



Informations
Technik
Zentrum Bund

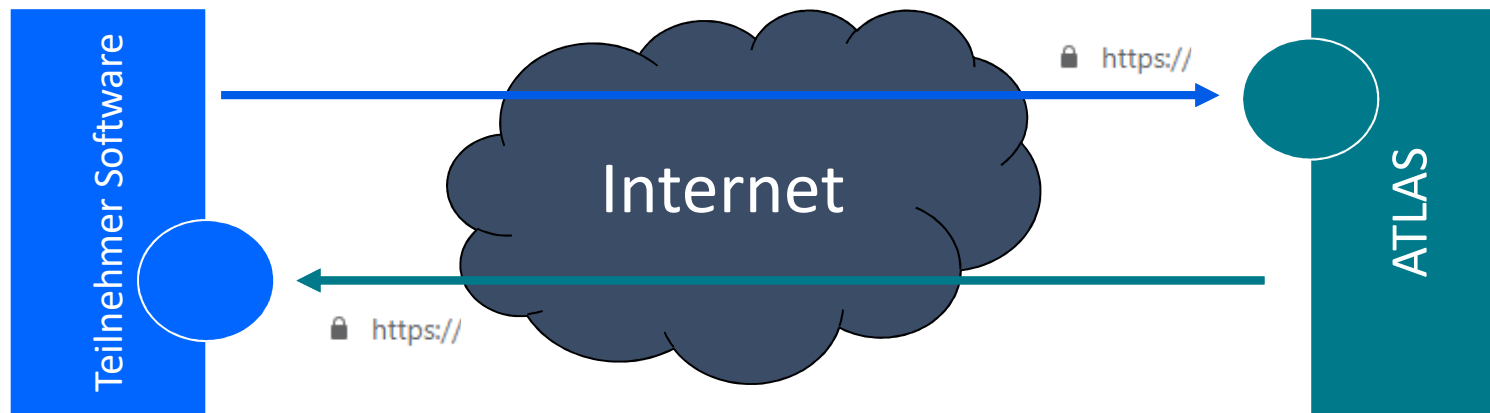
Änderungen in der externen Kommunikation

Webservice Kommunikation im Kontext von ATLAS-IMPOST

Vortragende: Andreas Oertel (ITZBund), Carsten Sensler (ext.)

Stand: 25.3.2021

Webservice-Kommunikation über das Internet.

