



*einheitliches XML-basiertes Transportverfahren*

**eXTra Basis-Standard**

**eXTra Standardnachrichten Version 1.2**

**Schnittstellenbeschreibung  
Ausgabestand 1.2.1**

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](mailto:info@awv-net.de)  
Web: [www.extra-standard.de](http://www.extra-standard.de), [www.awv-net.de](http://www.awv-net.de).

## Änderungsprotokoll für Version 1.2. Ausgabestand 1.2.1

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

## Änderungsprotokoll für Version 1.1. bis 1.1.1

<b>Autor[en]</b>	<b>Datum</b>	<b>Beschreibung</b>
[gelöscht]	01.11.2009	Erstellung 1.1
	08.03.2010	Freigabe

## Inhaltsverzeichnis

Einleitung und Übersicht.....	5
Layout und Notationen .....	7
Formatierungen .....	7
Beschreibung der Regeln .....	7
Beschreibung der Elemente und Attribute .....	7
Graphische Symbole .....	9
Definitionen [normativ].....	12
Begriffe.....	12
Namensräume und Präfixe .....	12
eXtra Codelisten [normativ] .....	13
Codeliste DataRequestPropertyNames .....	14
Codeliste PropertyNames .....	14
Codeliste EventNames .....	15
Nachrichtentyp ConfirmationOfReceipt [normativ] .....	16
Beispieldokument .....	17
Element ConfirmationOfReceipt .....	18
Element Property .....	20
Element PropertySet .....	23
Element Value .....	25
Nachrichtentyp DataRequest [normativ] .....	26
Beispieldokument .....	27
Element DataRequest .....	29
Element Argument.....	31
Element Control.....	34
Element EQ .....	35
Element GE .....	36
Element GT .....	37
Element LE.....	38
Element LT .....	39
Element MaximumMessages.....	40
Element MaximumPackages .....	41
Element MaximumSize .....	42
Element Query .....	43
Nachrichtentyp ListOfConfirmationOfReceipt [normativ] .....	44
Beispieldokument .....	45
Element ListOfConfirmationOfReceipt .....	46
Nachrichtentyp ListOfDataRequest [normativ] .....	47
Beispieldokument .....	48
Element ListOfDataRequest .....	49
Ressourcen .....	50
XSD Schema Ressourcen .....	50
Literatur.....	51

## Einleitung und Übersicht

*Gültigkeit und Versionierung:* Dieses Dokument enthält die normative technische Spezifikation der Schnittstelle der eXtra Standardnachrichten in der Version 1.2.1 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 1.2) der zugrundeliegenden Schnittstelle der eXtra Standardnachrichten angeführt, während die letzte Ziffer r den jeweiligen Ausgabestand (hier Ausgabestand 1) 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>
<b>xcpt:Signatures</b>	+	+	+		+	
<b>xcpt:Data</b>				<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)

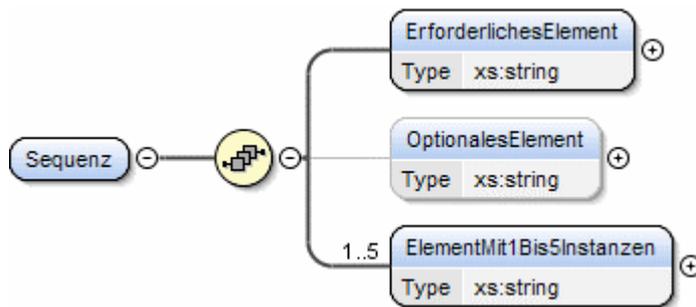
### **Graphische Symbole**

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

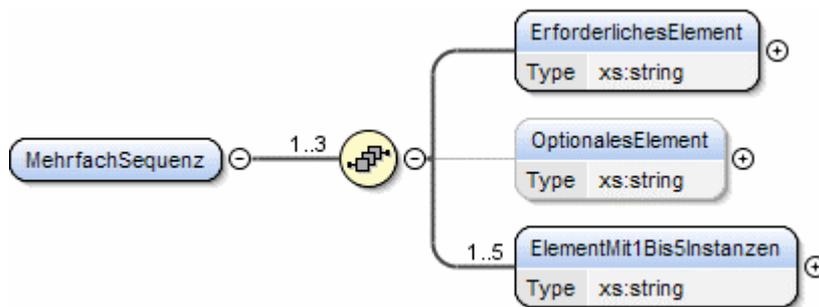
*Hinweis*: In den meisten Fällen erscheinen Elemente und Attribute in einem im Farbruck gelb eingefärbten Rechteck. Dieses stellt eine XSD-Typdefinition dar und trägt deren Namen. Typdefinitionen sind schemaspezifisch und daher nicht normativ. Aus diesem Grund findet sich in den folgenden Mustergraphiken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

