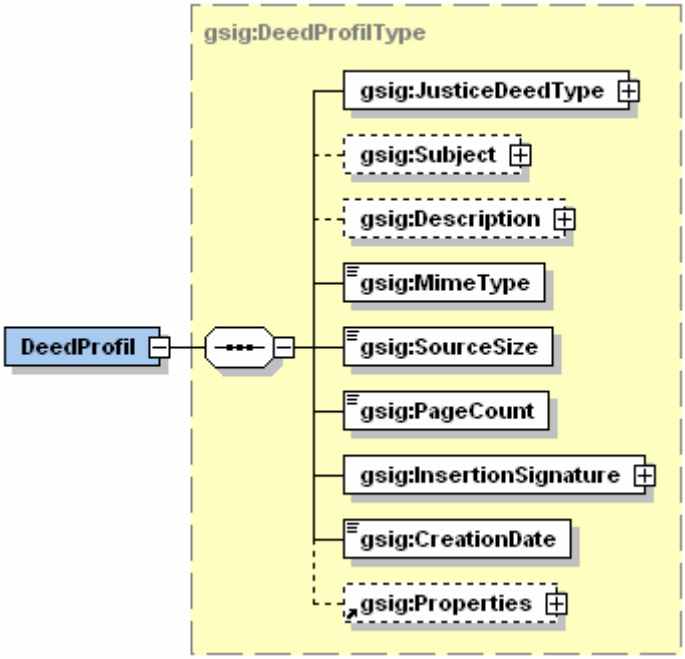

Abbildung 22: Überblick Metadaten

Name	Type	Pflicht	Bedeutung
Elemente			
gsig:JusticeDeedType	gsig:ValueType	✓	Gegenstand der Urkunde (Justiz) - Klassifizierung der Urkunde entsprechend den von der Justiz vorgegeben Typen (R2002) • Der Datentyp für den ValueType ist "String"
gsig:Subject	gsig:ValueType		Gegenstand der Urkunde (Berufsgruppe) - Dieses Feld muss ausgefüllt sein, wenn `gsig:JusticeDeedType` gleich "Sonstige Urkunde" ist. (R2003) • Der Datentyp für den ValueType ist "String"
gsig:Description	gsig:ValueType		Beschreibung der Urkunde • Der Datentyp für die Description ist "String"
gsig:MimeType	string	✓	Dokumententyp: Es werden nur PDF- und TIFF-Dokumente akzeptiert (R2004)
gsig:SourceSize	long	✓	Größe der Urkunde in Byte
gsig:PageCount	long	✓	Anzahl der Seiten. Zur Ermittlung der Seitenanzahl muss die Regel "Berechnung der Seitenanzahl" (R2005) angewandt werden.
gsig:InsertionSignature	Element	✓	Beinhaltet den Zeitpunkt und Zertifikat der Beurkundungssignatur
gsig:CreationDate	dateTime	✓	Erstellungsdatum der Urkunde
gsig:Propierties	gsig:PropteriesType		Property-Value-Struktur, um Verfahrensspezifische Metadaten definierten und transportieren zu können.
en:EncryptedKey	Element		Transportschlüssel, um Metadaten und/oder Urkunde entschlüsseln zu können. Kann für folgende Metadaten benötigt werden: • gsig:JusticeDeedType • gsig:Subject • gsig:Description • gsig:Properties

8.4.3 gsig:InsertionSignature

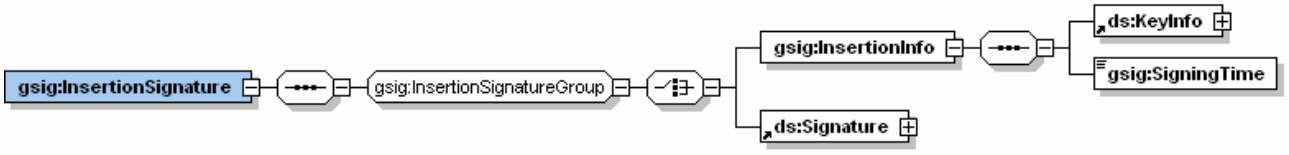

Abbildung 23

Name		Type	Bedeutung
Elemente			
gsig:InsertionSignatureGroup		Group[1]	gsig:InsertionSignatureGroup ist ein Platzhalter, der entweder durch den Zeitstempel plus Zertifikat der Einstellung oder einer Einstellungssignatur ersetzt wird (R2006):
entweder oder	gsig:InsertionInfo	Element	Enthält im Element `ds:KeyInfo` das Zertifikat des Einstellers und im Element `gsig:SigningTime` den Zeitstempel der Einstellung in das Archiv.
	ds:Signature	ds:SignatureType	Einstellungssignatur vom Type XMLDSig (R0001)

8.4.4 gsig:ValueType

Der ValueType kommt generell bei Informationen zum Einsatz, die der Archivbetreiber nicht im Klartext übermitteln darf.

