

---

# Spezifikation – XTA Erweiterung Notification

*Version 1 / 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](http://www.xoev.de) zu finden.*

# Inhaltsverzeichnis

Einleitung .....	1
1 Voraussetzungen .....	2
1.1 Konformität .....	2
2 Kommunikationsmodell .....	3
2.1 Anwendungsebene .....	3
2.1.1 Schnittstellentyp extension.notification.author .....	3
2.1.2 Schnittstellentyp extension.notification.reader .....	4
2.2 Transportebene .....	5
3 Datenmodell .....	6
3.1 Zusätzliche Parametrisierung des XTA 2 Kerns .....	6
3.1.1 MessageMetaDataExtensionType .....	6
3.2 NotifyAuthorType .....	6
4 Fehlerbehandlung .....	7
4.1 NotificationForFetchedMessageException .....	7
A Eingebundene externe Modelle/Standards .....	8
A.1 XTA 2 Version 5 .....	8
B Versionshistorie .....	9
B.1 Release E, Version 1 / Draft (26. Oktober 2023) .....	9

# 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 behandelt Benachrichtigungen im Rahmen des asynchronen Versands von Nachrichten. Sie unterteilt sich in Benachrichtigungen an den Autor über die erfolgreiche Übergabe von Nachrichten an den Empfänger (siehe Abschnitt [Kapitel 2 auf Seite 3](#)) und die Benachrichtigung von Lesern über den Eingang von neuen Nachrichten (siehe Abschnitt [Abschnitt 3.1 auf Seite 6](#)). Für die Benachrichtigung werden neue Methoden und zusätzliche Parameter für den als Erweiterung zu dem XTA Kern zur Verfügung gestellt.

# 1 Voraussetzungen

---

Die Erweiterung Notification Version 1 verwendet Datentypen aus XTA 2 Version 5 und kann somit nur mit der genannten Version des Kerns eingesetzt werden.

## 1.1 Konformität

Gegenstand der Konformitätsbewertung ist ein Software-Produkt, welches den Transportadapter für die Anwendungsebene oder für die Transportebene implementiert. Eine Implementierung für die Anwendungs-/Transportebene ist genau dann „konform zur Spezifikation Erweiterung Notification Version 1“, wenn alle Vorgaben aus der Spezifikation an die umgesetzte Rolle erfüllt wurden. Die Konformitätserklärung der Produktherstellerin erfolgt freiwillig und muss mindestens über folgende 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

## 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.

Der Parameter `MessageMetaData` des XTA Kerns ist mit zusätzlichen Inhalten zu belegen (siehe [MessageMetaDataExtensionType](#)).

### 2.1 Anwendungsebene

Der Autor muss dem Sender nachfolgende Methoden im asynchronen Kommunikationsszenario bereitstellen:

- [notifyMessageSent](#)

Der Leser muss dem Empfänger nachfolgende Methoden im asynchronen Kommunikationsszenario bereitstellen:

- [notifyMessageReceived](#)

#### 2.1.1 Schnittstellentyp `extension.notification.author`

In diesem Schnittstellentyp sind die Methoden des XTA-WS zusammengefasst, die vom Autor für den Sender zur Verfügung gestellt werden müssen.

##### 2.1.1.1 Methode `notifyMessageSent`

Die Methode `notifyMessageSent` ist vom Autor bereitzustellen und wird vom Sender aufgerufen, um den Autor über die Zustellung einer Nachricht zu informieren.

###### 2.1.1.1.1 Exceptions

`InvalidMessageIDException`

Die Exception signalisiert, dass zu der angegebenen `XTAMessageID` keine Protokollinformationen für die Autorisierung vorliegen.

`NotImplementedException`

Die Exception signalisiert, dass diese Methode nicht implementiert wurde.

`ParameterIsValidException`

Die Exception signalisiert fehlende oder fehlerhafte Eingabeparameter.

`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.

### 2.1.1.1.2 Input

**Tabelle 2.1. Header**

WSDL Message	Message Part	Part Element
XTAHeader	Party	xta-core:Party (vgl. <a href="#">Seite 8</a> )
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
NotifyMessageSentRequest	ReportResult	xta-core:ReportResult (vgl. <a href="#">Seite 8</a> )
Hiermit muss der Transportbericht des Transportauftrags übertragen werden, zu dessen Versand mit der Notifikation informiert wird.		

### 2.1.1.1.3 Output

Keine Header-Informationen.

**Tabelle 2.3. Body**

WSDL Message	Message Part	Part Element
EmptyBody		
Keine Body-Informationen.		

## 2.1.2 Schnittstellentyp `extension.notification.reader`

In diesem Schnittstellentyp sind die Methoden des XTA-WS zusammengefasst, die vom Leser für den Empfänger zur Verfügung gestellt werden müssen.