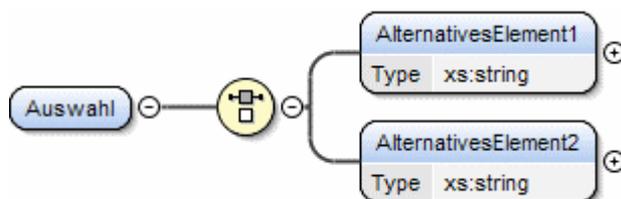
### Eine Sequenz von Elementen



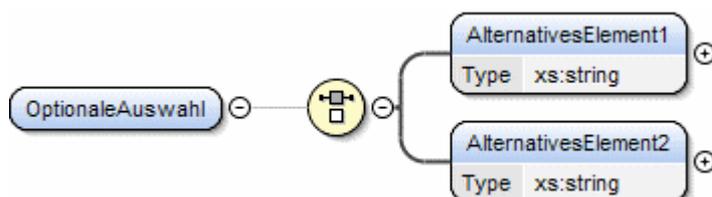
### Eine bis zu 3 mal wiederholte Sequenz von Elementen



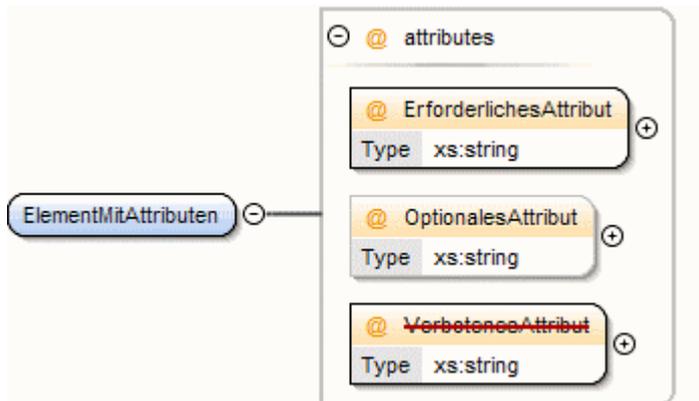
### Eine erforderliche Auswahl (Kindelemente sind Pflichtelemente)



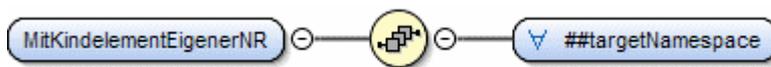
### Eine optionale Auswahl von (Kindelemente sind Pflichtelemente)



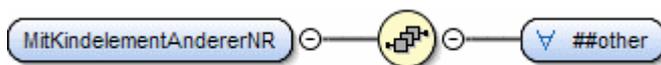
### Ein Element mit Attributen



### 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***

eXtra verwendet die folgenden Namensräume und Präfixe. 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

## eXtra Codelisten [normativ]

Die im Folgenden beschriebenen Codelisten sind im Schema-Modul `eXtra-codelists-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:

<b>Gültige Ausdrücke</b>		<b>Ungültige Ausdrücke</b>	
#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***

<b>xmsg:DataRequestPropertyNamesType</b>	
<i>Inhalt</i>	Identifikatoren abfragbare Eigenschaften
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Nein

### *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>

## **Beschreibung**

Die Codeliste MUSS ein Subset der Werte der Codeliste `xmsg:PropertyNamesType` darstellen.

## ***Codeliste PropertyNames***

<b>xmsg: PropertyNamesType</b>	
<i>Inhalt</i>	Identifikatoren abfragbare Eigenschaften
<i>Datentyp</i>	xsd:anyURI
<i>Individualisierbar</i>	Nein
<i>Beliebige Domain</i>	Nein

### *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>

## Beschreibung

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

## ***Codeliste EventNames***

	<b>xmsg:EventNamesType</b>
<i>Inhalt</i>	Identifikatoren abfragbare Eigenschaften
<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.