Beispielsweise dürfen Notare und Rechtsanwälte den Gegenstand der Urkunde (Berufsgruppe) nur verschlüsselt übermitteln.

Im Gegensatz dazu ist eine Verschlüsselung des Gegenstands der Urkunde bei Ziviltechnikern nicht notwendig, da es sich hier um öffentliche Urkunden handelt.

Aus diesem Grund bietet die Struktur die Möglichkeit die Information auch im Klartext zu transportieren.

Grundsätzlich gilt: Wenn der `gsig:ValueType` verwendet wird, muss in der Spezifikation festgehalten werden, welchen Datentyp der Wert hat:

- String (xs:string)
- Binary (xs:base64)
- Date (xs:dateTime)
- Decimal (xs:decimal)
- Integer (xs:long)

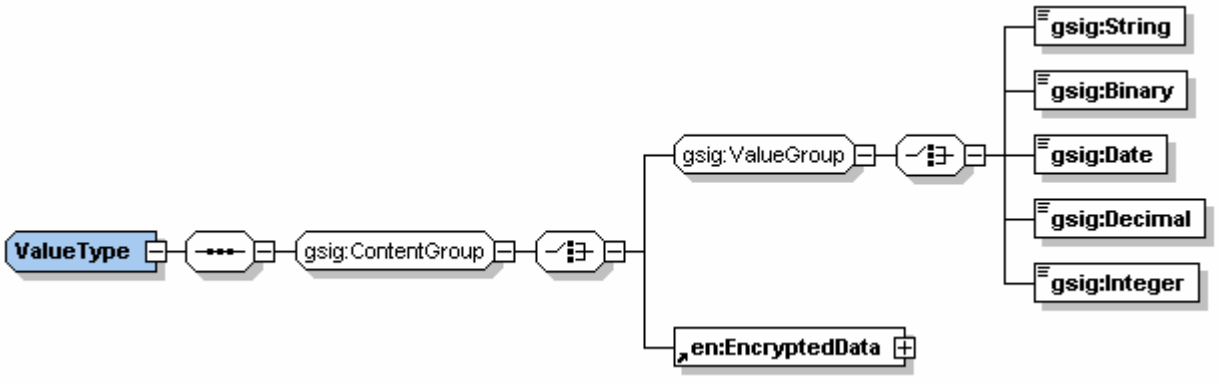

Abbildung 24: gsig:ValueType

[1] XML-Group ist ein Konstrukt das sehr gerne von XML-Zugriffsklassen Generatoren verwendet. Beispielsweise dient das XML-Group Konstrukt bei JAXB dazu ein XML-Choice sauber aufzulösen. Anmerkung: XML-Group ist ein reiner Platzhalter und scheint niemals ein eigenständiges Tag in einer XML-Struktur auf.

8.4.5 gsig:Properties (Properties-Value-Struktur)

