



einheitliches XML-basiertes Transportverfahren

eXTra Basis-Standard

eXTra Standardnachrichten Version 1.3

**Schnittstellenbeschreibung
Ausgabestand 1.3.0**

FINAL



Herausgeber:

AWV – Arbeitsgemeinschaft für wirtschaftliche Verwaltung e. V.
Düsseldorfer Str. 40
65760 Eschborn
Vereinsregister 73 VR 5158, Amtsgericht Frankfurt am Main
Telefon: 0 61 96/7 77 26-0
Fax: 0 61 96/7 77 26-51
Mail: info@awv-net.de
Web: www.extra-standard.de, www.awv-net.de.

Änderungsprotokoll für Version 1.3. Ausgabestand 1.3.0

Autor[en]	Datum	Beschreibung
[gelöscht]	24.10.2011	Erstellung 1.3 auf Basis Final 1.2 Ausgabestand 1.2.1 Codelisten <code>DataRequestPropertyNamesType</code> & <code>PropertyNamesType</code> um die Properties <code>Request*</code> und <code>Layer</code> erweitert sowie Properties aus beliebigen Domains zugelassen. Codeliste <code>StatusRequestPropertyNamesType</code> Nachricht <code>StatusRequest</code> Nachricht <code>StatusResponse</code> Nachricht <code>ListOfStatusResponse</code> Nachricht <code>RepeatResponse</code>

Änderungsprotokoll für Version 1.2. Ausgabestand 1.2.1

Autor[en]	Datum	Beschreibung
[gelöscht]	29.07.2010	Erstellung 1.2 auf Basis Final 1.1 <code>PropertySet: Attribut event</code> Nachricht <code>ListOfConfirmationOfReceipt</code> Nachricht <code>ListOfDataRequest</code>

Änderungsprotokoll für Version 1.1. bis 1.1.1

Autor[en]	Datum	Beschreibung
[gelöscht]	01.11.2009	Erstellung 1.1
	08.03.2010	Freigabe

Inhaltsverzeichnis

Einleitung und Übersicht.....	6
Layout und Notationen	8
Formatierungen	8
Beschreibung der Regeln	8
Beschreibung der Elemente und Attribute	8
Graphische Symbole	10
Definitionen [normativ].....	13
Begriffe.....	13
Namensräume und Präfixe	13
eXtra Codelisten [normativ]	15
Codeliste DataRequestPropertyNames	16
Codeliste EventNames	16
Codeliste LayerNames	17
Codeliste PropertyNames.....	17
Codeliste StatusNames	18
Codeliste StatusRequestPropertyNames.....	18
Codeliste WeightCodes	19
Codeliste XSDTypeCodes	19
Nachrichtentyp ConfirmationOfReceipt [normativ]	22
Beispieldokument	23
Element ConfirmationOfReceipt	24
Element Property	26
Element PropertySet	29
Element Value	31
Nachrichtentyp DataRequest [normativ]	32
Beispieldokument	33
Element DataRequest	35
Element Argument.....	37
Element Control.....	40
Element EQ.....	41
Element GE	42
Element GT	43
Element IN.....	44
Element LE.....	45
Element LT	46
Element MaximumMessages.....	47
Element MaximumPackages	48
Element MaximumResults	49
Element MaximumSize	50
Element Query	51
Nachrichtentyp ListOfConfirmationOfReceipt [normativ].....	52
Beispieldokument	53
Element ListOfConfirmationOfReceipt	56
Nachrichtentyp ListOfDataRequest [normativ]	57
Beispieldokument	58
Element ListOfDataRequest	60
Nachrichtentyp ListOfStatusResponse [normativ].....	61
Beispieldokument	62
Element ListOfStatusResponse	63
Nachrichtentyp RepeatResponse [normativ].....	64
Nachrichtentyp StatusRequest [normativ].....	65



Beispieldokument	66
Element StatusRequest	67
Element Argument	68
Element Query	70
Nachrichtentyp StatusResponse [normativ]	71
Beispieldokument	72
Element StatusResponse	73
Element Checkpoint	74
Element Code	75
Element Flag	76
Element Layer	77
Element Report	78
Element Status	79
Element Text	80
Element Timestamp	81
Element Trace	82
Ressourcen	83
XSD Schema Ressourcen	83
Literatur	84

Einleitung und Übersicht

Gültigkeit und Versionierung: Dieses Dokument enthält die normative technische Spezifikation der Schnittstelle der eXtra Standardnachrichten in der Version 1.3.0 des eXtra Basis Standards.

In der Versionsangabe dieses Dokuments V *n.m.r* ist in den ersten beiden Ziffern *n.m* die jeweilige Version (hier Version 1.3) der zugrundeliegenden Schnittstelle der eXtra Standardnachrichten angeführt, während die letzte Ziffer *r* den jeweiligen Ausgabestand (hier Ausgabestand 0) wiedergibt.

Hinweis zum Gebrauch: Bibliographische Referenzen stehen in eckigen Klammern ([*bibref*]) und sind am Ende dieser Spezifikation beschrieben.

eXtra, das *einheitliche XML-basierte Transportverfahren*, ist ein gemeinschaftlich von Unternehmen und Behörden entwickelter, offener Standard für die Datenübermittlung. Eine kurze Einführung sowie eine detaillierte Beschreibung finden sich in [EINF] bzw. [KOMP]. Diese Dokumente und sämtliche öffentlichen Informationen über eXtra sind im Internet unter der Adresse <http://www.extra-standard.de> abrufbar.

Dieses Dokument enthält die normative technische Spezifikation der eXtra *Standardnachrichten*. Standardnachrichten sind kein Teil der eXtra Kommunikationsschicht, sondern als Nutzdaten zu betrachten und wie diese zu verarbeiten. Aus diesem Grund befinden sie sich in einem separaten Namensraum und haben keinerlei Abhängigkeit zu den Schemata der eXtra Kommunikationsschicht. Sie standardisieren einige in vielen Anwendungsszenarien vorkommende Nachrichten wie z.B. Empfangsbestätigungen, und stellen damit eine weitere Vereinfachung bei der Einführung und Verwendung des eXtra Standards dar.

Diese Spezifikation beschreibt die Strukturen und Felder der eXtra Standardnachrichten. Regeln und Empfehlungen für den Entwurf, sowie die Implementierung eines Datenübermittlungssystems auf Basis von eXtra sind im Designleitfaden [DSIG] und im Implementierungsleitfaden [IMPL] beschrieben. Diese Dokumente bilden die Grundlage für den Entwurf und die Implementierung der eXtra Kommunikationsschnittstelle und richten sich vor allem an Anwendungsarchitekten und Entwickler. Die Profilierung der eXtra Kommunikationsschnittstelle ist in [PROF] erläutert.

eXtra Standardnachrichten basieren auf der *Extensible Markup Language – XML* – in der Version 1.0, wie beschrieben durch [XML] und sind daher stets XML-Instanzen. Für die Beschreibung und Prüfung von XML-Instanzen durch sogenannte *Schemata* stehen verschiedene Sprachmittel zur Verfügung, von denen am häufigsten *XML Schema Definition – XSD* – [XSD] zum Einsatz kommt. XSD wird von praktisch allen gängigen kommerziellen und nicht-kommerziellen XML-Entwicklungswerkzeugen unterstützt. Aus diesem Grund wurden auch die eXtra Standardnachrichten mit XSD Schema definiert.

eXtra vermeidet die Verwendung von XSD-Sprachmitteln, die XML-Instanzen an eine Validierung mittels bestimmter XSD-Instanzen binden, wie zum Beispiel die Typsubstitution. Dies eröffnet Anwendern die Möglichkeit, eigene Schemata zu verwenden, solange diese die Schnittstellenspezifikation unter Beachtung der Profilierung korrekt implementieren. Eine Beschränkung auf XSD als Sprachmittel besteht dabei nicht. Das Schema ist in diesem Sinne als ein Bestandteil Teil der Schnittstellenimplementierung zu betrachten, welche vollständig nach außen gekapselt ist. Jedes Schema, das die Schnittstellenspezifikation korrekt implementiert, muss jede Nachrichteninstanz, die alle Regeln der Schnittstellenspezifikation erfüllt, als valide erkennen.

Diese Spezifikation beschreibt Elemente, Attribute und Daten weitgehend neutral in Bezug auf die möglichen Sprachmittel für die Erstellung von Schemadefinitionen. Lediglich für die Beschreibung primitiver Datentypen wie Zeichenketten, numerischen Werten und Datumsangaben werden XSD Sprachmittel verwendet, da diese gut verständlich sind und auch in anderen Schemasprachen wie z.B. Relax NG [RELAX] Anwendung finden. In den graphischen Darstellungen der Elemente können schemaspezifische oder XSD-spezifische Konstrukte wie etwa Namen von Typdefinitionen erscheinen. Dies ist eine Folge der Generierung der Graphiken aus den eXtra XSD-Instanzen, hat aber keine normative Wirkung.

Der normative Spezifikationsteil beginnt mit den einleitenden Kapiteln, Definitionen und Aufbau von eXtra Standardnachrichten. Anschließend folgt die Spezifikation den Namensräumen [XNAM] der Elemente. Jeder Namensraum ist in einem separaten Kapitel beschrieben. Soweit in einem Namensraum ein Wurzelement existiert, ist dieses als erstes beschrieben. Ansonsten erscheinen die Elemente in alphabetischer Reihenfolge.

Die Kapitel des normativen Spezifikationsteiles sind mit **[normativ]** gekennzeichnet.

Layout und Notationen

Formatierungen

Es werden folgende Formatierungen verwendet:

- *Arial 11pt* für den laufenden Text einschließlich bibliographischer Referenzen.
- *Arial 11pt kursiv* für die erstmalige Erwähnung wichtiger Begriffe, sowie für fremdsprachliche Ausdrücke, soweit diese hervorgehoben werden sollen.
- *Courier New 11pt* für XML-Komponenten wie Elemente und Attribute sowie für konstante bzw. vordefinierte Werte.
- *Courier New 11pt* für kleinere Beispielfragmente aus XML-Instanzen.
- *Courier New 8pt* für größere Beispielfragmente aus XML-Instanzen.

In Überschriften können Fontgewicht und Fontgröße von diesen Angaben abweichen.

Beschreibung der Regeln

Jede Regel des eXtra Regelwerkes trägt eine eindeutige Regelbezeichnung und ist einem Regelbereich zugeordnet, der durch einen Buchstaben identifiziert wird:

Buchstabenkürzel	Regelbereich
I	<i>Implementierung</i>
P	<i>Profilierung</i>
X	<i>XML Schnittstelle</i>

In einem engen Zusammenhang stehende Regeln können zu einer Regelgruppe zusammengefasst sein. Regeln und Regelgruppen sind innerhalb eines Regelbereiches mit 1 beginnend durchnummeriert. Innerhalb einer Regelgruppe sind wiederum die Regeln durchnummeriert. Hieraus ergibt sich folgender Aufbau der Regelbezeichnung:

[*rb rn*] bzw.
[*rb rg.rn*]

wobei *rb* der Regelbereich, *rg* die Regelgruppe und *rn* die Regelnummer angibt. Beispiele:

[I 12], [P 8.3].

Beschreibung der Elemente und Attribute

Die Beschreibung eines Elementes und seiner Attribute setzt sich zusammen aus:

- Einer Tabelle „Element“, i.d.R. mit grafischer Darstellung,

- je einer Tabelle „Attribut“ pro Attribut,
- einer Tabelle „Profilierung“, falls vorhanden,
- einer textuellen Beschreibung.

Tabelle „Element“

	<i>präfigierter_elementname</i>
<i>Inhalt</i>	<i>kurzbeschreibung_element</i>
<i>Datentyp</i>	<i>inhaltstyp_element</i>
<i>Benutzt von</i>	<i>präfigierte_namen_elternelemente</i>
	<i>element_graphik</i>

Im Einzelnen:

- *Inhalt*: Eine kurze Beschreibung des Elementinhaltes.
- *Datentyp*: Beschreibt den Inhalt des Elementes in typisierender Weise oder durch Angabe eines XSD-Datentyps (Attribute werden in einer separaten Tabelle beschrieben);
 - `##elements`: Das Inhaltsmodell des Elementes enthält mindestens ein weiteres Element und ggf. ein oder mehrere Attribute
 - `##empty`: Das Inhaltsmodell des Elementes sieht keinen Inhalt, aber ggf. ein oder mehrere Attribute vor
 - Wenn das Inhaltsmodell nur Daten als Inhalt vorsieht, wird der entsprechende *built-in* XSD-Datentyp angegeben.
- *Benutzt von*: Die präfigierten Namen der Elemente, in denen das beschriebene Element verwendet wird, z.B. `xreq:MessageHeader`, `xres:MessageHeader`.

Tabelle „Attribut“

	<i>attributname</i>	<i>optionalität</i>
<i>Inhalt</i>	<i>kurzbeschreibung_attribut</i>	
<i>Datentyp</i>	<i>xsd-datentyp</i>	

Im Einzelnen:

- *attributname* : Der Name des Attributes in XPath-Notation [XPATH], d.h. mit vorausgehendem @-Symbol
- *optionalität*: Eines der Adjektive „erforderlich“ oder „optional“
- *Inhalt*: Eine kurze textuelle Beschreibung des Attributes.
- *Datentyp*: Ein einfacher XSD-Datentyp.