## Nachrichtentyp *ConfirmationOfReceipt* [normativ]

Eine Nachricht vom Typ *ConfirmationOfReceipt* ist eine Empfangsbestätigung für eine Menge von Anwendungsnachrichten. Die Namensraum- URI ist

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

Der lokale Name des Wurzelementes ist *ConfirmationOfReceipt*. Es wird empfohlen, den Namensraumpräfix *xmsg* für alle eXtra Standardnachrichten zu verwenden. Für die Unterscheidung dieses Nachrichtentyps von anderen desselben Namensraumes wird alternativ der Namensraumpräfix *xcor* empfohlen.

## 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="1.2"
  >
  <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>

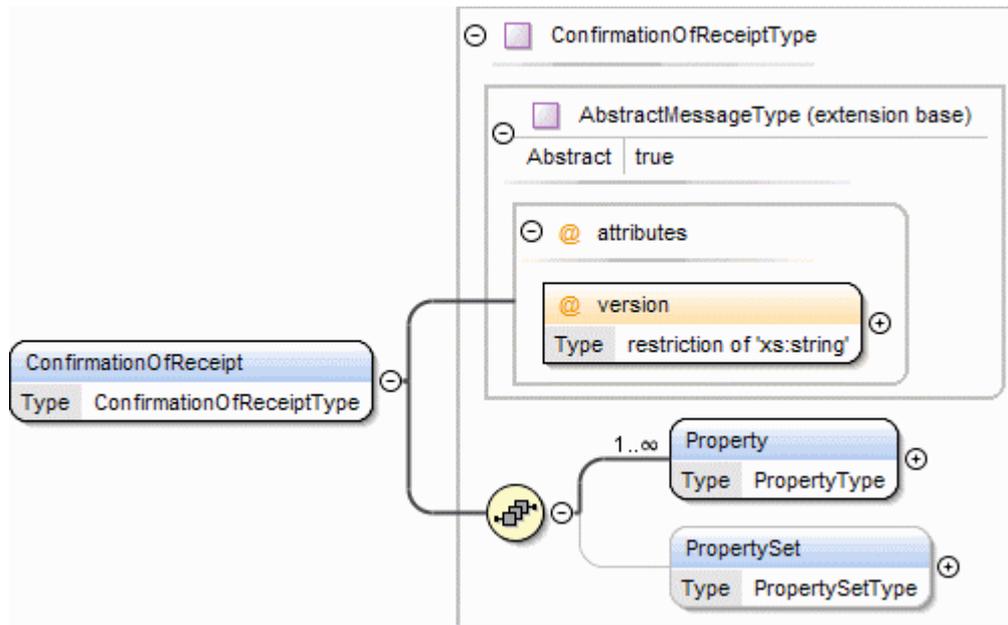
  <xmsg:PropertySet name="http://www.extra-
standard.de/property/ResponseID"
    <xmsg:Value>00008060</xmsg:Value>
    <xmsg:Value>00008062</xmsg:Value>
    <xmsg:Value>00008069</xmsg:Value>
  </xmsg:Property>

</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 1.2
<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.

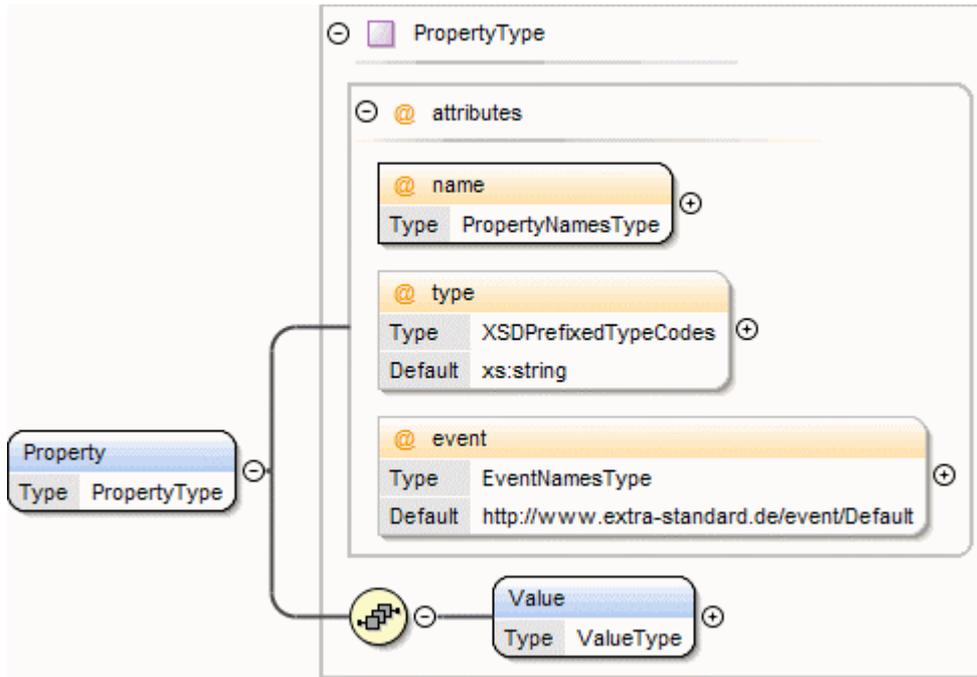
Eigenschaften werden mittels einer URI identifiziert. Für Eigenschaften, die im eXtra Kommunikationsprotokoll abgebildet werden, stehen vordefinierte URIs zur Verfügung.