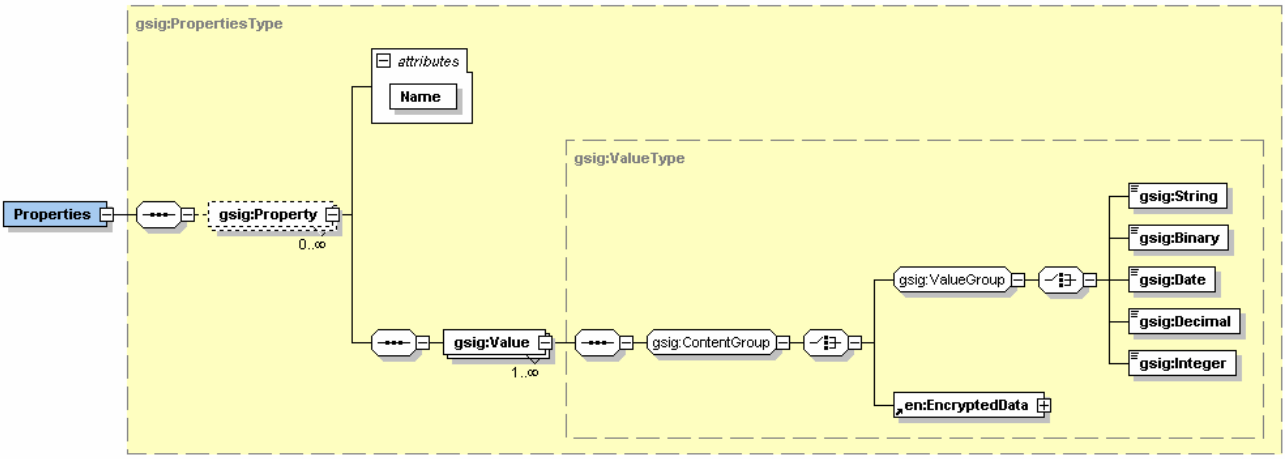

Abbildung 25: Struktur zur Abbildung der Liste der Property-Values

gsig:Properties

Name	Type	Anzahl	Bedeutung
Elemente			
gsig:Property	gisg:PropertyType	0..n	Liste von Eigenschaften

gsig:Property

Name	Type	Pflicht	Anzahl	Bedeutung
Attribute				
Name	string	✓		Name der Eigenschaft
Elemente				
gsig:Value	gsig:ValueType	✓	1..n	Zu einer Eigenschaft kann eine Liste von Werten zugeordnet werden.

gsig:Value

Name			Type	Bedeutung
Elemente				
gsig:ContentGroup			Group[2]	gsig:ContentGroup ist ein Platzhalter, der entweder durch verschlüsselte Daten oder dem tatsächlichen Wert ersetzt wird:
entweder oder	en:EncrptedData		en:EncryptedDataType	Wenn gsig:Value ein Element "en:EncrptedData" beinhaltet, dann ist darin der tatsächliche Wert verschlüsselt enthalten. Anderenfalls wird der Wert als String, Binary, Date, Decimal oder Integer übergeben.
	gsig:ValueGroup		Group[2]	gsig:ValueGroup ist ein reiner Platzhalter, der durch eines der folgenden Elemente ersetzt wird:
	entweder oder	gsig:String	string	
		gsig:Binary	base64	
		gsig:Date	dateTime	
		gsig:Decimal	decimal	
		gsig:Integer	long	

[2] XML-Group ist ein Konstrukt das sehr gerne von XML-Zugriffsklassen Generatoren verwendet. Beispielsweise dient das XML-Group Konstrukt bei JAXB dazu ein XML-Choice sauber aufzulösen. Anmerkung: XML-Group ist ein reiner Platzhalter und scheint niemals ein eigenständiges Tag in einer XML-Struktur auf.

9 Regeln

9.1 Allgemein (R0000)

Regel	Beschreibung
R0001	**Erstellung und Prüfung von Signaturen im XMLDSig Format** Basis für die Erstellung und Prüfung von Signaturen im XMLDSig Format sind Rahmenbedingungen wie in [1] "Minimale Umsetzung des Security-Layers" für die Bürgerkarten Umgebung definiert. (*http://www.buergerkarte.at/konzept/securitylayer/spezifikation/aktuell/minimum/Minimum.html*) Ferner gilt: 1) Die Archivsignatur muss im Element `dsig:SignedInfo` zumindest ein Element `dsig:Reference` enthalten. Dieses Element muss in seinem Attribut `URI` einen Wert enthalten, der folgendem Aufbau genügt: `file:<Dateiname>.<Extension>` (also z.B. file:Urkunde.pdf). D.h. es sind weder relative noch absolute Pfadangaben zulässig 2) Die Archivsignatur darf im Element `dsig:SignedInfo` neben der Referenz auf die Urkunde weitere `dsig:Reference` Elemente enthalten, um z.B. Signaturattribute innerhalb des die XMLDSIG-Signatur repräsentierenden XML-Dokuments zu referenzieren. In einem solchen Fall muss die unter (2) erläuterte Referenz jedoch das erste `dsig:Reference` Element in `dsig:SignedInfo` sein. 3) Die Archivsignatur muss im Element `dsig:KeyInfo` genau ein Element `dsig:X509Data` enthalten. Dieses Element muss zumindest ein Element `dsig:X509Certificate` enthalten, das als Textinhalt das Signaturzertifikat des Archivs enthält. `dsig:X509Data` darf darüber hinaus weitere `dsig:X509Certificate` Elemente enthalten, um z.B. weitere Zertifikate für die Zertifikatskettenbildung zu transportieren. In einem solchen Fall muss das `dsig:X509Certificate` mit dem Signaturzertifikat jedoch das erste innerhalb von `dsig:X509Data` sein.
R0002	**Eindeutiger String für Principal und Issuer** • Es dürfen alle Kurzbezeichnungen, die im RFC 1779 (A String Representation of Distinguished Names) explizit beschrieben werden verwendet werden z.B.: CN, O, OU, C • Alle Kurzbezeichnungen, die im RFC 1779 nicht explizit beschrieben sind, werden durch OIDs (z.B.: OID.2.5.4.3=Wert) ersetzt. • Der Distinguished Name muss entsprechend des RFC 1779 geparst werden.
R0003	**Urkunde ist nicht zwingend zu verschlüsseln** Urkunden müssen für den Transport nicht zwingend verschlüsselt werden. • Die Verschlüsselung der Urkunden ist für Notare und Rechtsanwälte verpflichtend, da diese Urkunden vertraulich sind. • Ziviltechniker versenden öffentliche Urkunden, d.h. es ist keine Verschlüsselung erforderlich.
R0004	**Identbegriff** • Jede Urkunde muss über eine eindeutige ID angesprochen werden können. • Die ID ist ein String. • Aus welchen Informationen die ID gebildet wird, ist dem Archivbetreiber überlassen.
R0005	**Gültigkeit von TIFFs** • Urkunden im TIFF-Format dürfen nur noch bis Ende 2009 erstellt werden.

