
Spezifikation – XTA Erweiterung Service Reporting

Version 1 / Final Draft – 26. Oktober 2023

Der Standard XTA 2 wird im Auftrag des IT-Planungsrates von der KoSIT betrieben und innerhalb einer definierten Gremienstruktur weiterentwickelt. Die vorliegende Spezifikation und weitere Informationen sind auf den Seiten der KoSIT unter www.xoev.de zu finden.

Inhaltsverzeichnis

Einleitung	1
1 Voraussetzungen	2
1.1 Konformität	2
2 Kommunikationsmodell	3
2.1 Anwendungsebene	3
2.2 Transportebene	3
2.2.1 Schnittstellentyp extension.serviceReporting.transport.base	3
3 Datenmodell	5
3.1 ReportSelector	5
3.2 ReportSelectorType	5
3.3 ServiceReportType	6
3.4 SelectionResultList	6
3.5 SelectionResultType	7
4 Fehlerbehandlung	8
4.1 TooManyResultsException	8
A Eingebundene externe Modelle/Standards	9
A.1 XML-Signature	9
A.2 XTA 2 Version 5	9
B Versionshistorie	10
B.1 Release E, Version 1 / Final Draft (26. Oktober 2023)	10

Einleitung

Diese Spezifikation wendet sich an Entwicklerinnen und Entwickler, welche die von XTA bereitgestellte Funktionalität in ihren Anwendungen implementieren.

Die vorliegende Erweiterung für den Standard XTA 2 nutzt die in XTA 2 Service Profilen definierte Basisstrukturen. Damit wird eine Datenstruktur für einen Bericht zur Erfassung von geleisteten Servicequalitäten dargelegt. Für den Abruf des Berichts wird eine neue Methode (siehe Abschnitt [Kapitel 2 auf Seite 3](#)) ergänzend zum XTA Kern zur Verfügung gestellt.

1 Voraussetzungen

Die Erweiterung *Service Reporting* verwendet in der vorliegenden Version Datentypen aus XTA 2 Version 5 und kann somit nur der mit genannten Version des Kerns eingesetzt werden. Um die Überprüfbarkeit der gesamten Transportstrecke (vom Autor bis zum Leser) gewährleisten zu können, muss die Erweiterung von allen Rollen bzw. beteiligten Software-Produkten implementiert werden.

1.1 Konformität

Gegenstand der Konformitätsbewertung ist ein Software-Produkt, welches XTA für die Anwendungsebene oder für die Transportebene implementiert. Eine Implementierung für die Anwendungs-/Transportebene ist genau dann „konform zur Spezifikation *Service Reporting Version 1*“, wenn alle Vorgaben aus der Spezifikation an die umgesetzte Rolle erfüllt wurden. Die Konformitätserklärung der Produktherstellerin erfolgt freiwillig und muss mindestens über die folgenden Informationen verfügen:

- Eindeutige Kennzeichnung als Selbsterklärung zur XTA-Konformität
- Eindeutiger Verweis auf die XTA-Spezifikation mit Angabe von Version und Datum, die umgesetzt wurde
 - Verweis auf die Quelle der Spezifikation (Link zur offiziellen Seite)
 - Es kann stets nur auf die XTA-Spezifikation verwiesen werden, nicht auf die Hilfsmittel, die zusammen mit der XTA-Spezifikation veröffentlicht werden (z.B. einzelne Schema- oder WSDL-Dateien)
- Eindeutige Identifikation des Produktes, insbesondere die Produktbezeichnung einschließlich Versionsangabe
 - Angabe, welche Rollen umgesetzt worden sind
- Eindeutige Angabe der Herstellerin einschließlich gültiger Kontaktinformationen
- Datum, Ort und Unterschrift einer vertretenden Person des Herstellers bzw. der Herstellerin

Die Fachverfahren beauftragen die Transportverfahren mit der Abwicklung eines Nachrichtentransports. Mit dem Modul *XTA Service Profile* können die Fachverfahren die erwarteten Servicequalitäten für den Betrieb bestimmen. Die Transportverfahren sind damit verpflichtet den Betrieb stets ordnungsgemäß durchzuführen. Die vorliegende Erweiterung für das Modul *XTA Webservice* definiert, wie ein Service Profil je Transportauftrag vorgegeben und die Erreichung von geforderten Servicequalitäten überprüft werden können.