### **Beispiel**

## Element Property

**xmsg:Property**

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



### Attribute

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

## 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="1.2"
  >
  <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](http://standard.de/property/ResponseID)"

event="<http://www.extra-standard.de/event/RequestData>"

<xmsg:Value>00008060</xmsg:Value>

</xmsg:Property>

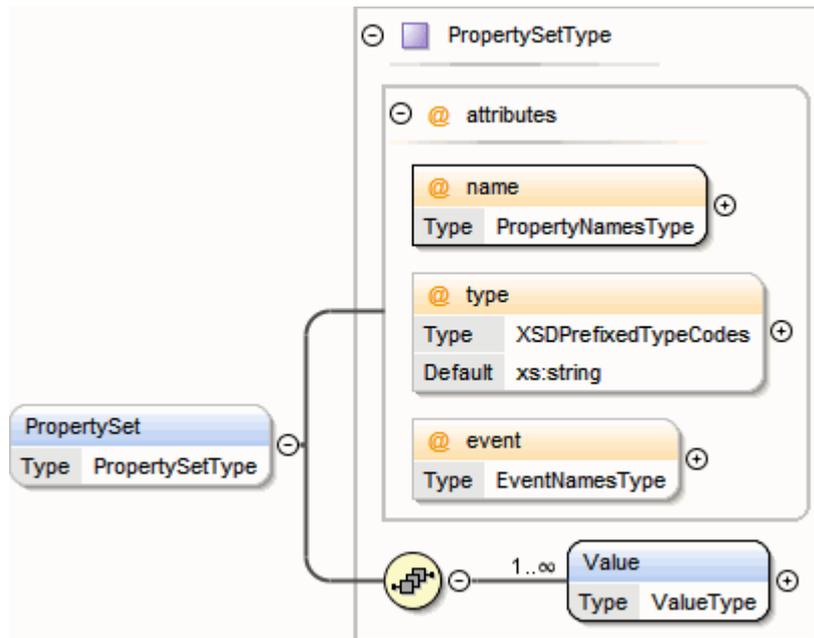
</xmsg:ConfirmationOfReceipt>

---

## 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>	-



### Attribute

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

### Beschreibung

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

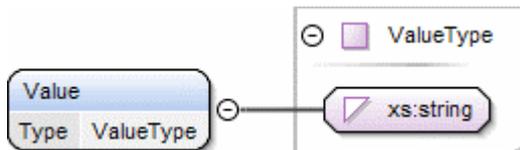


**Beispiel**

## Element Value

### xmsg:Value

<i>Inhalt</i>	Enthält den Wert einer Eigenschaft.
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	-



## Beschreibung

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

## Beispiel

## Nachrichtentyp DataRequest [normativ]

Mit Nachrichten vom Typ DataRequest ist es möglich, Anwendungsnachrichten oder Daten anhand bestimmter Kriterien abzurufen. Die Namensraum- URI ist

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

Der lokale Name des Wurzelementes ist `DataRequest`. Es wird empfohlen, den Namensraumpräfix `xmsg` für alle eXtra Standardnachrichten zu verwenden. Für die Unterscheidung dieses Nachrichtentyps von anderen desselben Namenraumes wird alternativ der Namensraumpräfix `xdrq` empfohlen.