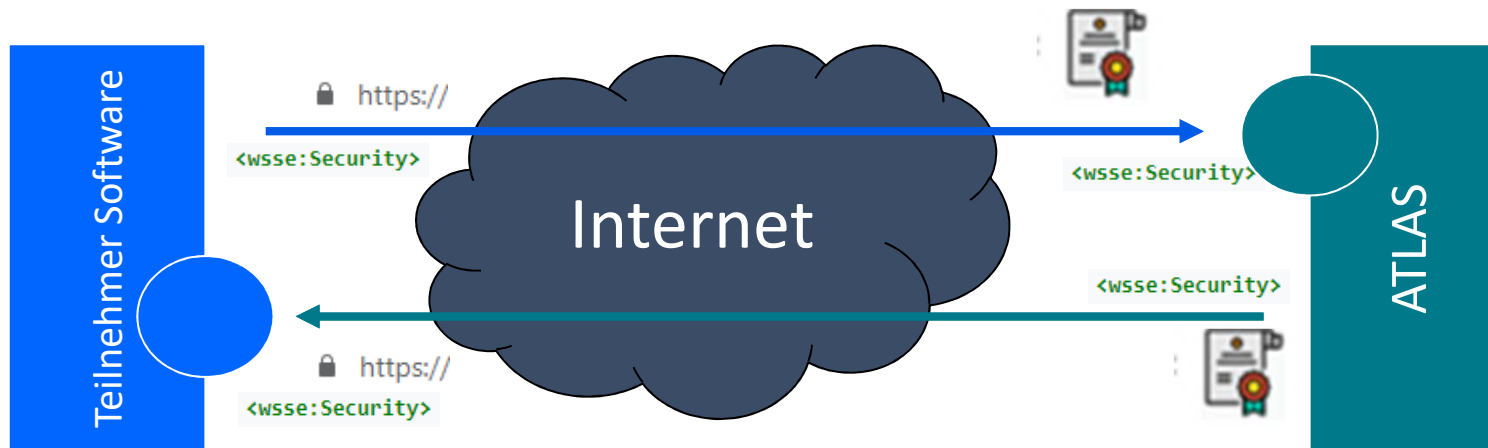


- Die Definition der Webservices erfolgt für beide Seiten durch ATLAS.
 - ATLAS wird Webservices für die Teilnehmer-Software bereitstellen.
 - Die Teilnehmer-Software muss Webservices für ATLAS bereitstellen.
- Die Webservices werden nur einzelne Datensätze verarbeiten.

Die Webservice Kommunikation basiert auf SOAP.

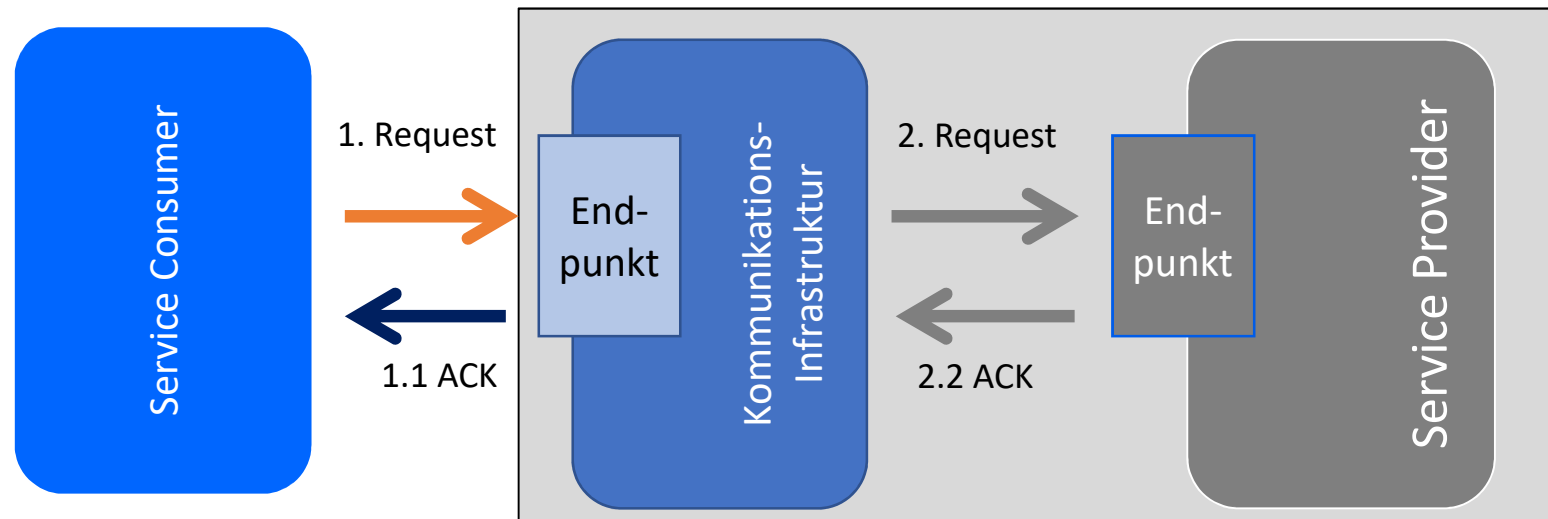
- Die formale Definition der Webservices erfolgt in WSDLs & XSDs.
- Die Webservices müssen mittels SOAP 1.1 bereitgestellt werden.
- ATLAS benötigt für die Zustellung von SOAP-Nachrichten die Webservice-Endpunkte der jeweiligen Teilnehmer-Software.
 - Unter Webservice-Endpunkt ist die erreichbare URL des Webservices gemeint.
 - Diese Informationen werden von der ATLAS-Teilnehmerverwaltung erhoben.

Die abgesicherte Webservice Kommunikation nutzt X.509 Zertifikate.