Regel	Beschreibung
R0006	**Versionierung von Urkunden** • Wenn sich eine Urkunde ändert, muss zwingend ein neuer Identbegriff vergeben werden. • Diese Regelung gilt auch für jede einzelne Urkunde oder jeden einzelnen Anhang eines Container-PDFs.
R0007	**Anforderung einer alten Urkundenversion** • Wenn von der Justiz oder generell über die Schnittstelle eine alte Urkundenversion angefordert wird, muss diese Anforderung mit einem Fehler quittiert werden.
R0010	**Archivsignatur** • Diese Archivsignatur ist unabhängig von der Urkunde im SOAP-Body als XMLDSig vom Typ XMLDSig mit den in R0001 definierten Einschränkungen zu transportieren. • XPath: `/gsig:ArchiveSignatur/ds:Signature`
R0011	**Ermittlung des Sessionkeys** 1) Der SessonKey wird auf dem XPath gmsg:GetDeedResponse/gsig:ArchiveSignature/gsig:DeedEncryption/en:EncryptedKey erwartet. 2) Der Verschlüsselungsalgorithums muss auf dem XPath gmsg:GetDeedResponse/gsig:ArchiveSignature/gsig:DeedEncryption /en:EncryptedKey/en:EncryptionMethod/@Algorithm angegeben sein.
R0012	**Verschlüsseln von Urkunden und Metadaten** Wenn im Request ein Element "gsig:DeedEncryption" definiert ist, müssen sowohl die Urkunde als auch die Metadatenfelder • gsig:JusticeDeedType • gsig:Subject • gsig:Description zwingend verschlüsselt werden.
R0013	**Es wird nur die XADES Versionen http://uri.etsi.org/01903/v1.1.1# unterstützt.**
R0014	**Die Signatur muss zwingend eine SigningTime enthalten** Die XMLDSig Signatur auf dem XPath /gmsg:GetDeedResponse/gsig:ArchiveSignature/ds:Signature/ muss QualifiedProperties aus dem Namespace http://uri.etsi.org/01903/v1.1.1# (Prefix etsi) enthalten. Die Qualified Properties wiederum müssen auf dem XPath /gmsg:GetDeedResponse/gsig:ArchiveSignature/ds:Signature/ /ds:Object/etsi:QualifyingProperties/etsi:SignedProperties/etsi:SignedSignatureProperties /etsi:SigningTime die SigningTime, d.h. den Zeitpunkt der Archivsignaturerstellung enthalten.