## 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="1.1"
  >

  <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:GT>R46934503</xmsg:GT>
</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>

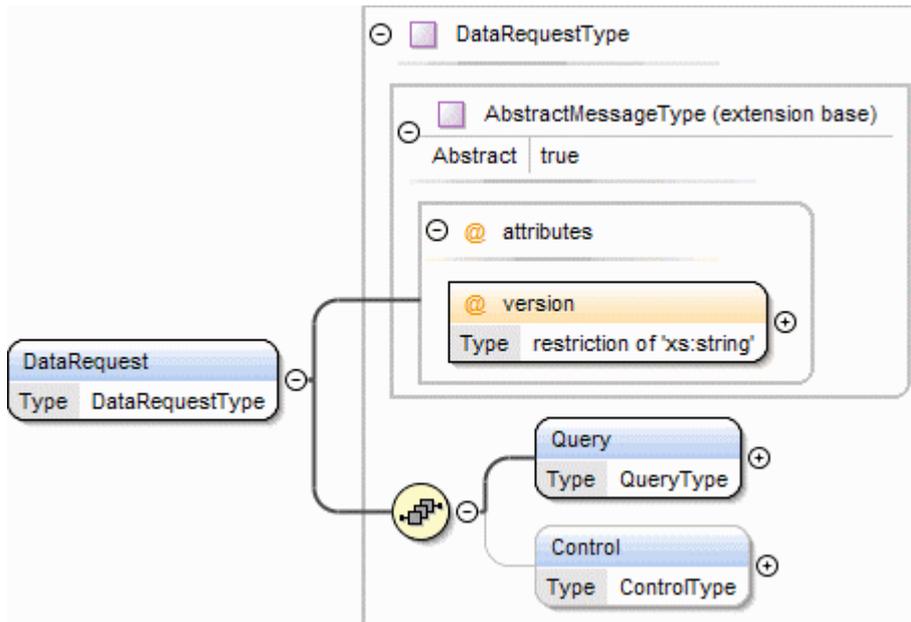
</xmsg:DataRequest>
```

---

## 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 1.1
<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.

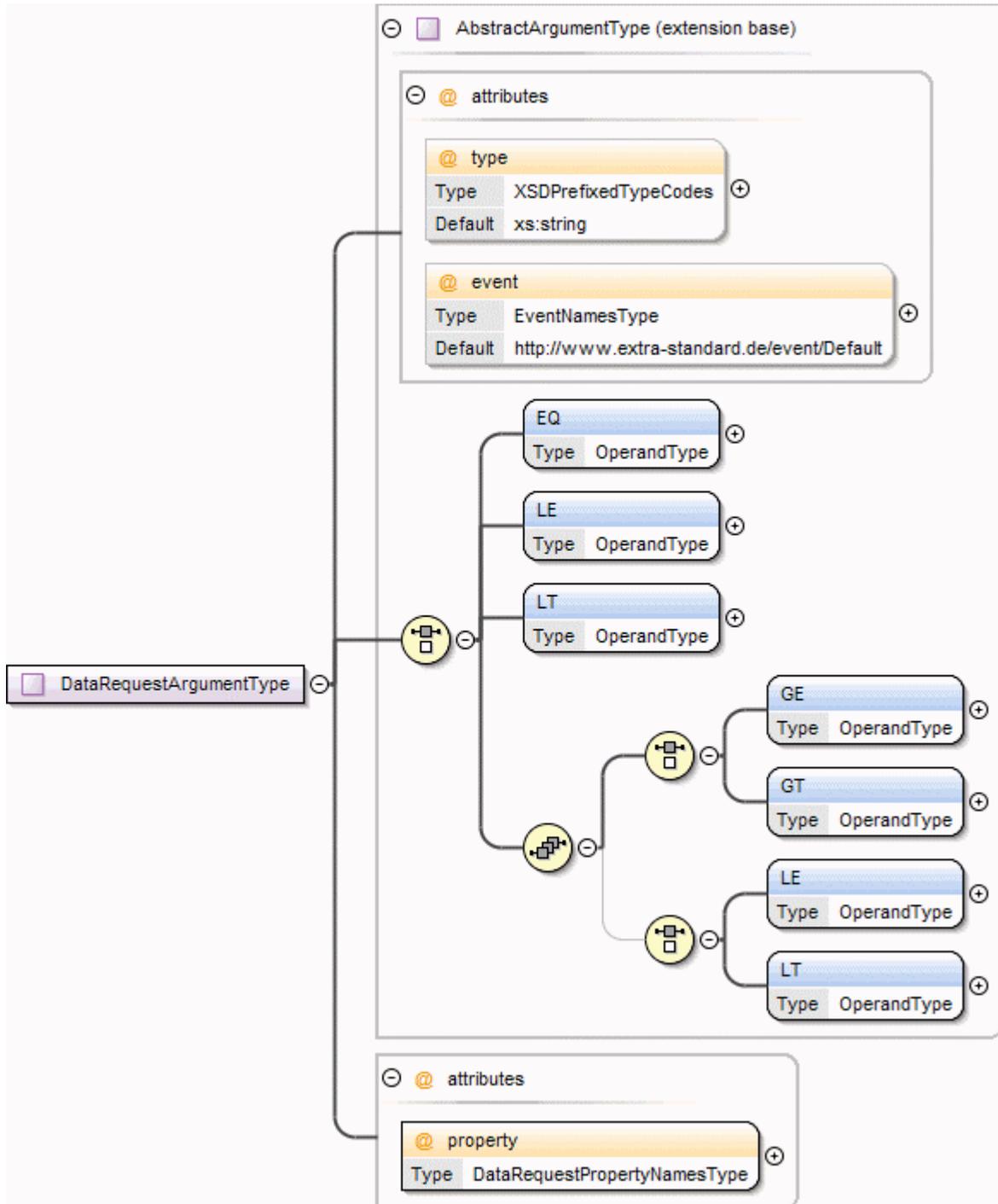
Ein Kriterium besteht aus einer abgefragten Eigenschaft (Attribut *property*), die per URI referenziert wird, und einem Wert, mit dem die entsprechende Eigenschaft der abgefragten Objekte verglichen wird. Der Wert wird als Einzelwert mit einem Kindelement oder als Wertebereich mit inklusiver Unter- und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

### **Beispiel**

## Element Argument

**xmsg:Argument**

Inhalt	Beschreibt ein Kriterium einer Abfrage
Datentyp	##elements
Benutzt von	-



Attribute

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

## Beschreibung

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.

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