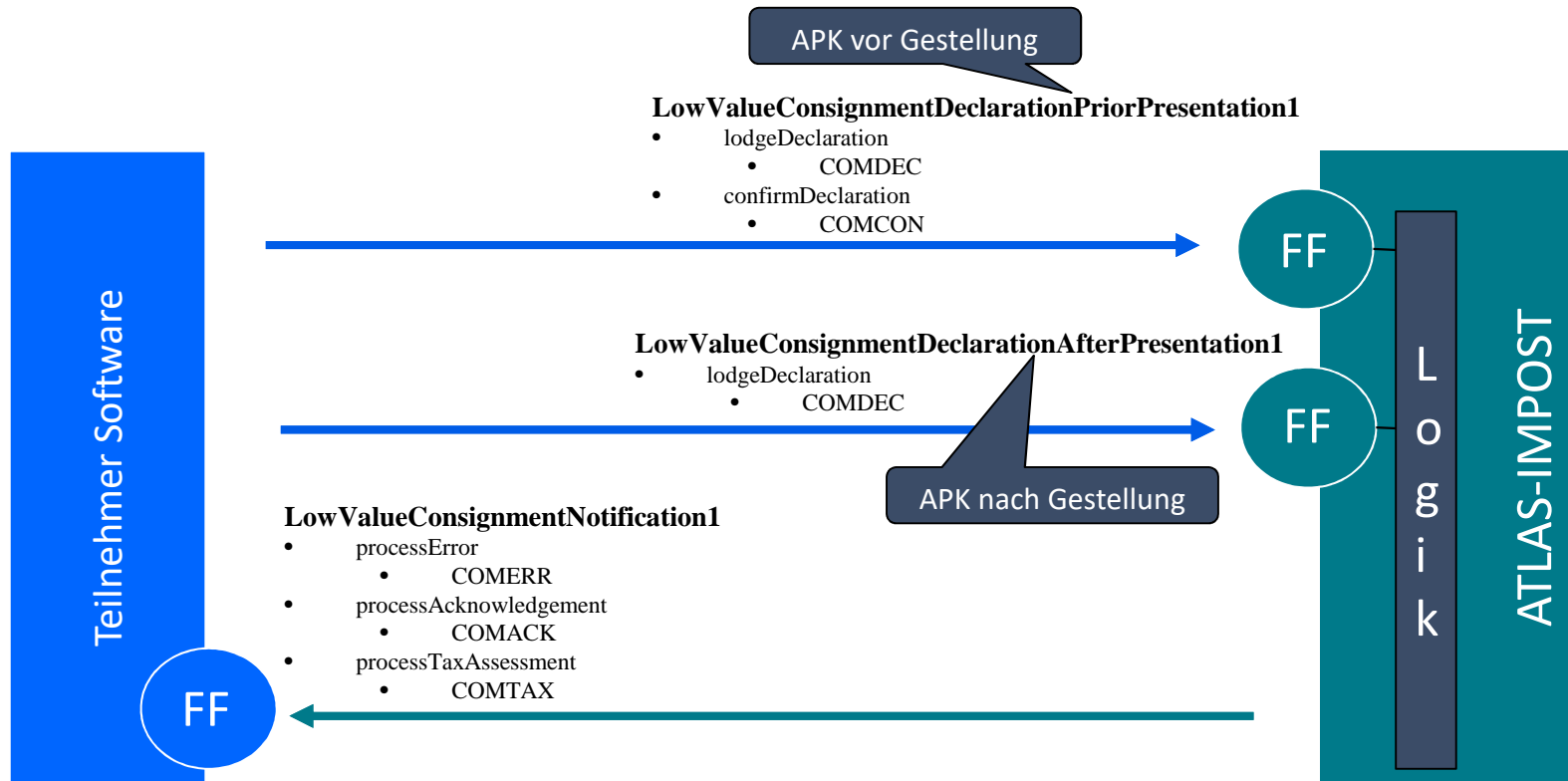
- Nutzung der Standards WS-Security und HTTPs
- Digitales Zertifikat (X.509 Zertifikat) ersetzt die BIN
- Ein Zertifikat je Kommunikations-Teilnehmer zur **Autorisierung**, **Integritätsprüfung** und **Authentifizierung**

Fire & Forget Message Exchange Pattern.