Tabelle „Profilierung“

<i>Komponente</i>	<i>REQ</i>	<i>OPT</i>	<i>FOR</i>	<i>MUL</i>	<i>IGN</i>	<i>VAL</i>
xcpt:Signatures	+	+	+		+	
xcpt:Data				<i>IMPLICIT</i>		

Diese Tabelle beschreibt für jede profilierbare Komponente des Elementes die Profilierungsoptionen, d.h. wie die Profilierung die Kardinalität und/oder den Inhalt einer Komponente beeinflussen kann. Die Profilierungsoptionen sind als Folge von Spalten dargestellt; ein + in der durch eine Kombination von Komponente und Option bestimmten Zelle kennzeichnet die Option als zulässig. Bestimmte Komponenten werden implizit profiliert, d.h. ihre Profilierung leitet sich aus der Profilierung einer anderen Komponente, einer Funktion oder einer Struktur ab. In diesem Fall sind die Profilierungsspalten zusammengefasst und enthalten das Wort *IMPLICIT*.

Die Bezeichner bedeuten:

- *REQ*: Profilierung kann die Komponente als erforderlich (required) deklarieren
- *OPT*: Profilierung kann die Komponente als optional (optional) deklarieren
- *FOR*: Profilierung kann die Komponente als verboten (forbidden) deklarieren
- *MUL*: Profilierung kann die Komponente als m bis n mal vorkommend deklarieren
- *IGN*: Profilierung kann die Komponente als ignoriert (ignored) deklarieren
- *VAL*: Der Wert (value) ist profilierbar (nur Elemente mit Dateninhalt und Attribute)

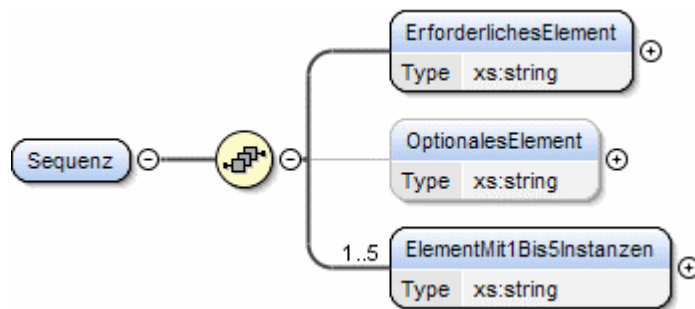
Graphische Symbole

Die Strukturen und Inhalte der eXtra Elemente werden mittels der unten beschriebenen generierten graphischen Symbole dargestellt.

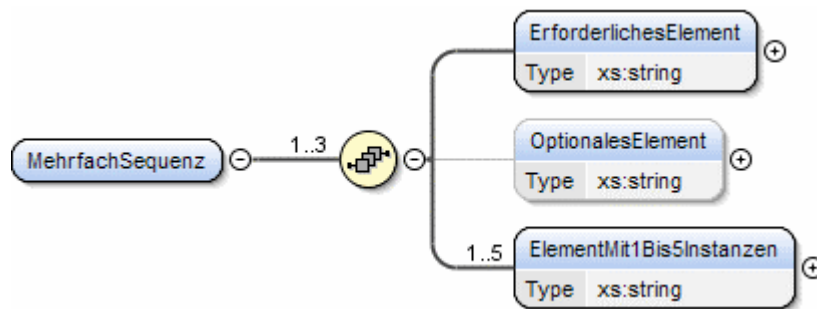
Hinweis: In den meisten Fällen erscheinen Elemente und Attribute in einem im Farbruck gelb eingefärbten Rechteck. Dieses stellt eine XSD-Typdefinition dar und trägt deren Namen. Typdefinitionen sind schemaspezifisch und daher nicht normativ. Aus diesem Grund findet sich in den folgenden Mustergraphiken auch kein Beispiel für eine Typdefinition.

Hinweis: Elemente, Sequenzen und Auswahlsequenzen, die nicht fehlen dürfen, sind mit der sie beinhaltenden Komponente mittels einer **fett** dargestellten Linie verbunden. Die Symbole von Pflichtelementen und -attributen haben einen **fetten** Rand.

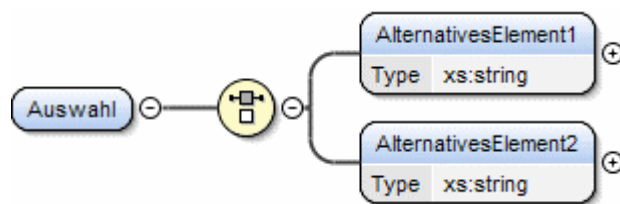
Eine Sequenz von Elementen



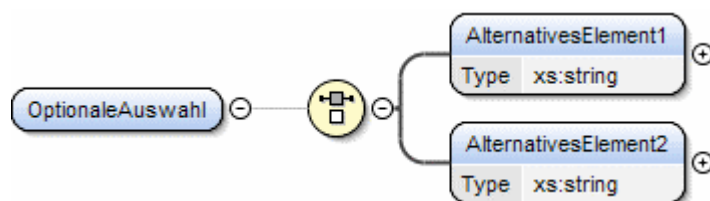
Eine bis zu 3 mal wiederholte Sequenz von Elementen



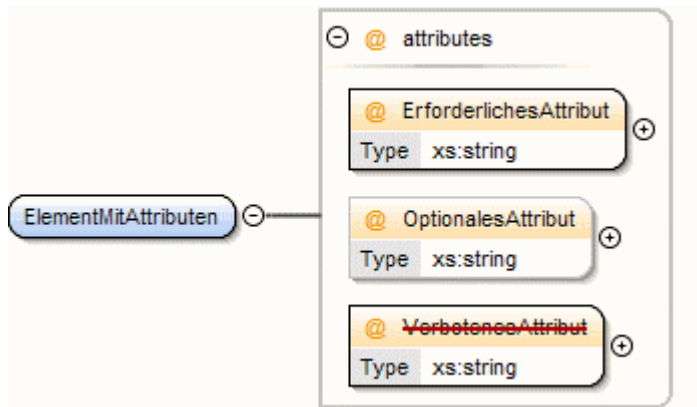
Eine erforderliche Auswahl (Kindelemente sind Pflichtelemente)



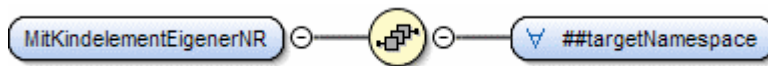
Eine optionale Auswahl (Kindelemente sind Pflichtelemente)



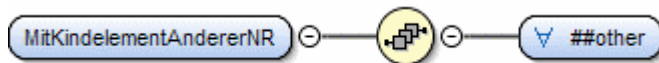
Ein Element mit Attributen



Ein Element mit einem beliebigem Kindelement aus dem eigenen Namensraum



Ein Element mit einem beliebigem Kindelement aus einem anderen Namensraum



Definitionen [normativ]

Begriffe

In Anlehnung an [RFC2119] verwendet diese Spezifikation die unten aufgeführten Modalverben und Adjektive wie dargestellt in durchgängiger Großschreibung und mit der beschriebenen Bedeutung, wenn sie an der Verwendungsstelle eine normative Wirkung entfalten. Sie können innerhalb der mit **[normativ]** gekennzeichneten Kapitel in allen konjugierten bzw. deklinierten Formen erscheinen.

- MUSS / ERFORDERLICH : Die Anforderung ist zwingend und ohne Ausnahme zu erfüllen
- DARF NICHT / VERBOTEN : Das Verbot ist zwingend und ohne Ausnahme zu einzuhalten
- SOLL / EMPFOHLEN : Die Anforderung ist zu erfüllen, wenn nicht besondere technische oder fachliche Umstände eine Ausnahme begründen
- SOLL NICHT / NICHT EMPFOHLEN : Das Verbot ist einzuhalten, wenn nicht besondere technische oder fachliche Umstände eine Ausnahme begründen
- KANN / OPTIONAL : Die Anforderung ist vollständig optional und ihre Erfüllung daher in das Belieben des Implementierenden gestellt; Eine Anwendung, die mit einer anderen Implementierung kommuniziert, MUSS sich darauf einstellen, dass diese nicht die betroffene Anforderung erfüllt.

Werden diese Begriffe in Kleinschreibung verwendet, sind die mit ihnen beschriebenen Eigenschaften oder Vorgänge entweder nicht Teil einer Regel, oder sie wiederholen an anderer Stelle des eXtra Regelwerkes definierte Normen. Sie haben im zweiten Fall trotzdem keine normative Wirkung, da jede Regel innerhalb des eXtra Regelwerkes nur an einer Stelle als Norm definiert ist.

Namensräume und Präfixe

Für eXtra Standardnachrichten finden die nachfolgend aufgeführten Namensräume und Präfixe Anwendung. Die Verwendung einheitlicher Präfixe für einen Namensraum ist eine XML Best Practice, und es wird daher EMPFOHLEN, nur die genannten Präfixe zu verwenden.

Namensraum

<http://www.extra-standard.de/namespace/message/1>

Präfix

xmsg

Zur Unterscheidung einzelner Standardnachrichten können alternativ die folgenden Präfixe verwendet werden (in [] stehende Werte sind obsolet und SOLLEN NICHT mehr angewendet werden):

Standardnachricht	Präfix
<i>ConfirmationOfReceipt</i>	xmcor [xcor]
<i>DataRequest</i>	xmdrq [xdrq]
<i>ListConfirmationOfReceipt</i>	xmcor [xcor]
<i>ListOfDataRequest</i>	xmdrq [xdrq]
<i>ListOfStatusResponse</i>	xmsrp
<i>StatusRequest</i>	xmsrq
<i>StatusResponse</i>	xmsrp

eXtra Codelisten [normativ]

Die im Folgenden beschriebenen Codelisten sind im Schema-Modul `eXtra-messages-1.xsd` definiert.

Codelisten, die auf dem Datentyp `xsd:anyURI` basieren, haben in der Regel vordefinierte Werte aus der Domain `http://www.extra-standard.de`. In den meisten Fällen können diese verfahrensspezifisch erweitert werden. Dieser Vorgang wird als *Wertindividualisierung* bezeichnet. Es vordefinierter Wert wird dabei durch einen *Individualisierungsausdruck* erweitert. Die Syntax eines Individualisierungsausdrucks ist abgeleitet von der Syntax sogenannter *Anker*, wie sie in URLs (die syntaktisch mit URIs identisch sind) der Adressierung von Dokumentfragmenten dienen.

Ein Individualisierungsausdruck beginnt mit einem ‚#‘ Zeichen. Diesem folgt eine Sequenz beliebig vieler Ziffern und/oder Buchstaben des englischen Alphabetes in Groß- oder Kleinschreibung. Diese Sequenz kann durch einzelne Minuszeichen und Punkte unterteilt werden. Vor und nach einem Minuszeichen oder Punkt MUSS eine Ziffer oder ein Buchstabe stehen. Individualisierungsausdrücke werden mit dem regulären Ausdruck

`(#[a-zA-Z0-9]([".\-"][a-zA-Z0-9]+)*)?`

geprüft Beispiel:

Gültige Ausdrücke		Ungültige Ausdrücke	
#0	Nur Ziffern erlaubt	0	‚#‘ fehlt
#noSig	Nur Buchstaben erlaubt	#no/Sig	Unzulässiges Zeichen ‚/‘
#A-1.B	Gültige Unterteilung	#.A--1	Ungültige Unterteilung

Einige Codelisten lassen zudem die Angabe von URIs beliebiger Domains zu.

Codeliste DataRequestPropertyNames

xmsg:DataRequestPropertyNamesType	
<i>Inhalt</i>	Menge der per <i>Data Request</i> abfragbaren Eigenschaften
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Ja

Vordefinierte Werte:

<http://www.extra-standard.de/property/SenderID>
<http://www.extra-standard.de/property/ReceiverID>
<http://www.extra-standard.de/property/Procedure>
<http://www.extra-standard.de/property/DataType>
<http://www.extra-standard.de/property/ResponseID>
<http://www.extra-standard.de/property/ResponseCreationTimeStamp>
<http://www.extra-standard.de/property/ResponseFileName>
<http://www.extra-standard.de/property/Layer>

Beschreibung

Die Codeliste definiert die Menge der per *Data Request* abfragbaren Eigenschaften und MUSS ein Subset der Werte der Codeliste `xmsg:PropertyNamesType` darstellen.

Codeliste EventNames

xmsg:EventNamesType	
<i>Inhalt</i>	Menge der abfragbaren Ereignisse
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Ja

Vordefinierte Werte:

<http://www.extra-standard.de/event/Default>
<http://www.extra-standard.de/event/SendData>
<http://www.extra-standard.de/event/RequestData>

Beschreibung

Die Codeliste enthält URIs, die typische Ereignisse im Ablauf eines eXtra-basierten Kommunikationsszenarios darstellen.

Codeliste LayerNames

xmsg: ExtraLayerNamesType	
<i>Inhalt</i>	Ebenen der eXtra Kommunikationsarchitektur
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Nein

Vordefinierte Werte:

<http://www.extra-standard.de/layer/Transport>
<http://www.extra-standard.de/layer/Package>
<http://www.extra-standard.de/layer/Message>
<http://www.extra-standard.de/layer/Application>
<http://www.extra-standard.de/layer/Delivery>

Beschreibung

Die Codeliste enumeriert die Ebenen der eXtra Kommunikationsarchitektur.

Codeliste PropertyNames

xmsg: PropertyNamesType	
<i>Inhalt</i>	Menge allgemein abfragbarer Eigenschaften
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Ja

Vordefinierte Werte:

<http://www.extra-standard.de/property/SenderID>
<http://www.extra-standard.de/property/ReceiverID>
<http://www.extra-standard.de/property/Procedure>
<http://www.extra-standard.de/property/DataType>
<http://www.extra-standard.de/property/RequestID>
<http://www.extra-standard.de/property/RequestCreationTimeStamp>
<http://www.extra-standard.de/property/RequestFileName>
<http://www.extra-standard.de/property/ResponseID>
<http://www.extra-standard.de/property/ResponseCreationTimeStamp>
<http://www.extra-standard.de/property/ResponseFileName>
<http://www.extra-standard.de/property/Layer>

Beschreibung

Die Codeliste enthält die Obermenge aller URIs, die abfragbare Eigenschaften von eXtra Nachrichten repräsentieren.