9.2 PDF-Regeln (R1000)

Regel	Beschreibung
R1001	**Container PDFs werden nicht unterstützt** • Auf Grund organisatorischer und rechtlicher Gründe können Container PDF Dokumente nicht unterstützt werden. • Definition: Unter einem "Container PDF" wird eine PDF-Datei verstanden, die lediglich als Behälter für alle zu einem Geschäftsfall gehörenden Urkunden und Anhänge dient. Oder anders ausgedrückt: das Container PDF stellt den Aktendeckel für eine Menge von zusammengehörenden Urkunden und Anhängen dar.
R1002	**Jedes einzelne Dokument eines Container PDFs muss eindeutig adressiert werden können.** • Da Container PDFs nicht unterstützt werden können (vgl. R1001), muss jedes einzelne Dokument des Containers über eine eindeutige ID adressiert werden können. • Damit wird sichergestellt, dass jedes einzelne Dokument des Containers über die gegenständlich spezifizierte Schnittstelle abgerufen werden kann. • siehe auch Identbegriff (R0004) • siehe auch Versionierung von Urkunden (R0006)
~~R1003~~	~~**Signaturen müssen in PDF-Dokument eingebettet werden**~~ ~~• Wenn Urkunden im PDF-Format übertragen werden, muss die Beurkundungs- und Archivsignatur ins PDF-Dokument eingebettet werden.~~ ~~• siehe auch Übergangsregelung (R9001)~~
~~R1004~~	~~**Signaturtyp für eingebettete Signaturen**~~ ~~• Die in das PDF-Dokument eingebetteten Signaturen müssen vom Typ "PDF-Amtssignatur" sein, damit diese Signaturen mit Hilfe des MOA Modules PDF-AS geprüft werden können.~~ ~~• Es gilt die *PDF-AS-Spezifikation-2.0.0*~~
~~R1005~~	~~**Es ist die binäre PDF-Signatur zu verwenden**~~ ~~• Es ist die binäre Variante der PDF-Signatur laut *PDF-AS-Spezifikation-2.0.0* (vgl. R1004) zu verwenden.~~
R1006	**Das PDF-Dokument muss PDF/A-1b entsprechen** 1) Alle GOG Archive verpflichten sich PDF-Dokumente, die in das Archiv eingestellt werden sollen, gegen PDF/A-1b zu validieren. Hierbei obliegt es dem Archivbetreiber, ob ein Validator oder ein PDF zu PDF/A-1b Konverter eingesetzt wird. 2) Für die Validierung bzw. Konvertierung wird kein spezielles Tool vorgeschrieben, jedoch muss die vom Archivbetreiber eingesetzte Software den Isartor PDF/A-1b Test erfüllen. 3) Im Gegenzug vertraut die Justiz darauf, dass die in den Archiven erfolgreich eingestellten Dokumente PDF/A-1b entsprechen. D.h. beim Abholen der Dokumente durch die Justiz werden die Dokumente nicht nochmals validiert.
R1007	**Es gilt der Isartor PDF/A-1b Test in der Version 1.0 vom 13.8.2008**

9.3 Metadaten Regeln (R2000)

Regel	Beschreibung
R2001	**Unterscheidungsmerkmal "Gegenstand der Urkunde"** Sowohl auf Seite der Berufsgruppen als auch auf Seite der Justiz wird der Begriff "Gegenstand der Urkunde" verwendet. Je nach Partei hat dieser Begriff jedoch eine andere Bedeutung. Um diesen Unterschied hervorzuheben wird zwischen "Gegenstand der Urkunde (Berufsgruppe)" und "Gegenstand der Urkunde (Justiz)" unterscheiden: • "Gegenstand der Urkunde (Justiz)" ist die Klassifizierung der Urkunde, wie sie die Justiz erwartet. Hierfür steht eine Liste fix vorgegebener Urkundentypen zur Verfügung (R2002) XPath: `/gsig:ArchiveSignatur/gsig:DeedProfile/gsig:JusticeDeedType` • "Gegenstand der Urkunde (Berufsgruppe)" ist die Klassifizierung der Urkunde entsprechend den Usancen der einzelnen Berufsgruppen. XPath: `/gsig:ArchiveSignatur/gsig:DeedProfile/gsig:Subject`
R2002	**Gegenstand der Urkunde (Justiz)** Der Gegenstand der Urkunde (Justiz) am XPath: `/gsig:ArchiveSignatur/gsig:DeedProfile/gsig:JusticeDeedType` muss einen der folgenden Urkundentypen enthalten: Abtrennungsbewilligung Abtretungsvertrag Amtsbestätigung Amtsurkunde Anmeldungsbogen Anmeldungsbogen gem § 12 VermG Anmeldungsbogen gem § 13 LTG Anmeldungsbogen gem § 15 LTG Aufhebungsvertrag Aufsandungserklärung Ausstattungsvertrag Auszug aus Sterbebuch Baulandbestätigung Baurechtsvertrag Bescheid Bescheinigung Beschluss Bestandsvertrag Bestätigung Beurkundung Darlehens- und Pfandbestellungsurkunde Dienstbarkeitsvertrag Diplom Dissolutionsvertrag Ehepakt Einantwortungsbeschluss Einantwortungsurkunde Einbringungsvertrag Erbteilungsübereinkommen Erbverzichtsvertrag Erklärung gerichtl Vergleich Gesellschaftsvertrag Gutachten Heiratsurkunde Kaufvertrag Kautionsbestellungsurkunde Konkursedikt Leibrentenvertrag Löschungserklärung Löschungsquittung Löschungsurkunde Meistbotsverteilungsbeschluss Meldezettel Nachtrag Nachtrag zum Kaufvertrag Niederschrift Nutzwertgutachten Pachtvertrag Pfandauflassungserklärung Pfandausdehnungsurkunde Pfandbestellungsurkunde Pfandrechtsvormerkung Pfandurkunde Pflichtteilsübereinkommen Plan Realteilungsvertrag Reisepass Rückstandsausweis Schenkungsvertrag Schuld- und Pfandbestellungsurkunde Schuldschein Selbstberechnung Spezialvollmacht Staatsbürgerschaftsnachweis Sterbeurkunde Strassengrundabtretungserklärung Tauschvertrag Teillöschungserklärung Teilungsplan Treuhandvertrag Übereinkommen Übergabsvertrag Unbedenklichkeitsbescheinigung Urteil Vereinbarung Vergleich Verhandlungsschrift Vermessungsurkunde Verordnung Vertrag Verweisungsblatt Vollmacht Vorrangeinräumungserklärung Wohnungseigentumsvertrag Zahlungsauftrag Zahlungsbefehl Zeichnungsbestätigung Zusage gem § 40 (2) WEG 2002 Zusicherung Zustimmungserklärung