```
<?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="1.1"
  >
  <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:GT>R12345000</xmsg:GT>
      <xmsg:LE>R12345100</xmsg:LE>
    </xmsg:Argument>

  </xmsg:Query>

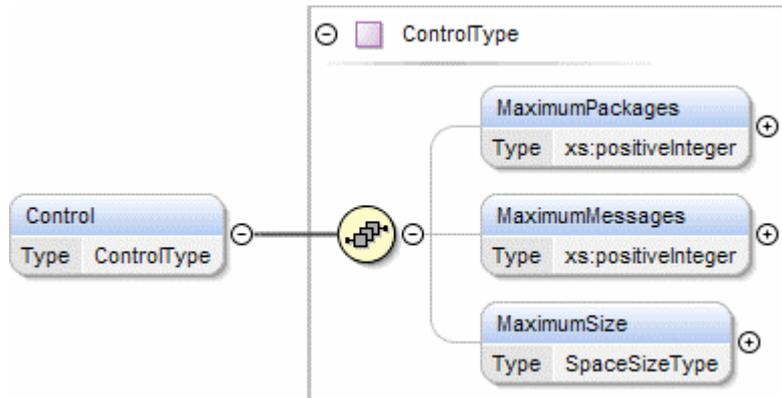
  <xmsg:Control>
    <xmsg:MaximumSize>10M</xmsg:MaximumSize>
  </xmsg:Control>