- Das technische Interaktionsmuster der Webservices folgt dem Fire & Forget-Konzept mit **verlässlicher Zustellung (reliable messaging)**
- Die Response (1.1 ACK) ist eine technische Quittung als Bestätigung, dass der Request erfolgreich zur Verarbeitung entgegengenommen wurde.
- Fire & Forget mit „**at least once**“ Auslieferung.
 - Der Service Provider muss die Idempotenz seiner Service-Implementierung sicherstellen.

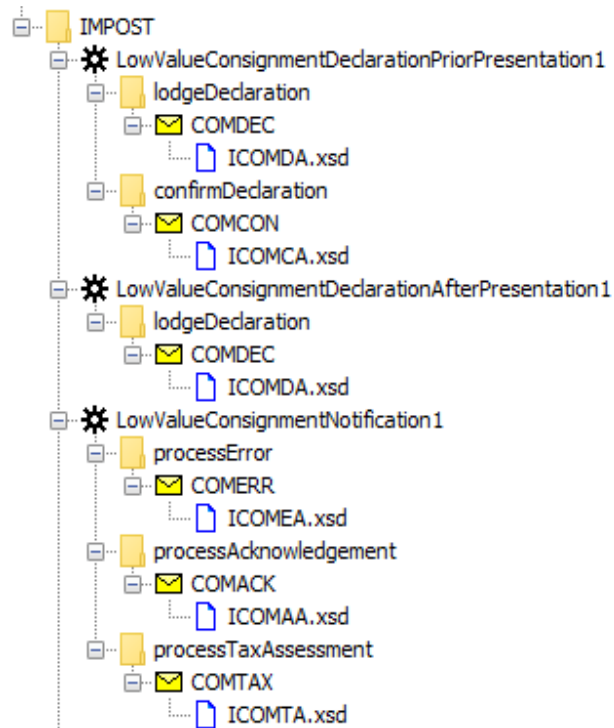
ATLAS-IMPOST Webservice Architektur.



Konkrete Service Endpunkte und Verfügbarkeit siehe TN Merkblatt

Kommunikation im Kontext von ATLAS-IMPOST

ATLAS-IMPOST Webservices und der Zusammenhang zu Operationen und Nachrichten



Webservice	Webservice-Version (1)	WSDL-Version (4)
Verfahrensbereich "IMPOST"		
LowValueConsignmentDeclarationPriorPresentation1	1	1.0
LowValueConsignmentDeclarationAfterPresentation1	1	1.0
LowValueConsignmentNotification1	1	1.0

Quelle: EDI-IHB 10.0

- Formal spezifiziert durch WSDLs/ XSDs und jeweils als ZIP-Datei veröffentlicht:
 - im EDI-IHB in der Datei „ATLAS-Webservices-WSDL.zip“

ATLAS-IMPOST Webservicegruppen.

Webservice-Gruppe	Webservice-Name Provider
LVCDPP	LowValueConsignmentDeclarationPriorPresentation1 Provider: ATLAS-IMPOST
	LowValueConsignmentNotification1 Provider: Teilnehmer
LVCDAP	LowValueConsignmentDeclarationAfterPresentation1 Provider: ATLAS-IMPOST
	LowValueConsignmentNotification1 Provider: Teilnehmer

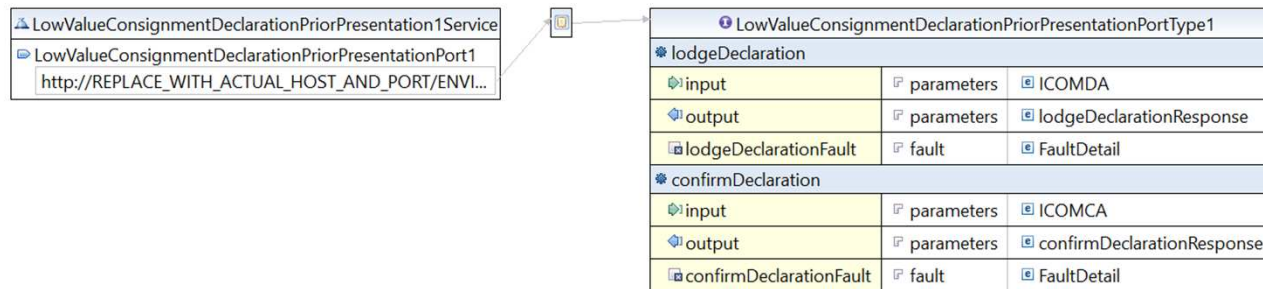
- Fachlich notwendige Webservices für einen Anwendungsfall werden in einer **Webservicegruppe** zusammengefasst.
- Webservicegruppen sind zertifizierungsrelevant!