Codeliste StatusNames

	xmsg:StatusNamesType
<i>Inhalt</i>	Stati eines eXtra Request
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Ja

Vordefinierte Werte:

`http://www.extra-standard.de/status/ACCEPTED`
`http://www.extra-standard.de/status/PROCESSING`
`http://www.extra-standard.de/status/COMPLETED`
`http://www.extra-standard.de/status/CONFIRMED`
`http://www.extra-standard.de/status/FAILED`

Beschreibung

Die Codeliste definiert die Stati, die ein eXtra Request an spezifischen Kontrollpunkten (Checkpoints) der Prozskette haben kann.

Codeliste StatusRequestPropertyNames

	xmsg:StatusRequestPropertyNamesType
<i>Inhalt</i>	Menge der per <i>Status Request</i> abfragbaren Eigenschaften
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Ja

Vordefinierte Werte:

`http://www.extra-standard.de/property/SenderID`
`http://www.extra-standard.de/property/ReceiverID`
`http://www.extra-standard.de/property/Procedure`
`http://www.extra-standard.de/property/DataType`
`http://www.extra-standard.de/property/RequestID`
`http://www.extra-standard.de/property/RequestCreationTimeStamp`
`http://www.extra-standard.de/property/RequestFileName`
`http://www.extra-standard.de/property/ResponseID`
`http://www.extra-standard.de/property/ResponseCreationTimeStamp`

<http://www.extra-standard.de/property/ResponseFileName>
<http://www.extra-standard.de/property/Layer>

Beschreibung

Die Codeliste definiert die Menge der per *Status Request* abfragbaren Eigenschaften und MUSS ein Subset der Werte der Codeliste `xmsg:PropertyNamesType` darstellen.

Codeliste *WeightCodes*

	xmsg:WeightCodeType
<i>Inhalt</i>	Fehlgewichte
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Ja
<i>Beliebige Domain</i>	Nein

Vordefinierte Werte:

<http://www.extra-standard.de/weight/OK>
<http://www.extra-standard.de/weight/INFO>
<http://www.extra-standard.de/weight/WARNING>
<http://www.extra-standard.de/weight/ERROR>

Beschreibung

Die Codeliste enumeriert die Fehlgewichte für Flags. Die Werte sind individualisierbar.

Codeliste *XSDTypeCodes*

	xmsg:XSDPrefixedTypeCodes
<i>Inhalt</i>	XSD Typnamen
<i>Datentyp</i>	xsd:QName
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Nein

Vordefinierte Werte:

<!-- Strings and names -->

```
<xs:enumeration value="xs:string" />
<xs:enumeration value="xs:normalizedString" />
<xs:enumeration value="xs:token" />
<xs:enumeration value="xs:Name" />
<xs:enumeration value="xs:NCName" />
<xs:enumeration value="xs:QName" />
<xs:enumeration value="xs:language" />
```

<!-- Numeric -->

```
<xs:enumeration value="xs:float" />
<xs:enumeration value="xs:double" />
<xs:enumeration value="xs:decimal" />
<xs:enumeration value="xs:integer" />
<xs:enumeration value="xs:long" />
<xs:enumeration value="xs:int" />
<xs:enumeration value="xs:short" />
<xs:enumeration value="xs:byte" />
<xs:enumeration value="xs:positiveInteger" />
<xs:enumeration value="xs:nonPositiveInteger" />
<xs:enumeration value="xs:negativeInteger" />
<xs:enumeration value="xs:nonNegativeInteger" />
<xs:enumeration value="xs:unsignedLong" />
<xs:enumeration value="xs:unsignedInt" />
<xs:enumeration value="xs:unsignedShort" />
<xs:enumeration value="xs:unsignedByte" />
```

<!-- Date and time -->

```
<xs:enumeration value="xs:duration" />
<xs:enumeration value="xs:dateTime" />
<xs:enumeration value="xs:date" />
<xs:enumeration value="xs:time" />
<xs:enumeration value="xs:gYear" />
<xs:enumeration value="xs:gMonth" />
<xs:enumeration value="xs:gYearMonth" />
<xs:enumeration value="xs:gMonthDay" />
<xs:enumeration value="xs:gDay" />
```

<!-- Legacy types -->

```
<xs:enumeration value="xs:ID" />
<xs:enumeration value="xs:IDREF" />
<xs:enumeration value="xs:IDREFS" />
<xs:enumeration value="xs:ENTITY" />
<xs:enumeration value="xs:ENTITIES" />
<xs:enumeration value="xs:NMTOKEN" />
<xs:enumeration value="xs:NMTOKENS" />
<xs:enumeration value="xs:NOTATION" />
```

<!-- Other -->

```
<xs:enumeration value="xs:boolean" />
<xs:enumeration value="xs:hexBinary" />
```

```
<xs:enumeration value="xs:base64Binary" />  
<xs:enumeration value="xs:anyURI" />
```

```
<!-- Undefined -->
```

```
<xs:enumeration value="xmsg:UNDEFINED" />
```

Beschreibung

Die Codeliste enumeriert XSD Schema Typreferenzen. Sie ist weder individualisierbar noch erweiterbar.

Nachrichtentyp ConfirmationOfReceipt [normativ]

Eine Nachricht vom Typ *ConfirmationOfReceipt* ist eine Empfangsbestätigung für eine Menge von Anwendungsnachrichten.

Der Name des Wurzelementes ist `ConfirmationOfReceipt`.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

Beispieldokument

Der folgende *ConfirmationOfReceipt* bestätigt den Empfang eines Dokumentes mit den angegebenen Eigenschaften des eXtra Kommunikationsprotokolls.

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:ConfirmationOfReceipt
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.3"
  >

  <xmsg:Property name="http://www.extra-standard.de/property/SenderID" type="xs:string">
    <xmsg:Value>77700000</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property name="http://www.extra-standard.de/property/ReceiverID" type="xs:string">
    <xmsg:Value>55550000</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property name="http://www.extra-standard.de/property/Procedure" type="xs:anyURI">
    <xmsg:Value>http://www.extra-standard.de/procedures/DEUEV</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property name="http://www.extra-standard.de/property/DataType" type="xs:anyURI">
    <xmsg:Value>http://www.extra-standard.de/datatypes/RSVNr</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property
    name="http://www.extra-standard.de/property/ResponseCreationTimeStamp"
    type="xs:dateTime">
    <xmsg:Value>2006-05-04T18:13:51.0Z</xmsg:Value>
  </xmsg:Property>

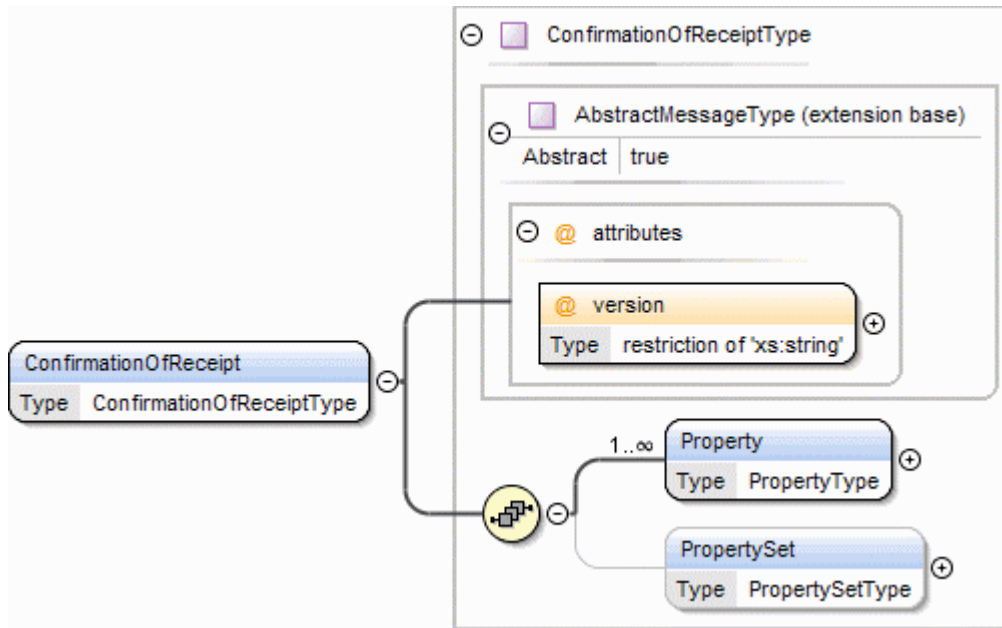
  <xmsg:PropertySet name="http://www.extra-standard.de/property/ResponseID"
    type="xs:string" event="http://www.extra-standard.de/event/RequestData">
    <xmsg:Value>res20090223152540-00008060</xmsg:Value>
    <xmsg:Value>res20090223152540-00008061</xmsg:Value>
    <xmsg:Value>res20090223152540-00008062</xmsg:Value>
    <xmsg:Value>res20090223152540-00008063</xmsg:Value>
    <xmsg:Value>res20090223152540-00008064</xmsg:Value>
  </xmsg:PropertySet>

</xmsg:ConfirmationOfReceipt>
```

Element ConfirmationOfReceipt

xmsg:ConfirmationOfReceipt

<i>Inhalt</i>	Beschreibt den Empfang einer Nachricht oder eine Menge von Nachrichten.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.3
<i>Datentyp</i>	xsd:string

Beschreibung

`xmsg:ConfirmationOfReceipt` ist das Wurzelement einer eXtra Nachricht vom Typ `ConfirmationOfReceipt`. Das Attribut `version` enthält die Versionsbezeichnung.

Eine Nachricht dieses Typs dient zur Bestätigung des Empfangs einzelner oder Mengen von *Anwendungsnachrichten*. Eine Anwendungsnachricht ist eine Nachricht in Sinne einer fachlichen Anwendung, die mittels des eXtra Kommunikationsprotokolls übertragen wurde. Die bestätigten Anwendungsnachrichten werden mittels der Elemente `Property` und `PropertySet` anhand einer Menge von Eigenschaften beschrieben. Diese Eigenschaften sollen jeweils eine individuelle Anwendungsnachricht identifizieren, d.h. die Kombination der Werte dieser Eigenschaften soll eindeutig sein. Für eine ausgewählte Eigenschaft erlaubt das Element

`PropertySet`, mehr als einen Wert anzugeben und damit eine Menge von Anwendungsnachrichten zu identifizieren.

Enthält die Nachricht `ConfirmationOfReceipt` nur `Property`-Elemente, so wird mit dieser Folge genau eine Anwendungsnachricht identifiziert. Die `Property`-Elemente werden somit wie ein logisches UND miteinander verknüpft.

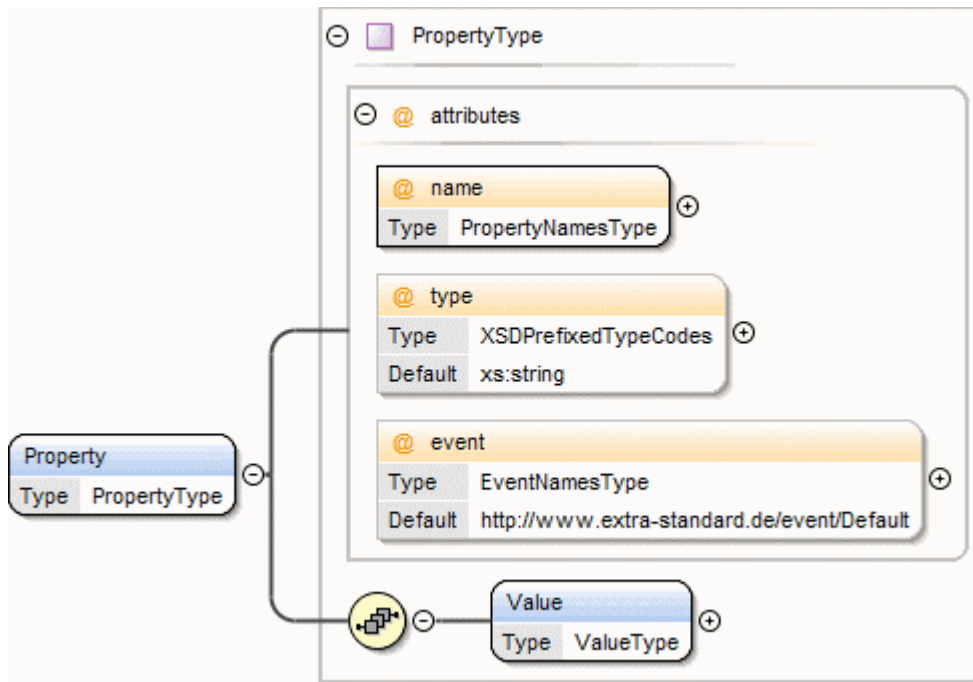
Enthält die Nachricht `ConfirmationOfReceipt` mehrere `Property`-Elemente und ein `PropertySet`-Element, so müssen alle Anwendungsnachrichten, die im `PropertySet`-Element aufgeführt werden allen Eigenschaften genügen, die in den `Property`-Elementen spezifiziert wurden. Die Folge von `Property`-Elementen wirken wie ein logisches UND, während die Werte im `PropertySet`-Element als logisches ODER zu verstehen sind.

Eigenschaften werden mittels einer URI identifiziert. Für Eigenschaften, die im eXtra Kommunikationsprotokoll abgebildet werden, stehen vordefinierte URIs zur Verfügung.

Element Property

xmsg:Property

<i>Inhalt</i>	Liefert für eine Eigenschaft genau einen Wert.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:ConfirmationOfReceipt



Attribute

	@event	<i>optional</i>
<i>Inhalt</i>	Bezieht die abgefragte Eigenschaft auf ein Ereignis im Kommunikationsablauf	
<i>Datentyp</i>	xsd:anyURI	
<i>Werteraum</i>	##codelist:EventNames	
<i>Default</i>	http://www.extra-standard.de/event/Default	
	@name	<i>erforderlich</i>
<i>Inhalt</i>	Der Name der Eigenschaft	
<i>Werteraum</i>	##codelist:PropertyNames	
<i>Datentyp</i>	xsd:anyURI	
	@type	<i>optional</i>
<i>Inhalt</i>	Der Datentyp der Eigenschaft	
<i>Datentyp</i>	xsd:string	
<i>Werteraum</i>	##codelist:XSDTypeCodes	
<i>Default</i>	xsd:string	

Profilierung

Komponente	REQ	OPT	FOR	MUL	IGN	VAL
@event						+

Beschreibung

`xmsg:Property` beschreibt eine Eigenschaft und liefert genau einen Wert dieser Eigenschaft.

Mit dem Attribut `event` wird das Ereignis im Kommunikationsablauf spezifiziert, für das der Wert der Eigenschaft gültig ist. Die vordefinierte Werteliste kann durch nutzerspezifische URIs ersetzt bzw. ergänzt werden.

Beispiel

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:ConfirmationOfReceipt
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.3"
  >
  <xmsg:Property name="http://www.extra-
standard.de/property/SenderID"
  type="xs:string">
    <xmsg:Value>77700000</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property name="http://www.extra-
standard.de/property/ReceiverID"
  type="xs:string">
    <xmsg:Value>55550000</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property name="http://www.extra-
standard.de/property/Procedure"
  type="xs:anyURI">
    <xmsg:Value
      >http://www.extra-standard.de/procedures/DEUEV<
```

```
    /xmsg:Value>  
  </xmsg:Property>
```