Regel	Beschreibung
	Erteilung des Zuschlages Firmenbuchauszug Flurbereinigungsübereinkommen Freilassungserklärung Servitutsvertrag Siedlungs- bzw. Flurbereinigungsverfahren (Einleitung) Sonstige Urkunde
R2003	**Sonstige Urkunden** • Wenn im Element `gsig:JusticeDeedType` der Wert "sonstige Urkunde" eingetragen ist, muss die Urkunde im Element `gsig:Subject` näher spezifiziert werden.
R2004	**Es werden nur TIFF und PDF Dokumente unterstützt** • Das Element `gsig:MIMEType` darf nur die beiden Werte `application/pdf` oder `image/tiff` enthalten.
R2005	**Berechnung der Seitenanzahl** • Die Seitenanzahl der Seiten eines Dokumentes ist unabhängig vom Format der einzelnen Seiten zu bestimmen.
R2006	**Einstellungsinformation** Das Pflichtelement `gsig:InsertionSignature` enthält entweder • den Zeitstempel der Einstellung der Urkunde ins Archiv plus dem Zertifikat des Einstellers • oder eine Einstellungssignatur vom Typ XMLDSig mit den in R0001 definierten Einschränkungen.
R2007	**Verschlüsseln von Properties** Beim Verschlüsseln der Values einer Property-Value Struktur darf der Namespace weggelassen werden: Beispiel: Statt der Langform <?xml version="1.0" encoding="UTF-8"?> <gsig:String xmlns:gsig="http://brz.gv.at/GOGArchive/Signature/v1.0#"> Das ist ein Test </gsig:String> ist auch folgend Kurzform zulässig <String>Das ist ein Test</String>
R2008	**Verschlüsseln von Metadaten** Bei der Verschlüsselung der Metadatenfeldern • gsig:JusticeDeedType • gsig:Subject • gsig:Description gilt die Regel R2007. Zusätzlich darf die Angabe des Typs entfallen. Beispielsweise kann für den Gegenstand der Urkunde (Justiz) (gsig:JusticDeedType) statt <String> Gesellschaftsvertrag </String> auch nur Gesellschaftsvertrag angeben werden.

9.4 Technische Regeln (R3000)

Die Klasse der technischen Regeln betrifft Konventionen, die bei der Erzeugung der XML-Strukturen eingehalten werden müssen. Konkret wurden dann technische Regeln definiert, wenn die XSD nicht präzise genug definiert werden können oder bestimmte Wertebereiche eingehalten werden müssen.