</xmsg:DataRequest>
```

## 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>	-



### 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.

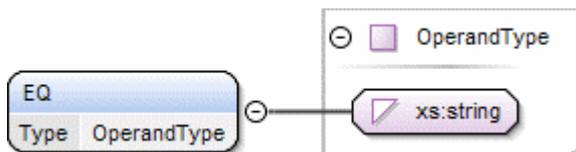
Mit dem Element `MaximumSize` wird die maximale Größe der Antwort auf die Datenanforderung im Element `TransportBody` festgelegt.

### Beispiel

## Element EQ

**xmsg : EQ**

<i>Inhalt</i>	Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der GLEICH dem angegebenen Wert ist.
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	-



### Beschreibung

Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der GLEICH dem angegebenen Wert ist.

### Beispiel

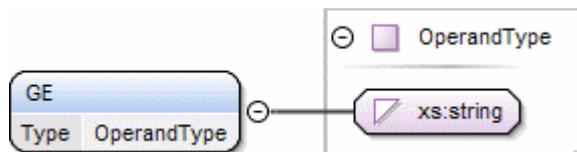
## Element GE

**xmsg : GE**

*Inhalt* Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der GRÖSSER oder GLEICH dem angegebenen Wert ist.

*Datentyp* xsd:string

*Benutzt von* -



## Beschreibung

Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der GRÖSSER oder GLEICH dem angegebenen Wert ist.

## Beispiel

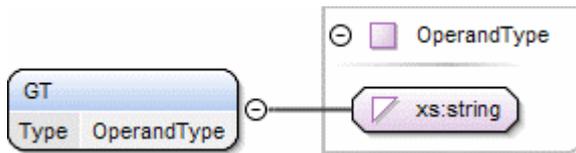
## Element GT

**xmsg : GT**

*Inhalt* Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der GRÖßER als der angegebene Wert ist.

*Datentyp* xsd:string

*Benutzt von* -



## Beschreibung

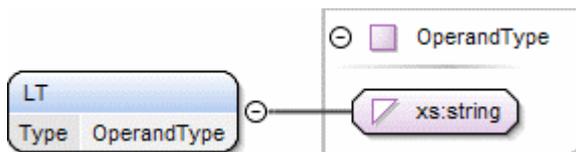
Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der GRÖßER als der angegebene Wert ist.

## Beispiel

## Element LE

**xmsg : LE**

<i>Inhalt</i>	Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der KLEINER oder GLEICH dem angegebenen Wert ist.
<i>Datentyp</i>	xsd:string
<i>Benutzt von</i>	-



## Beschreibung

Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der KLEINER oder GLEICH dem angegebenen Wert ist.

## Beispiel

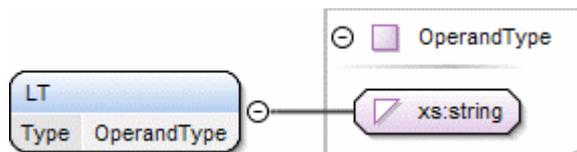
## Element *LT*

**xmsg:LT**

*Inhalt* Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der KLEINER als der angegebene Wert ist.

*Datentyp* xsd:string

*Benutzt von* -



## Beschreibung

Enthält den Wert eines Kriteriums und gibt an, dass der Vergleich typgerecht in der Weise erfolgen soll, dass jeder Vergleichswert das Kriterium befriedigt, der KLEINER als der angegebene Wert ist.

## Beispiel

## ***Element MaximumMessages***

### **xmsg:MaximumMessages**

<i>Inhalt</i>	Gibt an, wieviele Anwendungsnachrichten maximal übertragen werden sollen, um die Anforderung zu befriedigen.
<i>Datentyp</i>	xsd:positiveInteger
<i>Benutzt von</i>	-



### **Beschreibung**

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