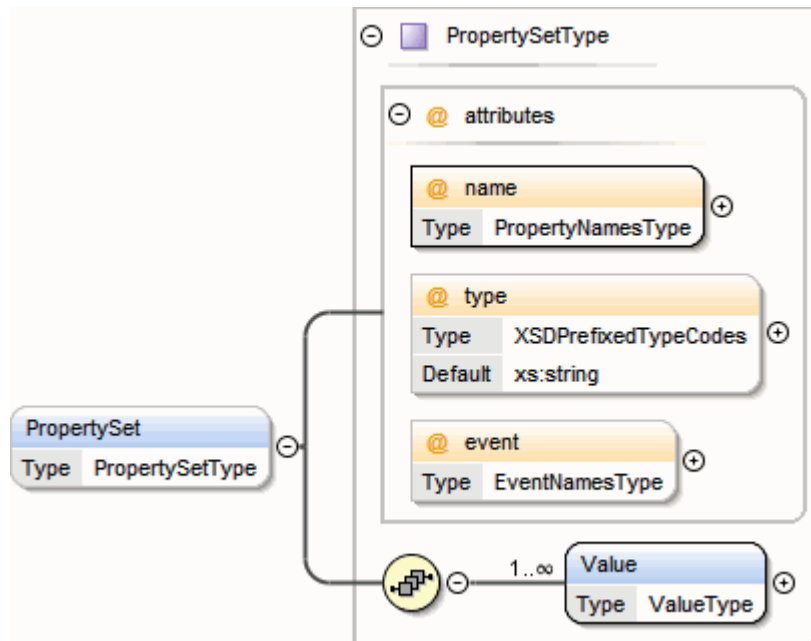
```
    <xmsg:Property name="http://www.extra-  
standard.de/property/ResponseID"  
event="http://www.extra-standard.de/event/RequestData"  
    <xmsg:Value>00008060</xmsg:Value>  
  </xmsg:Property>
```

```
</xmsg:ConfirmationOfReceipt>
```

Element PropertySet

xmsg:PropertySet

<i>Inhalt</i>	Liefert für eine Eigenschaft eine Menge von ein oder mehr Werten.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:ConfirmationOfReceipt



Attribute

	@event	<i>optional</i>
<i>Inhalt</i>	Bezieht die abgefragte Eigenschaft auf ein Ereignis im Kommunikationsablauf	
<i>Datentyp</i>	xsd:anyURI	
<i>Werteraum</i>	##codelist:EventNames	
<i>Default</i>	http://www.extra-standard.de/event/Default	
	@name	<i>erforderlich</i>
<i>Inhalt</i>	Der Name der Eigenschaft	
<i>Werteraum</i>	##codelist:PropertyNames	
<i>Datentyp</i>	xsd:anyURI	
	@type	<i>optional</i>
<i>Inhalt</i>	Der Datentyp der Eigenschaft	
<i>Werteraum</i>	##codelist:XSDTypeCodes	
<i>Datentyp</i>	xsd:string	

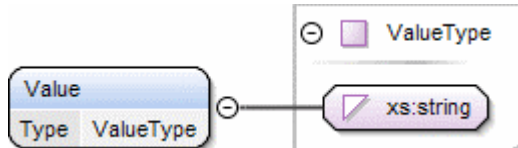
Beschreibung

`xmsg:PropertySet` beschreibt eine Eigenschaft und liefert eine beliebig große Menge von Werten dieser Eigenschaft.

Element Value

xmsg:Value

<i>Inhalt</i>	Enthält den Wert einer Eigenschaft.
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Property, xmsg:PropertySet



Beschreibung

Enthält den Wert einer Eigenschaft. Der Datentyp wird durch das Attribut `type` des übergeordneten `Property` oder `PropertySet` Elementes festgelegt.

Nachrichtentyp DataRequest [normativ]

Mit Nachrichten vom Typ *DataRequest* ist es möglich, Anwendungsnachrichten oder Daten anhand bestimmter Kriterien abzurufen.

Der Name des Wurzelementes ist `DataRequest`.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

Beispieldokument

Der folgende *DataRequest* fragt eine Reihe von Eigenschaften des eXtra Kommunikationsprotokolls ab und begrenzt die Größe der Antwort.

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:DataRequest
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.2"
  >

  <xmsg:Query>

    <xmsg:Argument property="http://www.extra-standard.de/property/SenderID"
      type="xs:string">
      <xmsg:EQ>5550000</xmsg:EQ>
    </xmsg:Argument>

    <xmsg:Argument property="http://www.extra-standard.de/property/ReceiverID"
      type="xs:string">
      <xmsg:EQ>7770000</xmsg:EQ>
    </xmsg:Argument>

    <xmsg:Argument property="http://www.extra-standard.de/property/Procedure"
      type="xs:anyURI">
      <xmsg:EQ>http://my.procedure.com/procl</xmsg:EQ>
    </xmsg:Argument>

    <xmsg:Argument property="http://www.extra-standard.de/property/DataType"
      type="xs:anyURI">
      <xmsg:EQ>http://my.datatype.com/type1</xmsg:EQ>
    </xmsg:Argument>

    <xmsg:Argument property="http://www.extra-standard.de/property/ResponseID"
      type="xs:string"
      event="http://www.extra-standard.de/event/RequestData">
      <xmsg:IN>
        <xmsg:EQ>R46934503</xmsg:EQ>
        <xmsg:EQ>R46934504</xmsg:EQ>
        <xmsg:EQ>R46934508</xmsg:EQ>
      </xmsg:IN>
    </xmsg:Argument>

    <xmsg:Argument property="http://www.extra-standard.de/property/ResponseFileName"
      type="xs:string"
      event="http://www.extra-standard.de/event/RequestData">
```

```
<xmsg:EQ>R46934503</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument
  property="http://www.extra-standard.de/property/ResponseCreationTimeStamp"
  type="xs:dateTime">
  <xmsg:GT>2009-03-16T09:00:00</xmsg:GT>
  <xmsg:LE>2009-03-17T09:00:00</xmsg:LE>
</xmsg:Argument>

</xmsg:Query>

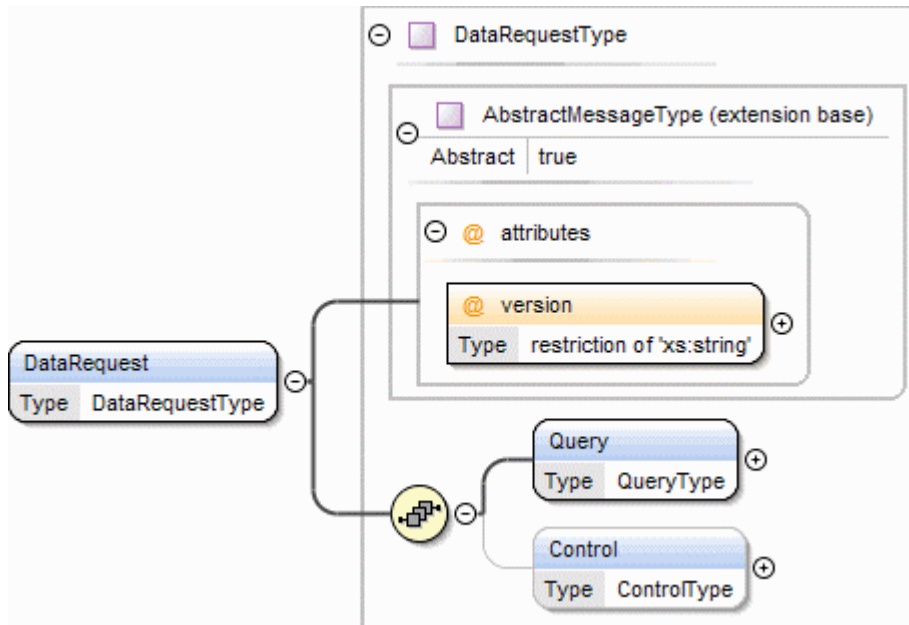
<xmsg:Control>
  <xmsg:MaximumPackages>10</xmsg:MaximumPackages>
  <xmsg:MaximumMessages>100</xmsg:MaximumMessages>
  <xmsg:MaximumSize>10M</xmsg:MaximumSize>
</xmsg:Control>