2 Kommunikationsmodell

In diesem Kapitel werden die Schnittstellen für die Kommunikation zwischen der Anwendungsebene und Transportebene beschrieben. Hiermit wird auch festgelegt, welche Rolle welche Schnittstellen bereitstellen und aufrufen können muss.

2.1 Anwendungsebene

Der Leser bzw. Autor muss nachfolgende Schnittstellen des Empfängers bzw. Senders aufrufen können:

- [getServiceReport](#)

2.2 Transportebene

Der Empfänger bzw. Sender muss dem Leser bzw. Autor nachfolgende Methoden bereitstellen:

- [getServiceReport](#)

2.2.1 Schnittstellentyp `extension.serviceReporting.transport.base`

In diesem Schnittstellentyp sind alle Methoden zusammengefasst, die vom Sender dem Autor und vom Empfänger dem Leser zur Verfügung gestellt werden müssen.

2.2.1.1 Methode `getServiceReport`

Die Methode liefert dem Autor Berichte über die Einhaltung der Servicequalitäten des Versands der Nachrichten. Wird die Funktion vom Leser aufgerufen, werden die Berichte über die Einhaltung der Servicequalitäten beim Empfang der Nachrichten geliefert. Ein Abruf der Berichte (auch im Falle einer Vertretung: Header/Party weicht vom ReportSelector/ReaderIdentifier oder AuthorIdentifier ab) muss seitens Sender bzw. Empfänger mittels expliziter Berechtigungen, definiert durch die Fachverfahren, kontrolliert werden. Alle Abrufe werden beim Sender bzw. Empfänger zur Zugriffsnachverfolgbarkeit intern protokolliert.

2.2.1.1.1 Exceptions

`ParameterNotSupportedException`

Die Exception signalisiert den Aufruf einer Methode mit spezifikationsgemäßen, aber von der Implementierung nicht unterstützten Parametern.

`PermissionDeniedException`

Die Exception signalisiert einen Autorisierungsfehler.

`TechnicalProblemException`

Die Exception signalisiert einen allgemeinen Fehler während der Verarbeitung.

`TooManyResultsException`

Die Exception signalisiert, dass die Ergebnismenge für die gewählten Parameter eines Abrufs zu groß ist.

2.2.1.1.2 Input

Tabelle 2.1. Header

WSDL Message	Message Part	Part Element
XTAHeader	Party	xta-core:Party (vgl. Seite 9)
Element des Typs PartyType mit Informationen, die zur Autorisierung von Methodenaufrufen verwendet wird. Es ergänzt damit die Informationen, die bereits aus dem Verbindungsaufbau (z.B. HTTP) vorhanden sind.		

Tabelle 2.2. Body

WSDL Message	Message Part	Part Element
GetServiceReportRequest	ReportSelector	serviceReporting:ReportSelector (vgl. Seite 5)
Hiermit muss die Auswahl- oder Filterung von ServiceReports vorgenommen werden.		

2.2.1.1.3 Output

Keine Header-Informationen.

Tabelle 2.3. Body

WSDL Message	Message Part	Part Element
GetServiceReportResponse	SelectionResultList	serviceReporting:SelectionResultList (vgl. Seite 6)
Hiermit können die ServiceReports unter Berücksichtigung der gesetzten Auswahl- oder Filterkriterien übertragen werden.		

3 Datenmodell

In diesem Kapitel werden die innerhalb der Schnittstellen verwendeten Datentypen für die Kommunikation zwischen der Anwendungsebene und Transportebene beschrieben.

3.1 ReportSelector

Globales Element: *ReportSelector*

Element des Typs *ReportSelectorType* (siehe [Abschnitt 3.2 auf Seite 5](#)).

3.2 ReportSelectorType

Typ: *ReportSelectorType*

Parameter für die Auswahl/Filterung von ServiceReports unter Einbeziehung der Autorisierung aus dem Verbindungsaufbau.