### 2.1.2.1 Methode `notifyMessageReceived`

Mit der Methode `notifyMessageReceived` informiert der Empfänger den Leser über den Eingang einer Nachricht. Die Methode wird nur für die vom Leser mit dem Empfänger vereinbarten Postfächer verwendet. Der Leser wird für jede Nachricht der vereinbarten Postfächer nur einmal über deren Eingang informiert.

#### 2.1.2.1.1 Exceptions

`NotImplementedException`

Die Exception signalisiert, dass diese Methode nicht implementiert wurde.

`NotificationForFetchedMessageException`

Die Exception signalisiert eine bereits erfolgte Abholung der Fachnachricht.

`ParameterIsValidException`

Die Exception signalisiert fehlende oder fehlerhafte Eingabeparameter.

**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.

**2.1.2.1.2 Input****Tabelle 2.4. Header**

WSDL Message	Message Part	Part Element
XTAHeader	MessageMetaData	xta-core:MessageMetaData (vgl. <a href="#">Seite 8</a> )
In dieser Struktur werden die Steuerungsdaten des Transportauftrags festgelegt.		
XTAHeader	Party	xta-core:Party (vgl. <a href="#">Seite 8</a> )
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 2.5. Body**

WSDL Message	Message Part	Part Element
EmptyBody		

**2.1.2.1.3 Output****Tabelle 2.6. Header**

WSDL Message	Message Part	Part Element
XTAHeader	MessageMetaData	xta-core:MessageMetaData (vgl. <a href="#">Seite 8</a> )
In dieser Struktur werden die Steuerungsdaten des Transportauftrags festgelegt.		

**Tabelle 2.7. Body**

WSDL Message	Message Part	Part Element
EmptyBody		

**2.2 Transportebene**

Der Sender muss nachfolgende Methoden des Autors aufrufen können:

- [notifyMessageSent](#)

Der Empfänger muss nachfolgende Methoden des Lesers aufrufen können:

- [notifyMessageReceived](#)

## 3 Datenmodell

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

### 3.1 Zusätzliche Parametrisierung des XTA 2 Kerns

Die Datentypen der nachfolgenden Unterkapitel werden über die XTA 2 Kernparameter gemäß der Spezifikation XTA 2 Version 5 zusätzlich übertragen.

#### 3.1.1 MessageMetaDataExtensionType

Typ: *MessageMetaDataExtensionType*

Dieser Typ beschreibt zusätzliche Parameter, die beim Aufruf der XTA Webservice Kernmethode `sendMessage` unter `MessageMetaData/Extensions` übergeben werden müssen.

Kindelement von <i>MessageMetaDataExtensionType</i>				
Kindelement	Typ	Anz.	Ref.	Seite
<b>NotifyAuthor</b>	<i>notification:NotifyAuthorType</i>	0..1	<a href="#">3.2</a>	<a href="#">6</a>
Benachrichtigung des Autors über erfolgreiche Zustellung erfordert.				

### 3.2 NotifyAuthorType

Typ: *NotifyAuthorType*

Angaben für die Benachrichtigung des Autors nach Zustellung einer Nachricht.

Kindelemente von <i>NotifyAuthorType</i>				
Kindelement	Typ	Anz.	Ref.	Seite
<b>NotifyAuthorURI</b>	<i>xs:anyURI</i>	0..1		
ServiceURI, unter welcher der Autor über die erfolgte Zustellung informiert werden kann.				
<b>NotifyAuthorEnabled</b>	<i>xs:boolean</i>	1		
Benachrichtigung des Autors über erfolgreiche Zustellung erfordert.				
Dieses Element/Attribut hat den Wert „false“, sofern kein anderer Wert übermittelt wird (default-Wert).				



## 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](http://www.xrepository.de)) unter Angabe folgender Kennung abgerufen werden:

`urn:xoev-de:kosit:codeliste:xta.notification.fehlernummer`

### 4.1 NotificationForFetchedMessageException

Exception: *NotificationForFetchedMessageException*

Die Exception signalisiert eine bereits erfolgte Abholung der Fachnachricht.

Der Fehler kann pro Nachricht mehrfach auftreten, z.B. aufgrund von Latenzen. Leser und Empfänger müssen festlegen, wie damit umgegangen wird, wenn der Fehler mehrfach auftritt.

# 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 XTA 2 Version 5

XTA 2; Version 5

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

- ExceptionType
- InvalidMessageIDException
- MessageMetaData
- MessageSchemaViolationException
- MessageVirusDetectionException
- NotImplementedException
- ParameterIsNotValidException
- ParameterNotSupportedException
- Party
- PermissionDeniedException
- ReportResult
- SyncAsyncException
- TechnicalProblemException

# B Versionshistorie

---

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

### CR 2021-10 Erweiterung Notification

Im Rahmen der Aufteilung von XTA 2 in einen Kern und die Erweiterungen wurden die Inhalte dieser Spezifikation als eine Erweiterung definiert.

`ReceiptRequests` bzw. Quittungen werden zukünftig in einer eigenen Erweiterung für die Quittungsmechanismen weitergeführt (nicht Bestandteil dieses Releases).