</xmsg:DataRequest>
```

Element *DataRequest*

xmsg:DataRequest

<i>Inhalt</i>	Beschreibt eine Anforderung von Anwendungsnachrichten oder Daten anhand bestimmter Kriterien.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.2
<i>Datentyp</i>	xsd:string

Beschreibung

Eine Nachricht dieses Typs dient zur Anforderung einzelner oder Mengen von Daten oder *Anwendungsnachrichten*. Eine Anwendungsnachricht ist eine Nachricht in Sinne einer fachlichen Anwendung, die mit Hilfe des eXtra Kommunikationsprotokolls angefordert wird. Die Kriterien der Anforderung werden mit dem Element *Query* und einer oder mehrerer Unter-elemente *Argument* beschrieben. Die Kriterien der Anforderung müssen nicht notwendigerweise genau eine individuelle Anwendungsnachricht identifizieren, sie können auch für eine Menge von bereitgestellten Anwendungsnachrichten zutreffen. Um für diesen Fall die Menge der zurück zu liefernden Anwendungsnachrichten bzw. die Größe des zurück zu liefernden eXtra-Dokumentes (genauer des Elements *TransportBody*) begrenzen zu können, gibt es das Element *Control*.

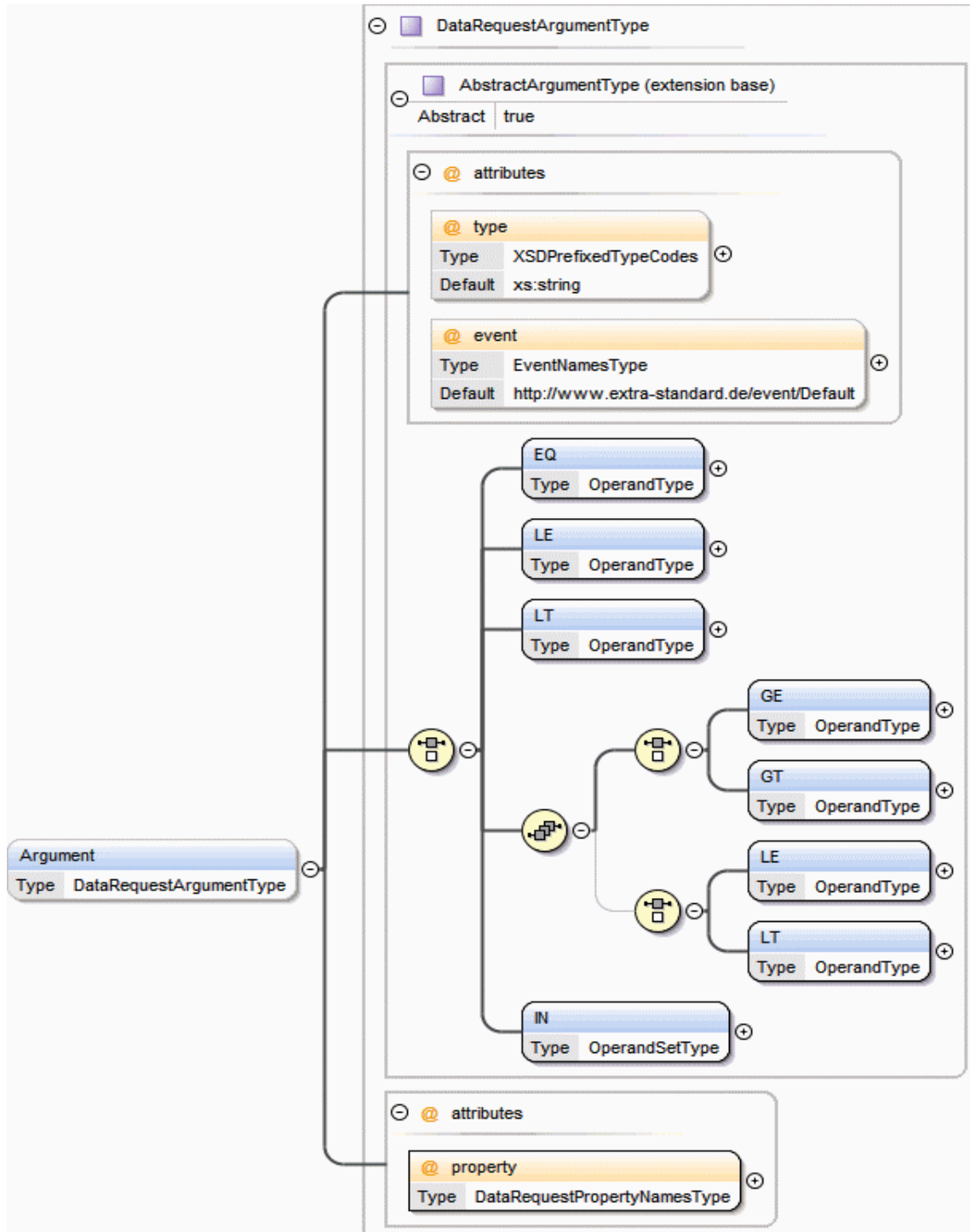
Enthält die Query mehrere *Argument*-Elemente, so müssen die angeforderten Anwendungsnachrichten bzw. Daten allen so spezifizierten Kriterien genügen. Die einzelnen Kriterien werden somit wie ein logisches UND miteinander verknüpft.

Ein Kriterium besteht aus einer abgefragten Eigenschaft (Attribut *property*), die per URI referenziert wird, und einem Wert, mit dem die entsprechende Eigenschaft der abgefragten Objekte verglichen wird. Der Wert wird als Einzelwert mit einem Kindelement oder als Wertebereich mit inklusiver Unter- und Obergrenze mit zwei Kindelementen angegeben. Ein Kindelement bestimmt zugleich den Operator für den Vergleich mit Einzelwert bzw. der Unter- oder Obergrenze.

Element Argument

xmsg:Argument

Inhalt	Beschreibt ein Kriterium einer Abfrage
Datentyp	##elements
Benutzt von	xmsg:Query



Attribute

	@event	<i>optional</i>
<i>Inhalt</i>	Bezieht die abgefragte Eigenschaft auf ein Ereignis im Kommunikationsablauf	
<i>Datentyp</i>	xsd:anyURI	
<i>Werteraum</i>	##codelist:EventNames	
<i>Default</i>	http://www.extra-standard.de/event/Default	
	@property	<i>erforderlich</i>
<i>Inhalt</i>	Bezeichnet die abgefragte Eigenschaft	
<i>Werteraum</i>	##codelist:DataRequestPropertyNamees	
<i>Datentyp</i>	xsd:anyURI	
	@type	<i>optional</i>
<i>Inhalt</i>	Der XSD Datentyp der abgefragten Eigenschaft	
<i>Datentyp</i>	xsd:string	
<i>Werteraum</i>	##codelist:XSDTypeCodes	
<i>Default</i>	xsd:string	

Profilierung

<i>Komponente</i>	<i>REQ</i>	<i>OPT</i>	<i>FOR</i>	<i>MUL</i>	<i>IGN</i>	<i>VAL</i>
@event						+

Beschreibung

Ein Kriterium besteht aus einer abgefragten Eigenschaft (Attribut `property`), die per URI referenziert wird, und einem Werteraum, mit dem die Werte der abgefragten Objekte verglichen werden. Der Werteraum wird als Einzelwert oder als Wertebereich mit inklusiver Unter- und Obergrenze oder als Werteliste angegeben. Die wertetragenden Elemente des Werteraums bestimmen zugleich den Operator für den Vergleich mit Einzelwerten bzw. der Unter- oder Obergrenze.

Das Attribut `type` spezifiziert den Datentyp des Arguments.

Mit dem Attribut `event` wird das Ereignis im Kommunikationsablauf spezifiziert, für das die Eigenschaft abgefragt werden soll. Die vordefinierte Werteliste kann durch nutzerspezifische URIs ersetzt bzw. ergänzt werden.

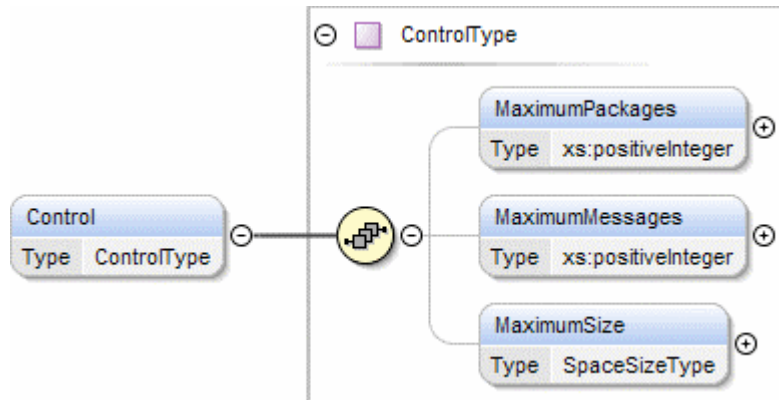
Beispiel

```
<xmsg:Argument property="http://www.extra-standard.de/property/ResponseCreationTimeStamp"  
  type="xs:dateTime">  
  <xmsg:GT>2009-03-16T09:00:00</xmsg:GT>  
  <xmsg:LE>2009-03-17T09:00:00</xmsg:LE>  
</xmsg:Argument>
```

Element Control

xmsg:Control

<i>Inhalt</i>	Enthält Angaben zur Steuerung der Erzeugung der Antwort auf die Datenanforderung.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:DataRequest



Beschreibung

Enthält Angaben zur Steuerung der Erzeugung der Antwort auf die Datenanforderung. Die Größe der Antwort kann durch mehrere optionale Parameter beschränkt werden. Ist mehr als ein Parameter angegeben, wirkt der, dessen Beschränkung sich als stärkste erweist.

Mit dem Element `MaximumSize` wird die maximale Größe der Antwort auf die Datenanforderung im Element `TransportBody` festgelegt.

Beispiel

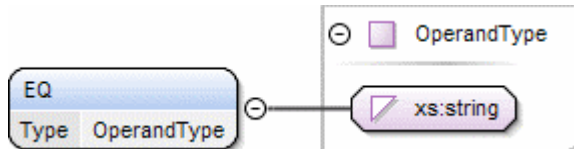
```

<xmsg:Control>
  <xmsg:MaximumMessages>100</xmsg:MaximumMessages>
</xmsg:Control>
    
```


Element EQ

xmsg:EQ

<i>Inhalt</i>	Der Wert eines Kriteriums
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Argument



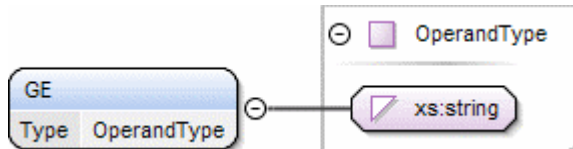
Beschreibung

Enthält den Wert eines Kriteriums. Der Vergleich soll gemäß dem in `xsd:Argument/@type` spezifizierten Typ typgerecht erfolgen. Ein Wert erfüllt den Vergleich, wenn er als GLEICH dem Wert des Kriteriums gilt.

Element GE

xmsg:GE

<i>Inhalt</i>	Der Wert eines Kriteriums
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Argument



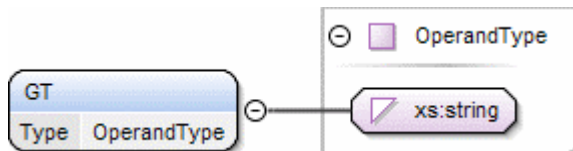
Beschreibung

Enthält den Wert eines Kriteriums. Der Vergleich soll gemäß dem in `xsd:Argument/@type` spezifizierten Typ typgerecht erfolgen. Ein Wert erfüllt den Vergleich, wenn er als GRÖSSER oder GLEICH dem Wert des Kriteriums gilt.

Element GT

xmsg:GT

<i>Inhalt</i>	Der Wert eines Kriteriums
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Argument



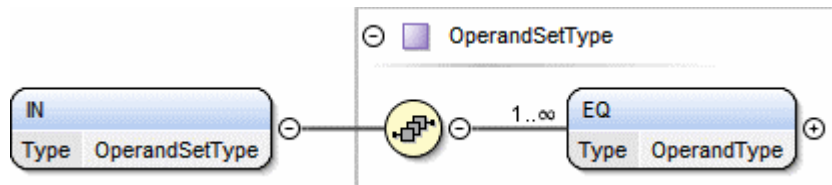
Beschreibung

Enthält den Wert eines Kriteriums. Der Vergleich soll gemäß dem in `xsd:Argument/@type` spezifizierten Typ typgerecht erfolgen. Ein Wert erfüllt den Vergleich, wenn er als GRÖSSER als der Wert des Kriteriums gilt.

Element IN

xmsg:IN

<i>Inhalt</i>	Eine Liste von Werten eines Kriteriums
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Argument



Beschreibung

Enthält Werte eines Kriteriums. Der Vergleich soll gemäß dem in `xsd:Argument/@type` spezifizierten Typ typgerecht erfolgen. Ein Wert erfüllt den Vergleich, wenn er als GLEICH einem der Werte des Kriteriums gilt.

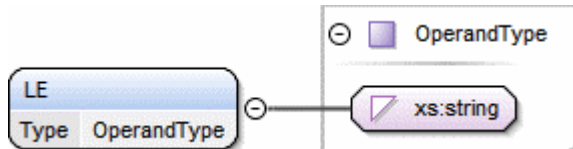
Beispiel

```
<xmsg:Argument property="http://www.extra-standard.de/property/ResponseID"
  type="xs:string"
  event="http://www.extra-standard.de/event/RequestData">
  <xmsg:IN>
    <xmsg:EQ>R46934503</xmsg:EQ>
    <xmsg:EQ>R46934504</xmsg:EQ>
    <xmsg:EQ>R46934508</xmsg:EQ>
  </xmsg:IN>
</xmsg:Argument>
```

Element LE

xmsg:LE

<i>Inhalt</i>	Der Wert eines Kriteriums
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Argument



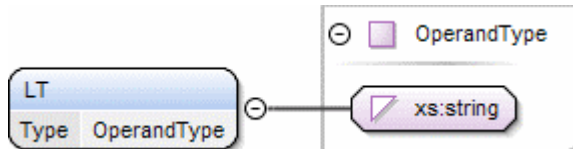
Beschreibung

Enthält den Wert eines Kriteriums. Der Vergleich soll gemäß dem in `xsd:Argument/@type` spezifizierten Typ typgerecht erfolgen. Ein Wert erfüllt den Vergleich, wenn er als KLEINER oder GLEICH dem Wert des Kriteriums gilt.

Element LT

xmsg:LT

<i>Inhalt</i>	Der Wert eines Kriteriums
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Argument



Beschreibung

Enthält den Wert eines Kriteriums. Der Vergleich soll gemäß dem in `xsd:Argument/@type` spezifizierten Typ typgerecht erfolgen. Ein Wert erfüllt den Vergleich, wenn als KLEINER als der Wert des Kriteriums gilt.

Element *MaximumMessages*

xmsg:MaximumMessages

<i>Inhalt</i>	Eine positive Ganzzahl
<i>Datentyp</i>	xsd:positiveInteger
<i>Benutzt von</i>	xmsg:Control



Beschreibung

xmsg:MaximumMessages gibt an, wieviele Anwendungsnachrichten maximal übertragen werden sollen, um die Anforderung zu befriedigen.

Beispiel

```
<xmsg:Control>
  <xmsg:MaximumMessages>100</xmsg:MaximumMessages>
</xmsg:Control>
```

Element *MaximumPackages*

xmsg:MaximumMessages

<i>Inhalt</i>	Eine positive Ganzzahl
<i>Datentyp</i>	xsd:positiveInteger
<i>Benutzt von</i>	xmsg:Control



Beschreibung

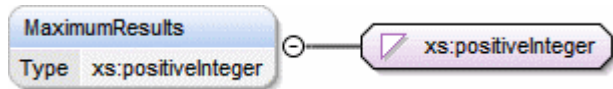
`xmsg:MaximumPackages` gibt an, aus wieviel Paketen eine Antwort maximal bestehen darf, um die Anforderung zu befriedigen.

Beispiel

```
<xmsg:Control>
  <xmsg:MaximumPackages>10</xmsg:MaximumPackages>
</xmsg:Control>
```


Element *MaximumResults*

	xmsg:MaximumResults
<i>Inhalt</i>	Eine positive Ganzzahl
<i>Datentyp</i>	xsd:positiveInteger
<i>Benutzt von</i>	xmsg:Control



Beschreibung

xmsg:MaximumResults gibt an, wieviele Treffer maximal übertragen werden sollen, um die Anforderung zu befriedigen.

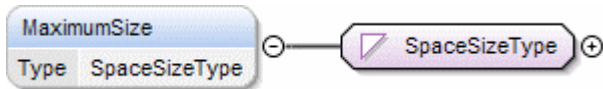
Beispiel