Regel	Beschreibung
R3001	**GetDeedResponse** • Wenn der Request erfolgreich verarbeitet werden konnte, muss sowohl ein Element `gmsg:Handle` als auch ein Element `gsig:ArchiveSignature` zurückgegeben werden. • Wenn ein fachlicher Fehler zurückgemeldet wird, darf nur ein Element `gmsg:ArchiveFault` zurückgegeben werden.
R3002	**GetDeedBlockResponse** • Wenn der Request erfolgreich verarbeitet werden konnte, muss ein Element `gsig:BlockContent` zurückgegeben werden. • Wenn ein fachlicher Fehler aufgetreten ist, muss ein Element `gmsg:ArchiveFault` zurückgegeben werden.
R3003	**Serverseitige Fehlerbehandlung** Vom Server dürfen sowohl fachliche Fehlercodes der Klasse F100 als auch technische Fehlercodes der Klassen S100 und S200 an den Client zurückgemeldet werden.
R3004	**Clientseitige fachliche Fehler (SendErrorNotification)** Vom Client dürfen nur fachliche Fehlercodes der Klasse F200 an den Server zurückgemeldet werden.

10 Fehlerbehandlung

10.1 Fachliche Fehler

Fehlerklasse F100 - Fehler auf der Serverseite (=Archivbetreiber)

Fehler code	Meldung
F100	**Interner Archivfehler** Wird für alle andern nicht klassifizierten Fehler auf Serverseite verwendet. Details können im Meldungstext angeführt werden.
F101	**Urkundenkennung ist ungültig** Syntaxfehler in der Urkundenkennung
F102	**Urkunde konnte nicht gefunden werden** Es gibt zur angegebenen Kennung keine Urkunde
F103	**Veraltete Version der Urkunde wurde angefordert** Zu der angeforderten Urkunde existiert bereits eine neuere Version

Fehlerklasse F200 - Fehler auf der Clientseite (=Nutzer)
Diese Fehlerklasse beschreibt Ausnahmen, die beim Verarbeiten des Response auf Clientseite auftreten, d.h. es handelt sich um Fehler, die sich direkt auf die Urkunde beziehen.

Über diese Fehler wird der Archivbetreiber mit Hilfe der Webserviceoperation "`SendErrorNotification`" informiert. Es liegt jedoch im Ermessen des Archivbetreibers diese Informationen auszuwerten.

Fehler code	Beschreibung
F200	**Response entspricht nicht der Spezifikation** Kommt zur Anwendung, wenn eine der Regeln der Spezifikation nicht erfüllt ist. Die Regelnummer plus Details werden im Meldungstext ausgegeben.
F201	**XML-Datenstruktur entspricht nicht dem Schema** Verletzung des Schnittstellenvertrags
F202	**Urkunde konnte nicht entschlüsselt werden** Entweder sind nicht alle Informationen verfügbar um die Urkunde entschlüsseln zu können, oder die entschlüsselte Datei enthält keinen gültigen TIFF- oder PDF-Header
F203	**Archivsignatur konnte nicht geprüft werden** Fehler wird ausgegeben, wenn nicht alle Informationen aus der XML-Struktur gelesen werden konnten, die für die Signaturprüfung erforderlich sind
F204	**Archivsignatur ist ungültig** Fehler wird ausgegeben, wenn die Signatur korrupt ist.
F205	**Archivzertifikat ist ungültig** Details, warum die Prüfung fehlgeschlagen ist, werden im Meldungstext angeführt, z.B.: • Archivzertifikat ist kein GOG-Archivzertifikat • Zertifikat ist abgelaufen etc…

Fehler code	Beschreibung
F206	**PDF ist nicht PDF/1A-b konform**
F207	**Urkunde ist zu groß** Die Größe der Urkunde überschreitet das Limit von 60MB
F208	**Urkunde konnte nicht vom TIFF ins PDF Format konvertiert werden**

10.2 Technische Fehler

Beschreibt technische Fehler im Bereich Webservice oder Backend Systeme auf Archivseite

Fehlerklasse S100 - Allgemeine technische Fehler

Fehler code	Beschreibung
S100	**Unbekannte Exception ist aufgetreten** Wird für selten auftretende Fehlersituationen verwendet. Exception plus Fehlermeldung kann im Meldungstext angeführt werden.