Mehr dazu im letzten Beitrag!

Kommunikation im Kontext von ATLAS-IMPOST

Seite 8
25.03.2021

WSDL Schema von LowValueConsignmentDeclarationPriorPresentation1



XSDs im Russian Doll Dialekt sind über „include“ in der WSDL referenziert.

*) Die anderen Services siehe Anhang

Kommunikation im Kontext von ATLAS-IMPOST

Seite 9
25.03.2021

Fehlerbehandlung: Grundsätzlich wird zwischen technischen und fachlichen Fehlern unterschieden.

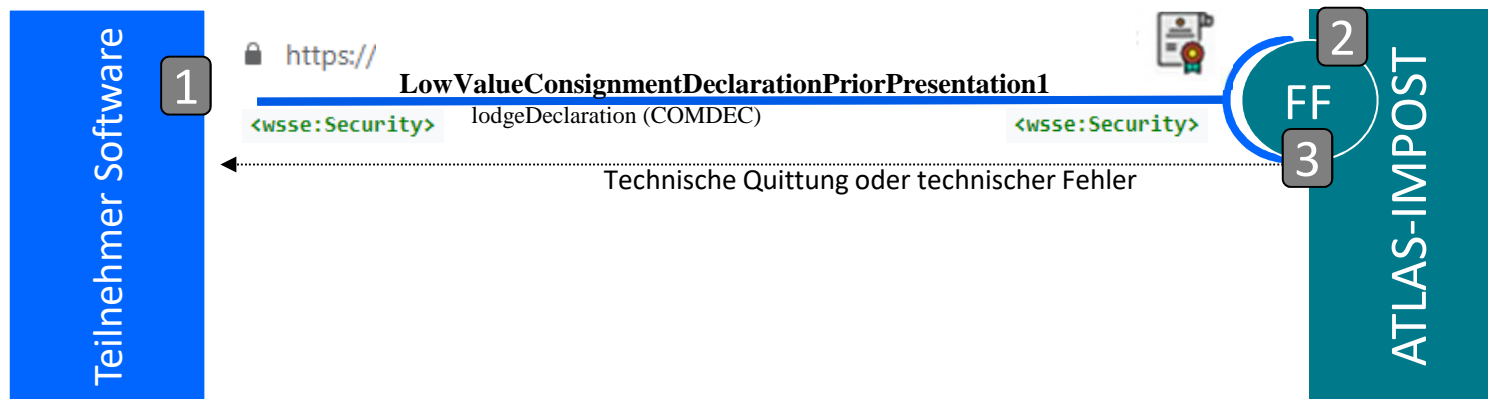
- Trennung technischer Fehler und fachlicher Fehler
 - **Technischer Fehler:**
 - ATLAS-seitig wird jede SOAP-Nachricht gemäß der XSDs validiert.
 - Im Fehlerfall wird der Request mit einem SOAP-Fault abgewiesen.
 - **Fachlicher Fehler:**
 - Service-Operation „processError“ von [LowValueConsignmentNotification1](#)

Webservice Security

- Die Konformität zum WS-Security Standard in den Webservices ist verpflichtend.
 - ATLAS erwartet eine Signatur über den Body sowie den Timestamp des Requests.
 - ATLAS signiert den Body und den Timestamp des Requests und erwartet eine entsprechende Überprüfung in der Teilnehmer-Software (Service Provider).

Beispielhaftes „logisches“ Kommunikationsmuster:

ATLAS-IMPOST ist Provider und die Teilnehmer Software ist Consumer.



1 Die Teilnehmer Software erzeugt eine COMDEC für „lodgeDeclaration“ und signiert den Body und Timestamp des Requests nach XMLDSig und ruft den Service-Endpunkt von ATLAS-IMPOST auf.