```
<xmsg:Control>
  <xmsg:MaximumResults>32</xmsg:MaximumResults>
</xmsg:Control>
```

Element *MaximumSize*

xmsg:MaximumSize

<i>Inhalt</i>	Eine positive Ganzzahl
<i>Datentyp</i>	xmsg:SpaceSizeType
<i>Benutzt von</i>	xmsg:Control



Beschreibung

Gibt an, wie groß die Antwort im Element `TransportBody` maximal sein darf, um die Anforderung zu befriedigen. Der Wert hat die Form nU , wobei n eine Ganzzahl > 0 und U einer der folgenden Großbuchstaben ist, der die Einheit angibt:

B	Byte
K	Kilobyte
M	Megabyte
G	Gigabyte

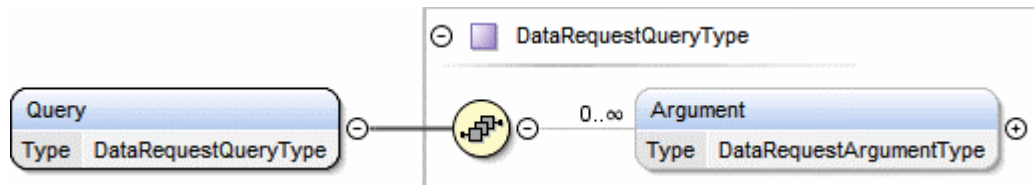
Beispiel

```
<xmsg:Control>
  <xmsg:MaximumSize>10M</xmsg:MaximumSize>
</xmsg:Control>
```

Element Query

xmsg:Query

<i>Inhalt</i>	Beschreibt den Abfrageteil der Anforderung.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:DataRequest



Beschreibung

Beschreibt eine Abfrage als ein Set von Kriterien (Element `Argument`). Jedes Kriterium ist mit einem Typ behaftet, spezifiziert einen oder mehrere Vergleichswerte und kann auf ein spezifisches Ereignis in einem eXtra basierten Prozess bezogen werden.

Nachrichtentyp ListOfConfirmationOfReceipt [normativ]

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs *ConfirmationOfReceipt*.

Der Name des Wurzelementes ist `ListOfConfirmationOfReceipt`.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

Beispieldokument

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:ListOfConfirmationOfReceipt
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.1"
  >

  <!-- #1 -->

  <xmsg:ConfirmationOfReceipt>
    <xmsg:Property
      name="http://www.extra-standard.de/property/SenderID"
      type="xs:string">
      <xmsg:Value>77700000</xmsg:Value>
    </xmsg:Property>

    <xmsg:Property
      name="http://www.extra-standard.de/property/ReceiverID"
      type="xs:string">
      <xmsg:Value>55550000</xmsg:Value>
    </xmsg:Property>

    <xmsg:Property
      name="http://www.extra-standard.de/property/Procedure"
      type="xs:anyURI">
      <xmsg:Value>http://www.extra-standard.de/procedures/DEUEV</xmsg:Value>
    </xmsg:Property>

    <xmsg:Property
      name="http://www.extra-standard.de/property/DataType"
      type="xs:anyURI">
      <xmsg:Value>http://www.extra-standard.de/datatypes/RSVNr</xmsg:Value>
    </xmsg:Property>

    <xmsg:Property
      name="http://www.extra-standard.de/property/ResponseCreationTimeStamp"
      type="xs:dateTime">
      <xmsg:Value>2006-05-04T18:13:51.0Z</xmsg:Value>
    </xmsg:Property>

    <xmsg:Property
      name="http://www.extra-standard.de/property/ResponseID"
      type="xs:string"
      event="http://www.extra-standard.de/event/RequestData">
      <xmsg:Value>res20090223152540-00008060</xmsg:Value>
    </xmsg:Property>
```

```
<xmsg:Property
  name="http://www.extra-standard.de/property/ResponseFileName"
  type="xs:string"
  event="http://www.extra-standard.de/event/RequestData">
  <xmsg:Value>file20090223152540-00008060</xmsg:Value>
</xmsg:Property>

</xmsg:ConfirmationOfReceipt>

<!-- #2 -->

<xmsg:ConfirmationOfReceipt>
  <xmsg:Property
    name="http://www.extra-standard.de/property/SenderID"
    type="xs:string">
    <xmsg:Value>88800000</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property
    name="http://www.extra-standard.de/property/ReceiverID"
    type="xs:string">
    <xmsg:Value>66660000</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property
    name="http://www.extra-standard.de/property/Procedure"
    type="xs:anyURI">
    <xmsg:Value>http://www.extra-standard.de/procedures/DEUEV</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property
    name="http://www.extra-standard.de/property/DataType"
    type="xs:anyURI">
    <xmsg:Value>http://www.extra-standard.de/datatypes/RSVNr</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property
    name="http://www.extra-standard.de/property/ResponseCreationTimeStamp"
    type="xs:dateTime">
    <xmsg:Value>2006-05-04T18:13:52.0Z</xmsg:Value>
  </xmsg:Property>

  <xmsg:Property
    name="http://www.extra-standard.de/property/ResponseID"
    type="xs:string"
    event="http://www.extra-standard.de/event/RequestData">
    <xmsg:Value>res20090223152541-00008060</xmsg:Value>
  </xmsg:Property>
```



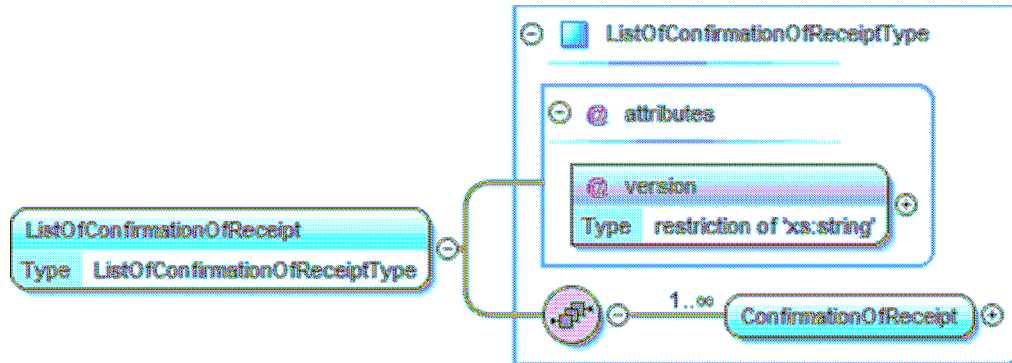
```
<xmsg:Property
  name="http://www.extra-standard.de/property/ResponseFileName"
  type="xs:string"
  event="http://www.extra-standard.de/event/RequestData">
  <xmsg:Value>file20090223152541-00008060</xmsg:Value>
</xmsg:Property>

</xmsg:ConfirmationOfReceipt>
</xmsg:ListOfConfirmationOfReceipt>
```

Element *ListOfConfirmationOfReceipt*

xmsg:ListOfConfirmationOfReceipt

<i>Inhalt</i>	Eine Menge von Nachrichten des Typs <i>ConfirmationOfReceipt</i> .
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.1
<i>Datentyp</i>	xsd:string

Beschreibung

`xmsg:ListOfConfirmationOfReceipt` ist das Wurzelement einer eXtra Nachricht vom Typ `ConfirmationOfReceipt`. Das Attribut `version` enthält die Versionsbezeichnung.

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs *ConfirmationOfReceipt*

Nachrichtentyp ListOfDataRequest [normativ]

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs DataRequest.

Der lokale Name des Wurzelementes ist `ListOfDataRequest`.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

Beispieldokument

```

<?xml version="1.0" encoding="UTF-8"?>
<xmsg:ListOfDataRequest
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.1"
  >
  <!-- #1 -->

  <xmsg:DataRequest>
    <xmsg:Query>

      <xmsg:Argument property="http://www.extra-standard.de/property/SenderID"
        type="xs:string">
        <xmsg:EQ>5550000</xmsg:EQ>
      </xmsg:Argument>

      <xmsg:Argument property="http://www.extra-standard.de/property/ReceiverID"
        type="xs:string">
        <xmsg:EQ>7770000</xmsg:EQ>
      </xmsg:Argument>

      <xmsg:Argument property="http://www.extra-standard.de/property/Procedure"
        type="xs:anyURI">
        <xmsg:EQ>http://my.procedure.com/procl</xmsg:EQ>
      </xmsg:Argument>

      <xmsg:Argument property="http://www.extra-standard.de/property/DataType"
        type="xs:anyURI">
        <xmsg:EQ>http://my.datatype.com/type1</xmsg:EQ>
      </xmsg:Argument>

      <xmsg:Argument property="http://www.extra-standard.de/property/ResponseID"
        type="xs:string"
        event="http://www.extra-standard.de/event/RequestData">
        <xmsg:EQ>R46934503</xmsg:EQ>
      </xmsg:Argument>

      <xmsg:Argument
        property="http://www.extra-standard.de/property/ResponseFileName"
        type="xs:string"
        event="http://www.extra-standard.de/event/RequestData">
        <xmsg:EQ>R46934503</xmsg:EQ>
      </xmsg:Argument>

      <xmsg:Argument
        property="http://www.extra-standard.de/property/ResponseCreationTimeStamp"
        type="xs:dateTime">
        <xmsg:GT>2009-03-16T09:00:00</xmsg:GT>
        <xmsg:LE>2009-03-17T09:00:00</xmsg:LE>
      </xmsg:Argument>

    </xmsg:Query>

    <xmsg:Control>
      <xmsg:MaximumPackages>10</xmsg:MaximumPackages>
      <xmsg:MaximumMessages>100</xmsg:MaximumMessages>
      <xmsg:MaximumSize>10M</xmsg:MaximumSize>
    </xmsg:Control>

  </xmsg:DataRequest>

  <!-- #2 -->

  <xmsg:DataRequest>
    <xmsg:Query>

      <xmsg:Argument property="http://www.extra-standard.de/property/SenderID"
        type="xs:string">

```

```
<xmsg:EQ>5550000</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument property="http://www.extra-standard.de/property/ReceiverID"
  type="xs:string">
  <xmsg:EQ>7770000</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument property="http://www.extra-standard.de/property/Procedure"
  type="xs:anyURI">
  <xmsg:EQ>http://my.procedure.com/procl</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument property="http://www.extra-standard.de/property/DataType"
  type="xs:anyURI">
  <xmsg:EQ>http://my.datatype.com/type1</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument property="http://www.extra-standard.de/property/ResponseID"
  type="xs:string"
  event="http://www.extra-standard.de/event/RequestData">
  <xmsg:EQ>R46934503</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument
  property="http://www.extra-standard.de/property/ResponseFileName"
  type="xs:string"
  event="http://www.extra-standard.de/event/RequestData">
  <xmsg:EQ>R46934503</xmsg:EQ>
</xmsg:Argument>

<xmsg:Argument
  property="http://www.extra-standard.de/property/ResponseCreationTimeStamp"
  type="xs:dateTime">
  <xmsg:GT>2009-03-16T09:00:00</xmsg:GT>
  <xmsg:LE>2009-03-17T09:00:00</xmsg:LE>
</xmsg:Argument>

</xmsg:Query>

<xmsg:Control>
  <xmsg:MaximumPackages>10</xmsg:MaximumPackages>
  <xmsg:MaximumMessages>100</xmsg:MaximumMessages>
  <xmsg:MaximumSize>10M</xmsg:MaximumSize>
</xmsg:Control>

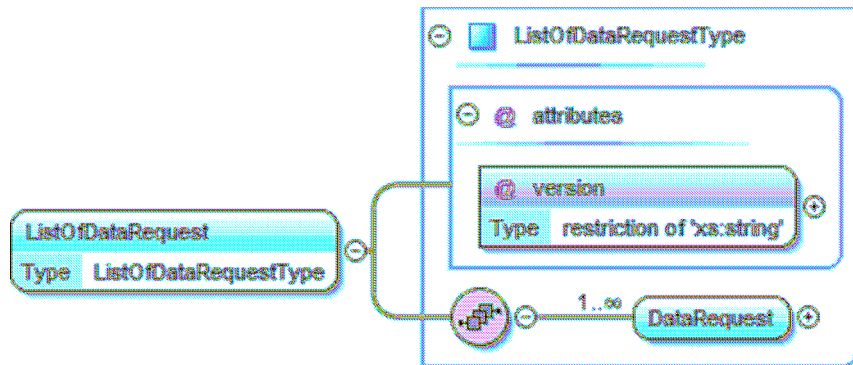
</xmsg:DataRequest>

</xmsg:ListOfDataRequest>
```

Element *ListOfDataRequest*

xmsg:ListOfDataRequest

<i>Inhalt</i>	Eine Menge von Nachrichten des Typs <i>DataRequest</i> .
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.1
<i>Datentyp</i>	<code>xsd:string</code>

Beschreibung

`xmsg:ListOfDataRequest` ist das Wurzelement einer eXtra Nachricht vom Typ `DataRequest`. Das Attribut `version` enthält die Versionsbezeichnung.

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs *DataRequest*

Nachrichtentyp ListOfStatusResponse [normativ]

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs *StatusResponse*.