Kindelemente von <i>ReportSelectorType</i>				
Kindelement	Typ	Anz.	Ref.	Seite
Auswahl: XTAMessageID	<i>xta-core:XTAMessageIDType</i>	1..n	A.2	9
Der eindeutige Identifikator des Transportauftrags.				
Auswahl: OtherCriteria		1		
AuthorIdentifier	<i>xta-core:PartyType</i>	0..1	A.2	9
Eindeutiges Identifikationsmerkmal zum Abruf der Berichte, die im Zuständigkeitsbereich eines bestimmten Autoren liegen.				
ReaderIdentifier	<i>xta-core:PartyType</i>	0..1	A.2	9
Eindeutiges Identifikationsmerkmal zum Abruf der Berichte, die im Zuständigkeitsbereich eines bestimmten Lesers liegen.				
EntryTimeFrom	<i>xs:dateTime</i>	0..1		
Untergrenze für den Eingangszeitpunkt der auszuwählenden Nachrichten. Der Eingangszeitpunkt (Delivery) ist größer oder gleich dieser Untergrenze.				
EntryTimeTo	<i>xs:dateTime</i>	0..1		
Obergrenze für den Eingangszeitpunkt der auszuwählenden Nachrichten. Der Eingangszeitpunkt (Delivery) ist echt kleiner als diese Obergrenze.				
MaxReports	<i>xs:positiveInteger</i>	0..1		
Vom Leser vorgegebene maximale Anzahl von Reports, die beim Aufruf der Methode <i>getServiceReport</i> zurückgeliefert werden sollen. Diese Obergrenze kann durch den Sender oder Empfänger selbstständig für die Postfachinstanz auf jeden Wert größer null beschränkt werden.				
Extensions	<i>xta-core:ExtensionsType</i>	0..1	A.2	9
Extensions für eine flexible Erweiterung der Selektionskriterien.				

3.3 ServiceReportType

Typ: *ServiceReportType*

Struktur eines Berichts über die Einhaltung der Servicequalitäten während der technischen Abarbeitung eines Transportauftrages. Er wird von Autoren und Lesern dazu verwendet, die Einhaltung des Transportauftrags bzw. der Servicequalitäten zu kontrollieren. Entsprechend können Sender und Empfänger mittels eines ServiceReport nachweisen, dass der Transportauftrag eingehalten wurde. Sender erstellen einen ServiceReport für Autoren, Empfänger einen ServiceReport für Leser. Stehen für einen Transportauftrag sowohl der ServiceReport des Senders als auch des Empfängers zur Verfügung, kann die Einhaltung des Transportauftrags vom Versand bis zum Empfang nachgewiesen bzw. kontrolliert werden.

Mit dem ServiceReport soll insbesondere auch die rechtskonforme Funktionsweise der den Transport durchführenden Infrastruktur nachvollzogen werden können. Der ServiceReport ermöglicht Aussagen über die Einhaltung gesetzlicher Anforderungen und die Erfüllung von Vorgaben aus Serviceprofilen. Er dient ferner als zusätzliches Mittel zur Fehlerverfolgung in Transportprozessen.

Die Ereignisse im ServiceReport werden vom Sender und Empfänger während der Abarbeitung von Transportaufträgen protokolliert. Aus diesen Ereignissen erstellen Sender und Empfänger den ServiceReport, wenn dieser durch den Autor oder den Leser angefordert wird. Autor oder Leser bestimmen beim Abruf, zu welchen Transportaufträgen ServiceReport-Ereignisse zu einem ServiceReport zusammengestellt werden sollen. Dazu können die Selektionskriterien gemäß ReportSelector ([Abschnitt 3.1 auf Seite 5](#)) verwendet werden. Der auf dieser Basis erstellte ServiceReport enthält immer alle Ereignisse zu den so bestimmten Transportaufträgen. Autoren und Leser fordern einen ServiceReport über eine entsprechende XTA-Webservice-Methode an.

Kindelemente von <i>ServiceReportType</i>				
Kindelement	Typ	Anz.	Ref.	Seite
id	<i>xs:ID</i>	1		
Hier muss die ID zur eindeutigen Referenzierung des XML-Elements eingetragen werden. Primär wird die Referenz für die Erstellung der Signatur im <i>SelectionResult</i> benötigt.				
MessageMetaData	<i>xta-core:MessageMetaDataType</i>	1	A.2	9
Der Transportauftrag zu einer Nachricht.				
XTAServerIdentity	<i>xs:anyURI</i>	1		
Hier gibt Sender oder Empfänger seine Identität beziehungsweise die seiner Software-Instanz an, indem er z. B. die Server-IP-Adresse oder die URI seines XTA-WS einträgt. Im Aufbau der URI soll erkennbar sein, welcher Organisation der Ersteller angehört.				
ReportTime	<i>xs:dateTime</i>	1		
Zeitpunkt der letzten Aktualisierung des Protokolls. Ist bei Fortschreibung des Protokolls zu überschreiben.				
EreignisList	<i>xta-sp:EreignisListType</i>	1	A.2	9