2 ATLAS-IMPOST empfängt den Request und prüft die Berechtigung und die Signatur des Requests.

3 Die erfolgreiche Zustellung wird mit einer leeren Response von ATLAS-IMPOST quittiert (im technischen Fehlerfall: SOAP Fault)

Beispielhaftes „logisches“ Kommunikationsmuster:

ATLAS-IMPOST ist Consumer und die Teilnehmer Software ist Provider.

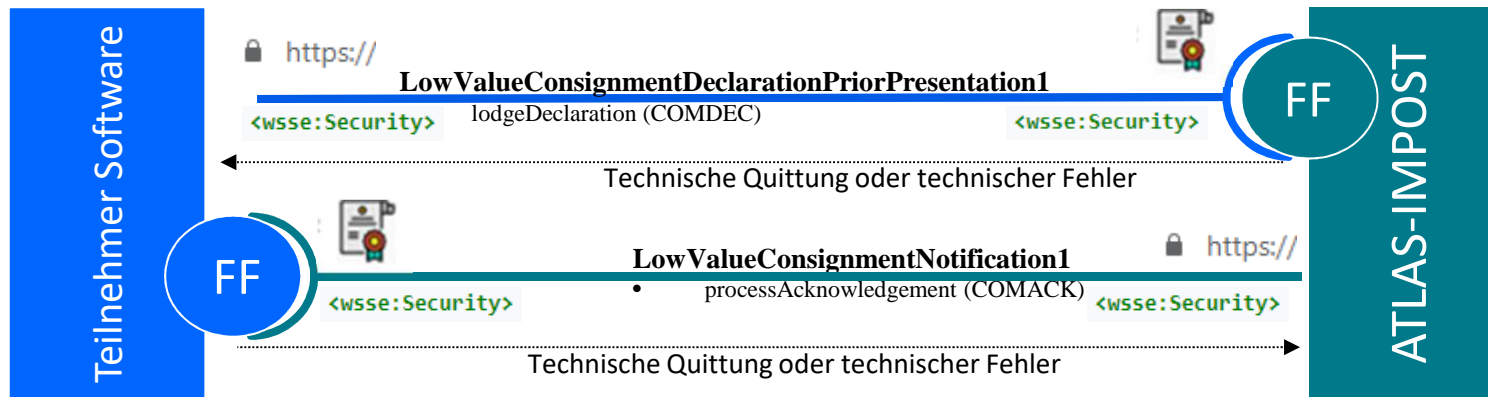


5 Die Teilnehmer Software schickt nach positiver Überprüfung der Signatur des Requests eine technische Quittung zurück und geht in die interne Verarbeitung.

4 ATLAS erstellt eine COMACK für „processAcknowledgement“, signiert den Body und den Timestamp des Requests und ruft den Service-Endpunkt der Teilnehmer Software auf.

Beispielhaftes „logisches“ Kommunikationsmuster:

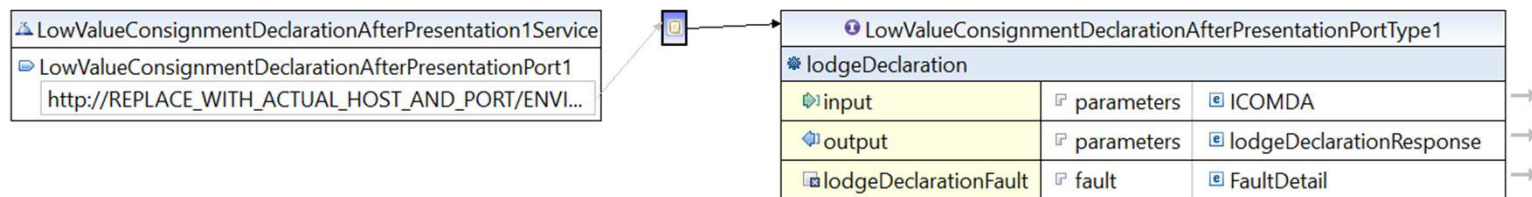
Interaktion zwischen ATLAS-IMPOST und der Teilnehmer Software.



Fragen?

Anhang

LowValueConsignmentDeclarationAfterPresentation1



LowValueConsignmentNotification1