Der Name des Wurzelementes ist `ListOfStatusResponse`.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

Beispieldokument

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:ListOfStatusResponse
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.0"
  >

  <xmsg:StatusResponse>

    <xmsg:Property name="http://www.extra-standard.de/property/RequestID">
      <xmsg:Value>12345</xmsg:Value>
    </xmsg:Property>

    <xmsg:Trace>
      <xmsg:Checkpoint>
        <xmsg:Layer>http://www.extra-standard.de/layer/Transport</xmsg:Layer>
        <xmsg:Status>http://www.extra-standard.de/status/COMPLETED</xmsg:Status>
        <xmsg:Timestamp>2011-05-08T22:00:10</xmsg:Timestamp>
      </xmsg:Checkpoint>
      <xmsg:Checkpoint>
        <xmsg:Layer>http://www.extra-standard.de/layer/Application</xmsg:Layer>
        <xmsg:Status>http://www.extra-standard.de/status/PROCESSING</xmsg:Status>
        <xmsg:Timestamp>2011-05-08T22:00:10</xmsg:Timestamp>
        <xmsg:Report highestWeight="http://www.extra-standard.de/weight/OK">
          <xmsg:Flag weight="http://www.extra-standard.de/weight/OK">
            <xmsg:Code>20</xmsg:Code>
            <xmsg:Text>Et it noch allet jot jejange!</xmsg:Text>
          </xmsg:Flag>
        </xmsg:Report>
      </xmsg:Checkpoint>
    </xmsg:Trace>

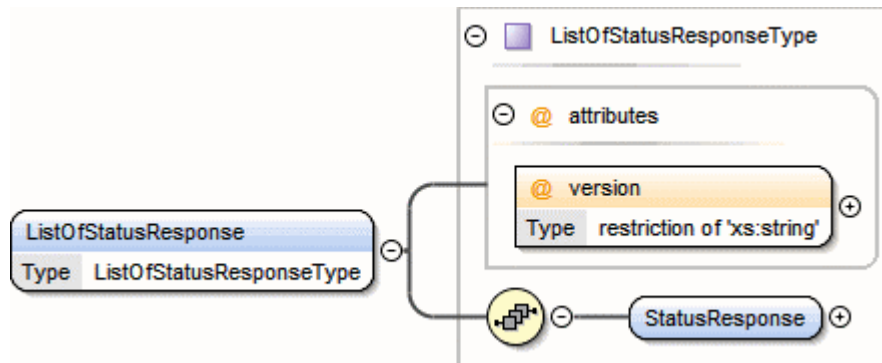
  </xmsg:StatusResponse>

</xmsg:ListOfStatusResponse>
```

Element *ListOfStatusResponse*

xmsg:ListOfDataRequest

<i>Inhalt</i>	Eine Menge von Nachrichten des Typs <i>StatusResponse</i>
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.0
<i>Datentyp</i>	<code>xsd:string</code>

Beschreibung

`xmsg:ListOfStatusResponse` ist das Wurzelement einer eXtra Nachricht vom Typ *ListOfStatusResponse*. Das Attribut `version` enthält die Versionsbezeichnung.

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs *StatusResponse*

Nachrichtentyp RepeatResponse [normativ]

Mit einer Nachricht des Typs *RepeatResponse* wird das erneute Senden einer Response angefordert.

Die Nachricht wird nicht durch eine XML Struktur repräsentiert, sondern entsteht durch Senden eines eXtra Request Headers, dessen Eigenschaft `DataType` auf den Wert

`http://www.extra-standard.de/datatype/RepeatResponse`

gesetzt ist.

Nachrichtentyp StatusRequest [normativ]

Mit einer Nachricht des Typs *StatusRequest* kann der Verarbeitungszustand ausgewählter eXtra Requests abgefragt werden.

Der Name des Wurzelementes ist *StatusRequest*.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

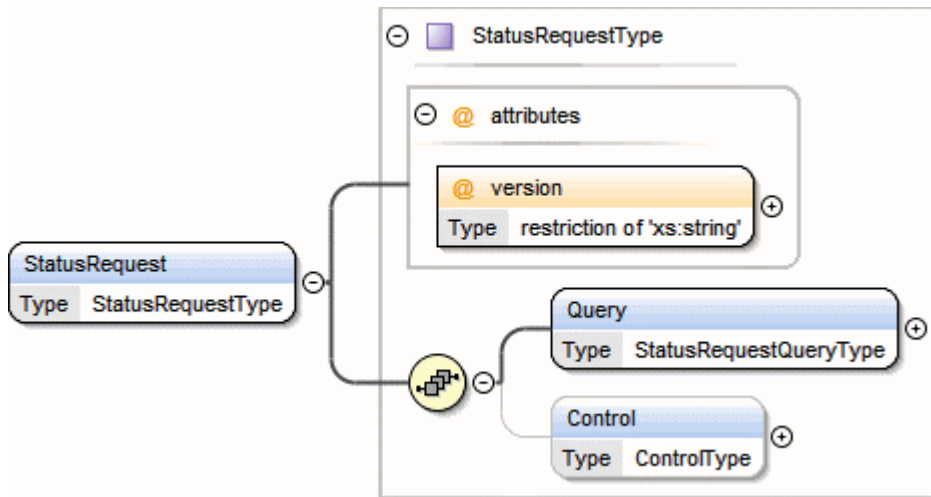
Beispieldokument

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:StatusRequest
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.0"
  >
  <xmsg:Query>
    <xmsg:Argument property="http://www.extra-standard.de/property/DataType">
      <xmsg:EQ>AAAAAA</xmsg:EQ>
    </xmsg:Argument>
    <xmsg:Argument property="http://www.extra-standard.de/property/RequestID">
      <xmsg:EQ>12345</xmsg:EQ>
    </xmsg:Argument>
    <xmsg:Argument property="http://www.extra-
standard.de/property/RequestCreationTimeStamp">
      <xmsg:EQ>2011-05-08T22:00:00</xmsg:EQ>
    </xmsg:Argument>
    <xmsg:Argument property="http://www.w3.org">
      <xmsg:EQ>aaaa</xmsg:EQ>
    </xmsg:Argument>
  </xmsg:Query>
</xmsg:StatusRequest>
```

Element *StatusRequest*

xmsg:StatusRequest

<i>Inhalt</i>	Eine Nachricht des Typs <i>StatusRequest</i>
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.0
<i>Datentyp</i>	xsd:string

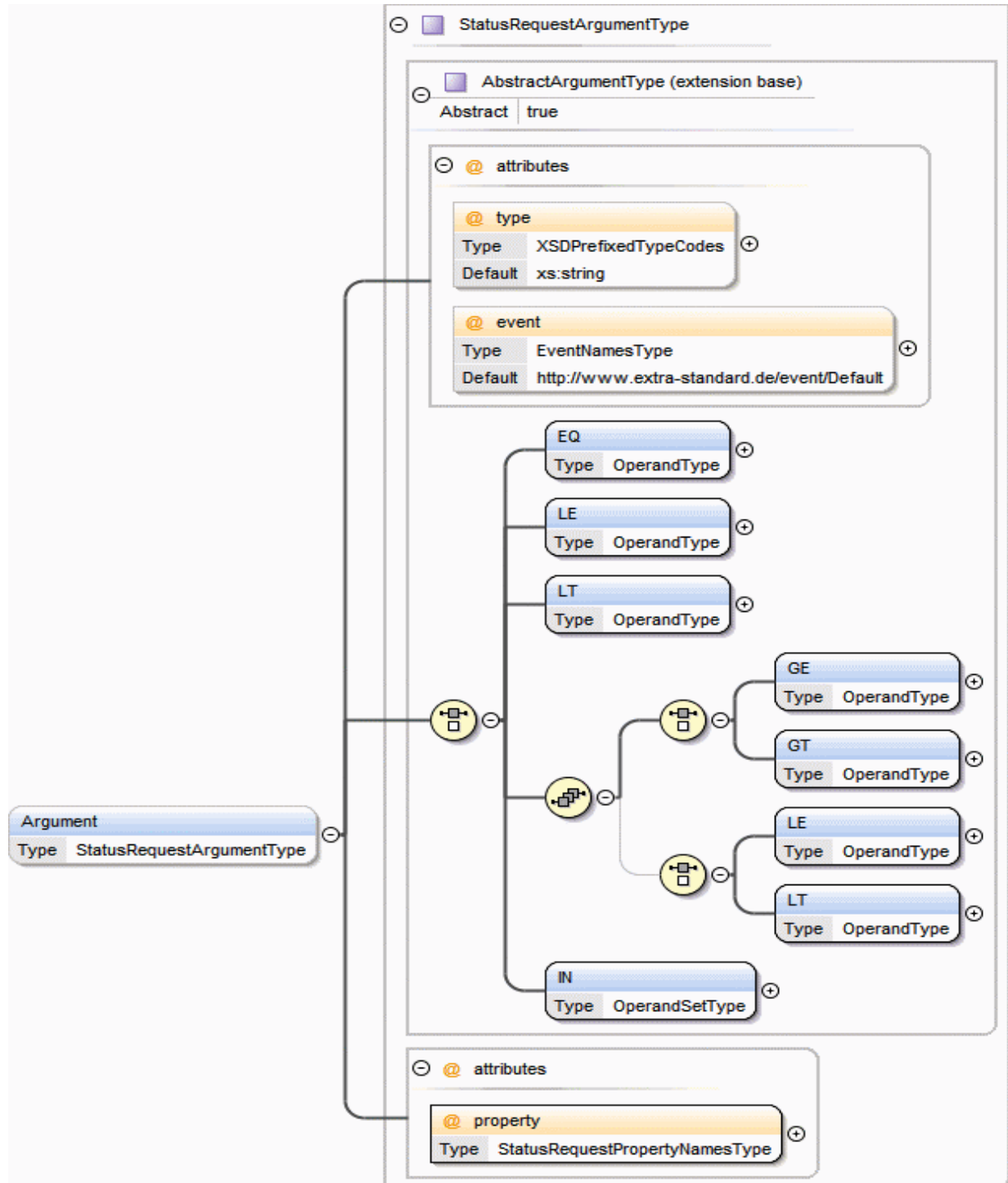
Beschreibung

`xmsg:StatusRequest` ist das Wurzelement einer eXtra Nachricht vom Typ *StatusRequest*. Das Attribut `version` enthält die Versionsbezeichnung.

Element Argument

xmsg:Argument

Inhalt	Beschreibt ein Kriterium einer Abfrage
Datentyp	##elements
Benutzt von	xmsg:Query



Attribute

	@event	<i>optional</i>
<i>Inhalt</i>	Bezieht die abgefragte Eigenschaft auf ein Ereignis im Kommunikationsablauf	
<i>Datentyp</i>	xsd:anyURI	
<i>Werteraum</i>	##codelist:EventNames	
<i>Default</i>	http:www.extra-standard.de/event/Default	
	@property	<i>erforderlich</i>
<i>Inhalt</i>	Bezeichnet die abgefragte Eigenschaft	
<i>Datentyp</i>	xsd:anyURI	
<i>Werteraum</i>	##codelist:StatusRequestPropertyNames	
	@type	<i>optional</i>
<i>Inhalt</i>	Der XSD Datentyp der abgefragten Eigenschaft	
<i>Datentyp</i>	xsd:string	
<i>Werteraum</i>	##codelist:LayerNames	
<i>Default</i>	xsd:string	

Profilierung

<i>Komponente</i>	<i>REQ</i>	<i>OPT</i>	<i>FOR</i>	<i>MUL</i>	<i>IGN</i>	<i>VAL</i>
@event						+

Beschreibung

Ein Kriterium besteht aus einer abgefragten Eigenschaft (Attribut `property`), die per URI referenziert wird, und einem Werteraum, mit dem die Werte der abgefragten Objekte verglichen wird. Der Werteraum wird als Einzelwert oder als Wertebereich mit inklusiver Unter- und Obergrenze oder als Werteliste angegeben. Die wertetragenden Elemente des Werteraums bestimmen zugleich den Operator für den Vergleich mit Einzelwerten bzw. der Unter- oder Obergrenze.

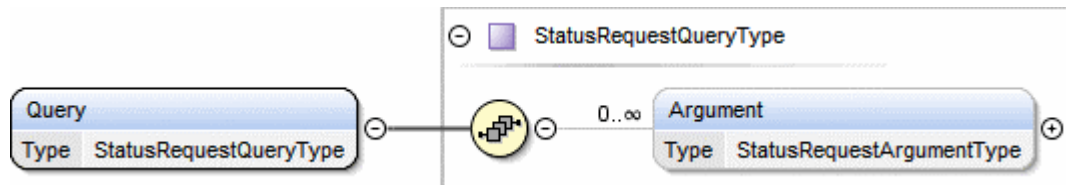
Das Attribut `type` spezifiziert den Datentyp des Arguments.

Mit dem Attribut `event` wird das Ereignis im Kommunikationsablauf spezifiziert, für das die Eigenschaft abgefragt werden soll. Die vordefinierte Werteliste kann durch nutzerspezifische URIs ersetzt bzw. ergänzt werden.

Element Query

xmsg:Query

<i>Inhalt</i>	Beschreibt den Abfrageteil der Anforderung.
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:StatusRequest



Beschreibung

Beschreibt eine Abfrage als ein Set von Kriterien (Element `Argument`). Jedes Kriterium ist mit einem Typ behaftet, spezifiziert einen oder mehrere Vergleichswerte und kann auf ein spezifisches Ereignis in einem eXtra basierten Prozess bezogen werden.

Nachrichtentyp `StatusResponse` [normativ]

Mit einer Nachricht des Typs *StatusResponse* wird das Ergebnis einer Statusabfrage über den Verarbeitungszustand einer fachlichen Nachricht übertragen. Mengen von Nachrichten dieses Typs können mit *ListOfStatusResponse* gebündelt werden. Statusabfragen erfolgen typischerweise per Nachrichten des Type *StatusRequest*.

Der Name des Wurzelementes ist `StatusResponse`.

Zu Namensraum und Präfix siehe Abschnitt *Namensräume und Präfixe*.