3.4 SelectionResultList

Globales Element: *SelectionResultList*

Kindelement von <i>SelectionResultList</i>				
Kindelement	Typ	Anz.	Ref.	Seite
SelectionResult	<i>serviceReporting:SelectionResultType</i>	0..n	3.5	7

3.5 SelectionResultType

Typ: *SelectionResultType*

Kindelemente von <i>SelectionResultType</i>				
Kindelement	Typ	Anz.	Ref.	Seite
ServiceReport	<i>serviceReporting:ServiceReportType</i>	1	3.3	6
Signature	<i>ds:SignatureType</i>	0..1	A.1	9
Falls der Report signiert ist, findet sich hier die Signatur.				

4 Fehlerbehandlung

Die vorliegende Erweiterung definiert eigene Exceptions und eine Codeliste für die Beschreibung der weiteren möglichen Fehlerfälle. Die Codeliste kann im XRepository (www.xrepository.de) unter Angabe folgender Kennung abgerufen werden:

`urn:xoev-de:kosit:codeliste:xta.service-reporting.fehlernummer`

4.1 TooManyResultsException

Exception: *TooManyResultsException*

Die Exception signalisiert, dass die Ergebnismenge für die gewählten Parameter eines Abrufs zu groß ist.

Der Grenzwert der Ergebnismenge wird außerhalb der XTA-Spezifikation, also beispielsweise vom Betreiber des XTA-Servers, festgelegt.

A Eingebundene externe Modelle/Standards

Folgende externe Modelle werden in dieser Spezifikation verwendet und sind auf den XÖV-Webseiten (siehe <http://www.xoev.de/de/produkte>) oder im XRepository (siehe <http://www.xrepository.de>) veröffentlicht:

A.1 XML-Signature

xmldsig-core; Version 1.1

Folgende Datentypen aus dem externen Modell werden in dieser Spezifikation verwendet:

- SignatureType

A.2 XTA 2 Version 5

XTA 2; Version 5

Folgende Datentypen aus dem externen Modell werden in dieser Spezifikation verwendet:

- EreignisListType
- ExceptionType
- ExtensionsType
- InvalidMessageIDException
- MessageMetaDataType
- MessageSchemaViolationException
- MessageVirusDetectionException
- NotImplementedException
- ParameterIsNotValidException
- ParameterNotSupportedException
- Party
- PartyType
- PermissionDeniedException
- SyncAsyncException
- TechnicalProblemException
- XTAMessageIDType

B Versionshistorie

B.1 Release E, Version 1 / Final Draft (26. Oktober 2023)

CR 2021-08 Erweiterung Service Reporting

Im Rahmen der Aufteilung von XTA 2 in einen Kern und die Erweiterungen wurden die Inhalte dieser Spezifikation als eine Erweiterung definiert. Ein Abruf der Berichte muss seitens Sender bzw. Empfänger mittels expliziter Berechtigungen, definiert durch die Fachverfahren, kontrolliert werden. Alle Abrufe werden beim Sender bzw. Empfänger zur Zugriffsnachverfolgbarkeit intern protokolliert.

Die Ausgabe von mehreren ServiceReports ist mit der Methode `getServiceReport` möglich indem der Output der Methode durch `SelectionResultList` ersetzt wurde.

Der `ServiceReport` kann nun einfacher vom Sender und Empfänger signiert werden: `Element Signature` liegt neben dem `ServiceReport` im `SelectionResult`. Das Element `ReportSelector` wurde entfernt und das Element `MessageMetaData` hinzugefügt.

Der `ReportSelector` wurde überarbeitet: Er lässt eine Auswahl mehrerer `XTAMessageID` oder anderer Kriterien zu. Als anderes Kriterium wurde `MaxReports` ergänzt, damit der Umfang der Rückgabe limitiert werden kann. Weitere Kriterien können als `Extensions` anhand eigener Erweiterungen definiert werden.

QSSPEC 2019-16 QS-Anforderungen Transport mit ServiceProfil-Datenstruktur verknüpfen

Die Vorgaben im Bezug auf die Rollen (alle Beteiligten müssen die Funktionalität implementieren) wurden im Kapitel Konformität hinzugefügt.

CR 2021-02 Trennung des Kerns und der Erweiterungen - Datentypen

`ServiceReport` wurde eine `xs:ID` hinzugefügt und zur verpflichtenden Angabe erklärt.