Fehlerklasse S200 - Fehler beim Blocktransfer der Urkunden

Fehler code	Beschreibung
S201	**Urkundenrequest existiert nicht mehr** Der Fehler tritt auf, wenn der Client zu viel Zeit zwischen den einzelnen `getDeedBlock` Aufrufen zu einer Urkunde vergehen lässt. Hintergrund: Der Server löscht nach einem bestimmten Timeout den Request aus der Queue.
S202	**Urkundenrequest ist schon gesperrt** Der Fehler tritt auf, wenn der Client einen Block anfordert, obwohl der Server zu dieser Urkunde bereits eine Anfrage abarbeitet.
S203	**Urkundenrequest wurde abgewiesen- maximaler Queueschwellwert wurde überschritten** Der Fehler tritt auf, wenn der Schwellwert für die maximale Queueauslastung überschritten wurde. Gegebenenfalls muss der Client - nach Verstreichen einer angemessenen Wartezeit - nochmals versuchen die Urkunde abzuholen.
S204	**Handle im Urkunden-Request ist nicht gesetzt.** Der Client hat beim Aufruf von GetDeedblock kein Handle-Element gesetzt

11 Begriffsdefinitionen

Amtssignatur: Technik wie eine Signatur entsprechend *EGIZ: PDF-Amtssignatur – Spezifikation, Version 2.0.0, E-Government Innovationszentrum (EGIZ), vom 4.10.2006. (nach E-Government Gesetzt - EGovG)* in ein PDF eingebettet wird.

Archivsignatur: Bezeichnet die elektronische Unterschrift des Archivs

Beurkundungssignatur: Bezeichnet die elektronische Unterschrift des Notars, des Rechtsanwalts oder des Ziviltechnikers im Sinne des elektronischen Originals (entspricht der Signatarsignatur).

Einstellungssignatur ist jene Signatur, die aufgebracht wird, wenn die Urkunde vom Notar oder Rechtsanwalt in das Archiv eingestellt wird. Diese Signatur ist ungleich der Beurkundungssignatur (Stichwort: elektronisches Original).

Identbegriff: Eindeutige ID zur Identifizierung einer Urkunde

Metadaten: Sind zusätzliche Informationen zur Urkunde.

Property-Value-Struktur: Beliebige Daten werden in der Form <Feldname> = <Wert> transportiert. Vereinfacht dargestelltes Beispiel: "Geburtsdatum" = "1957.03.16" .

Rückführung: PDF-Amtssignatur wird in XMLDSig übergeführt

Signatar: Ist der Unterzeichner einer Urkunde konkret ein Notar, ein Rechtsanwalt oder ein Ziviltechniker.

Signatarsignatur: Bezeichnet die elektronische Unterschrift des Notars, des Rechtsanwalts oder des Ziviltechnikers im Sinne des elektronischen Originals (entspricht der Beurkundungssignatur).

XMLDSig: Spezifikation der "XML digital signature processing rules and syntax" des W3C Konsortiums

12 Referenzen

[1] Minimale Umsetzung des Security-Layers 1.2.1
Für die Erstellung und Prüfung von Signaturen im XMLDSig Format gelten die gleichen Rahmenbedingungen wie in "Minimale Umsetzung des Security-Layers" für die Bürgerkarten Umgebung definiert, Februar 2008

http://www.buergerkarte.at/konzept/securitylayer/spezifikation/aktuell/minimum/Minimum.html

[2] PDF/A-1b Electronic document file format for long-term preservation
ISO-19005-1 - Document management – Electronic document file format for long-term preservation – Part 1: Use of PDF 1.4 (PDF/A-1).

[3] PDF/A Competence Center
Das PDF/A Competence Center (www.pdfa.org) ist ein Zusammenschluss weltweit führender Unternehmen und Fachleute im Bereich PDF-Technologie. Der Zweck des PDF/A Competence Center ist die Förderung des Informations- und Erfahrungsaustausches auf dem Gebiet Langzeitarchivierung gemäß ISO 19005: PDF/A.

[4] PDF Amtssignaturspezifikation 2.0.0
Die Spezifikation legt fest, wie PDF-Dokumente mit einer elektronischen Signatur zu versehen sind, Jänner 2008.

https://demo.egiz.gv.at/plain/content/download/527/3056/file/PDF-AS-Spezifikation-2.0.0.pdf

(siehe auch https://demo.egiz.gv.at/plain/projekte/signatur_im_e_government/pdf_signatur)

[5] XML Advanced Electronic Signatures (XAdES)
Erweiterung zum XMLDSig Standard in der Version ETSI TS 101 903 V1.1.1

http://uri.etsi.org/01903/v1.1.1/

[6] XMLDSig
Eastlake, Donald, Reagle, Joseph und Solo, David: XML-Signature Syntax and Processing. W3C Recommendation, Februar 2002.

http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/