Beispieldokument

```
<?xml version="1.0" encoding="UTF-8"?>
<xmsg:StatusResponse
  xmlns:xmsg="http://www.extra-standard.de/namespace/message/1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  version="1.0"
  >

  <xmsg:Property name="http://www.extra-standard.de/property/RequestID">
    <xmsg:Value>12345</xmsg:Value>
  </xmsg:Property>

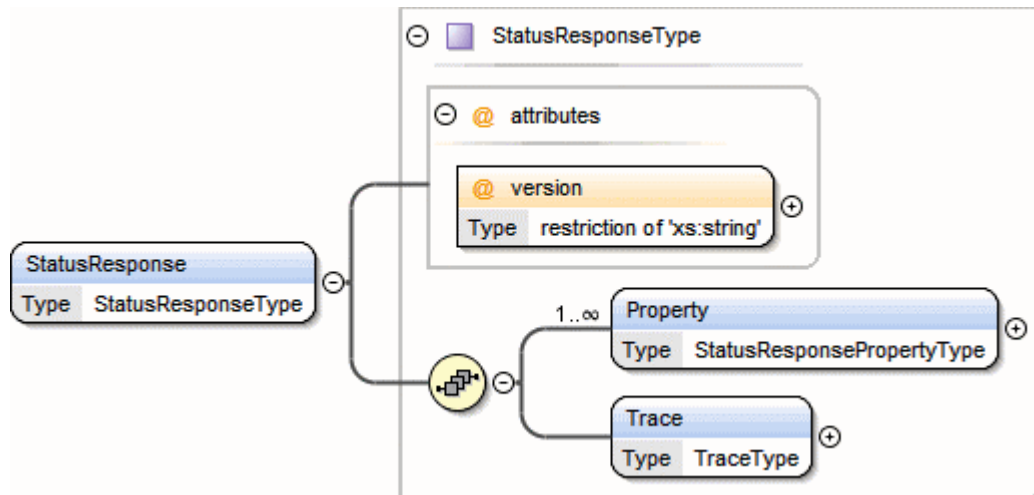
  <xmsg:Trace>
    <xmsg:Checkpoint>
      <xmsg:Layer>http://www.extra-standard.de/layer/Transport</xmsg:Layer>
      <xmsg:Status>http://www.extra-standard.de/status/COMPLETED</xmsg:Status>
      <xmsg:Timestamp>2011-05-08T22:00:10</xmsg:Timestamp>
    </xmsg:Checkpoint>
    <xmsg:Checkpoint>
      <xmsg:Layer>http://www.extra-standard.de/layer/Application</xmsg:Layer>
      <xmsg:Status>http://www.extra-standard.de/status/PROCESSING</xmsg:Status>
      <xmsg:Timestamp>2011-05-08T22:00:10</xmsg:Timestamp>
      <xmsg:Report highestWeight="http://www.extra-standard.de/weight/OK">
        <xmsg:Flag weight="http://www.extra-standard.de/weight/OK">
          <xmsg:Code>20</xmsg:Code>
          <xmsg:Text>Et it noch allet jot jejange!</xmsg:Text>
        </xmsg:Flag>
      </xmsg:Report>
    </xmsg:Checkpoint>
  </xmsg:Trace>

</xmsg:StatusResponse>
```


Element *StatusResponse*

xmsg:StatusResponse

<i>Inhalt</i>	Eine Nachricht des Typs <i>StatusResponse</i>
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	-



Attribute

@version

erforderlich

<i>Inhalt</i>	Enthält die Versionsbezeichnung dieser eXtra Standardnachricht, derzeit 1.0
<i>Datentyp</i>	xsd:string

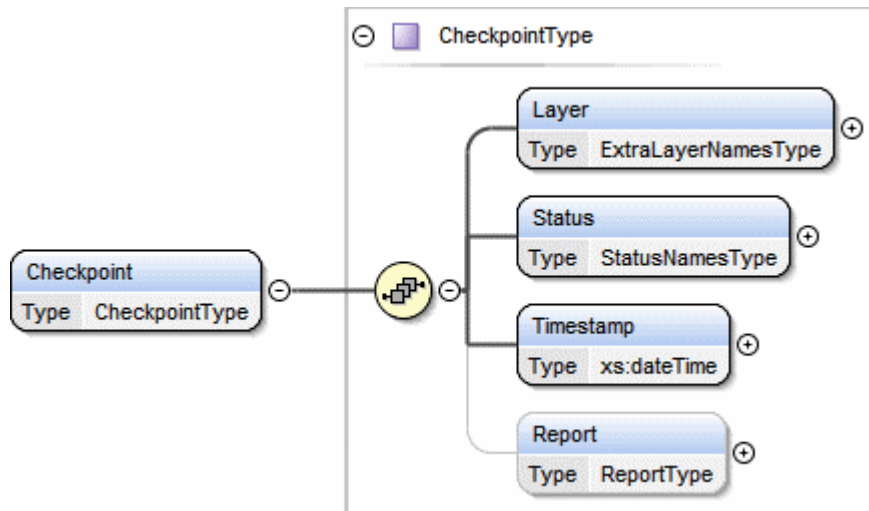
Beschreibung

`xmsg:StatusResponse` ist das Wurzelement einer eXtra Nachricht vom Typ *StatusResponse*. Das Attribut `version` enthält die Versionsbezeichnung.

Element Checkpoint

xmsg:Checkpoint

<i>Inhalt</i>	Checkpoint für einen definierten Punkt der Prozesskette
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:Trace



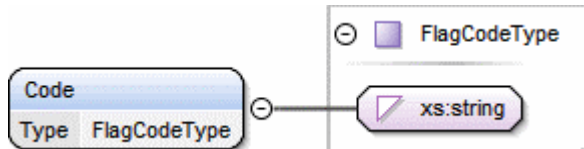
Beschreibung

Ein Checkpoint beschreibt den Verarbeitungszustand eines eXtra Request an einem definierten Punkt der Prozesskette.

Element Code

xmsg:Code

<i>Inhalt</i>	Ein Schlüssel, der eine kodierte Information darstellt, z.B. ein Fehlercode
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Flag



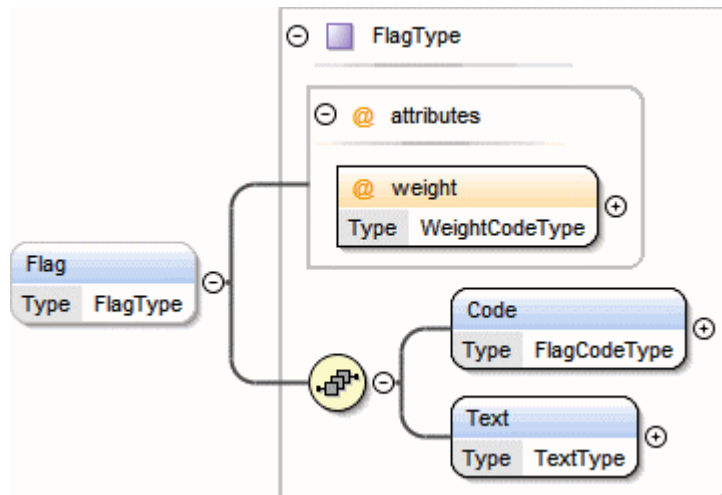
Beschreibung

`xmsg:Code` dient zur Angabe eines Fehlerschlüssels oder einer anderen kodierten Information in Kombination mit einer durch `xmsg:Text` dargestellten textuellen Ausprägung.

Element Flag

xmsg:Flag

<i>Inhalt</i>	Flag mit Gewicht, Code und Text
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:Report



Attribute

@weight

erforderlich

<i>Inhalt</i>	Gewicht des Flags
<i>Datentyp</i>	xsd:anyURI
<i>Werteraum</i>	##codelist:XSDWeightCodes

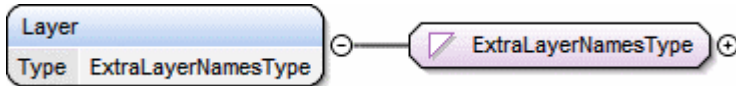
Beschreibung

Flags repräsentieren Verarbeitungsergebnisse durch einen formalen Code, einen erläuternden Text und ein Fehlergewicht.

Element Layer

xmsg:Layer

<i>Inhalt</i>	Spezifiziert eine Ebene des eXtra Kommunikationsmodells.
<i>Datentyp</i>	xsd:anyURI
<i>Werteraum</i>	##codelist:LayerNames
<i>Benutzt von</i>	xmsg:Checkpoint



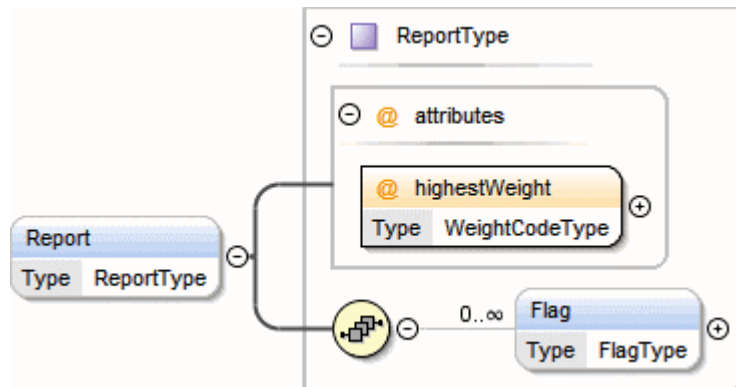
Beschreibung

`xmsg:Layer` spezifiziert in einem Checkpoint die Ebene des eXtra Kommunikationsmodells. Die Werte sind abschließend in der Codelist *LayerNames* definiert.

Element Report

xmsg:Report

<i>Inhalt</i>	Enthält alle Flags für den gegebenen Checkpoint
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:Checkpoint



Attribute

@highestWeight

erforderlich

<i>Inhalt</i>	Das höchste vergebene Fehlergewicht aller Flags des Reports.
<i>Datentyp</i>	xsd:anyURI
<i>Werteraum</i>	##codelist:XSDWeightCodes

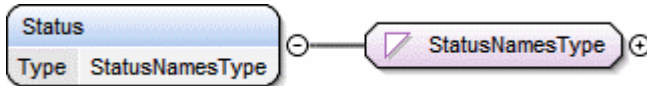
Beschreibung

Ein Report besteht aus einer Menge von Flags, die Informationen, Warnungen oder Fehler beschreiben.

Element Status

xmsg:Status

<i>Inhalt</i>	Spezifiziert den Status der Verarbeitung an einem Checkpoint.
<i>Datentyp</i>	xsd:anyURI
<i>Werteraum</i>	##codelist:StatusNames
<i>Benutzt von</i>	xmsg:Checkpoint

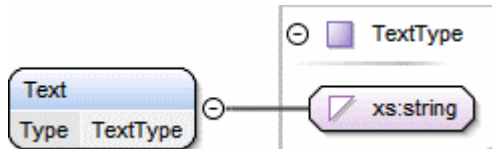


Beschreibung

`xmsg:Status` spezifiziert in einem Checkpoint den Verarbeitungsstatus. Die Werte sind in der Codelist *StatusNames* definiert.

Element Text

	xmsg:Text
<i>Inhalt</i>	Ein Text
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	xmsg:Flag



Beschreibung

`xmsg:Text` enthält in Verbindung mit `xmsg:Code` die textuelle Ausprägung einer kodierten Information wie z.B. eines Fehlerschlüssels..

Element Timestamp

	xmsg:Timestamp
<i>Inhalt</i>	Ein Zeitstempel im ISO Format
<i>Datentyp</i>	xsd:dateTime
<i>Benutzt von</i>	xmsg:Checkpoint



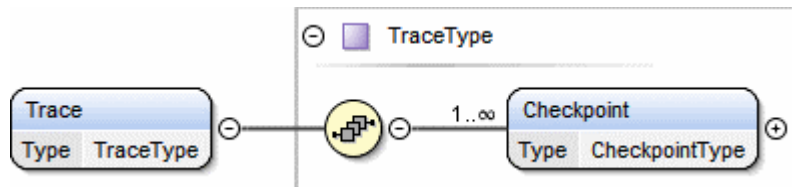
Beschreibung

xmsg:Timestamp beschreibt den Zeitpunkt, an dem ein eXtra Request an dem durch xmsg:Checkpoint definierten Punkt der Prozesskette angekommen ist.

Element Trace

xmsg:Trace

<i>Inhalt</i>	Eine Folge von Checkpoints
<i>Datentyp</i>	##elements
<i>Benutzt von</i>	xmsg:StatusResponse



Beschreibung

`xmsg:Trace` enthält eine Folge von Checkpoints. Jeder Checkpoint beschreibt den Verarbeitungsstatus eines Teilprozesses in einer eXtra basierten Prozesskette.

Ressourcen

XSD Schema Ressourcen

Das eXtra XSD Schema für Standardnachrichten besteht aus den folgenden Schemadateien:

eXtra-messages-1.xsd

eXtra Standardnachrichten

Literatur

Kurzname	Quelle
DSIG	<i>eXtra Design Guidelines</i> , zu finden unter www.extra-standard.de
EINF	<i>Einführung in den eXtra Standard</i> , zu finden unter www.extra-standard.de
EMSG	<i>eXtra Standardnachrichten, Schnittstellenbeschreibung</i> , zu finden unter www.extra-standard.de
EXSEC	<i>Sicherheit und Verfügbarkeit in einem eXtra spezifischen Datenübermittlungsverbund</i> , zu finden unter www.extra-standard.de
EXVS	<i>eXtra Versionierung</i> , zu finden unter www.extra-standard.de
EXWS	<i>eXtra und Webservices</i> , zu finden unter www.extra-standard.de
IFACE	<i>eXtra Transport Schnittstellenbeschreibung</i> , zu finden unter www.extra-standard.de
IMPL	<i>eXtra Implementierung</i>
KOMP	<i>eXtra Kompendium</i> , zu finden unter www.extra-standard.de
RFC2119	<i>Request for Comments: 2119</i> , S. Bradner, Harvard University, March 1997, http://www.ietf.org/rfc/rfc2119.txt
PROF	<i>eXtra Profilierung</i> , zu finden unter www.extra-standard.de
RELAX	http://relaxng.org/
XENC	<i>XML Encryption</i> , http://www.w3.org/TR/xmlenc-core/
XML	<i>XML Recommendation 1.0, 3rd Edition</i> , http://www.w3.org/XML
XSD	<i>XML Schema Definition</i> , http://www.w3.org/TR/xmlschema-0/
XSIG	<i>XML Signature</i> , http://www.w3.org/TR/xmldsig-core/
XSL	<i>XML Stylesheet Language</i> , http://www.w3.org/TR/1999/REC-xslt-19991116 , http://www.w3.org/TR/xslt20/