### **Beispiel**

## Element *MaximumPackages*

### **xmsg:MaximumMessages**

<i>Inhalt</i>	Gibt an, wieviele Pakete maximal übertragen werden sollen, um die Anforderung zu befriedigen.
<i>Datentyp</i>	xsd:positiveInteger
<i>Benutzt von</i>	-



### **Beschreibung**

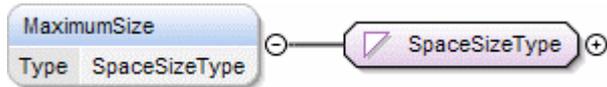
xmsg:MaximumPackages gibt an, aus wieviel Paketen eine Antwort maximal bestehen darf, um die Anforderung zu befriedigen.

### **Beispiel**

## Element *MaximumSize*

### **xmsg:MaximumSize**

<i>Inhalt</i>	Gibt an, wie groß die Antwort maximal sein darf, um die Anforderung zu befriedigen.
<i>Datentyp</i>	xmsg:SpaceSizeType
<i>Benutzt von</i>	-



## 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:

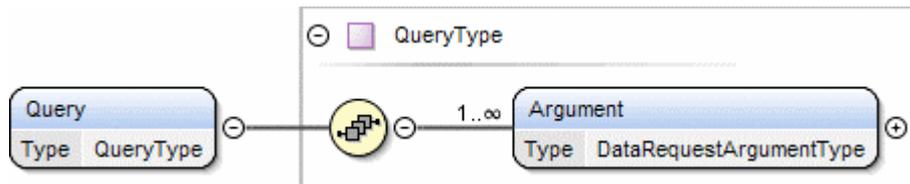
B	Byte
K	Kilobyte
M	Megabyte
G	Gigabyte

## Beispiel

## Element Query

**xmsg:Query**

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



### Beschreibung

Beschreibt eine Abfrage als ein Set von Kriterien (Element `Argument`).

### Beispiel

## Nachrichtentyp **ListOfConfirmationOfReceipt** [normativ]

Eine Nachricht dieses Typs dient zur Übertragung einer Menge von Nachrichten des Typs *ConfirmationOfReceipt*. Die Namensraum-URI ist

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

Der lokale Name des Wurzelementes ist *ConfirmationOfReceipt*. Es wird empfohlen, den Namensraumpräfix *xmsg* für alle eXtra Standardnachrichten zu verwenden. Für die Unterscheidung dieses Nachrichtentyps von anderen desselben Namensraumes wird alternativ der Namensraumpräfix *xcor* empfohlen.

## ***Beispieldokument***

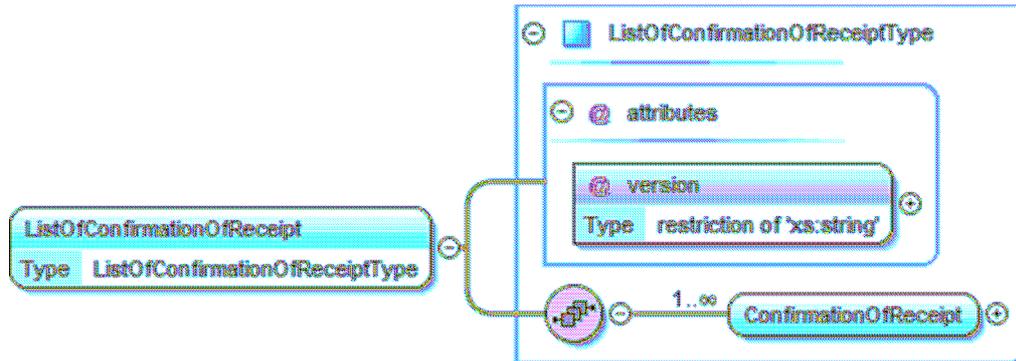
*Beispiel wird noch ergänzt.*

---

## 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 1.0
<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*. Die Namensraum-URI ist

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

Der lokale Name des Wurzelementes ist *ConfirmationOfReceipt*. Es wird empfohlen, den Namensraumpräfix *xmsg* für alle eXtra Standardnachrichten zu verwenden. Für die Unterscheidung dieses Nachrichtentyps von anderen desselben Namensraumes wird alternativ der Namensraumpräfix *xcor* empfohlen.

## ***Beispieldokument***

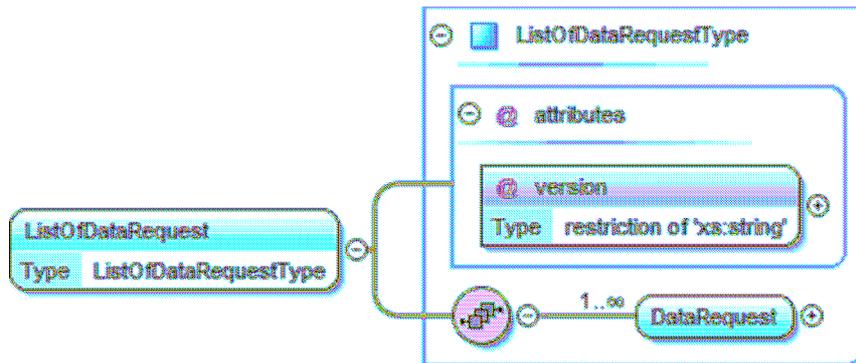
*Beispiel wird noch ergänzt.*

---

## 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 1.0
<i>Datentyp</i>	xsd:string

### 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*

## 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 <a href="http://www.extra-standard.de">www.extra-standard.de</a>
EINF	<i>Einführung in den eXtra Standard</i> , zu finden unter <a href="http://www.extra-standard.de">www.extra-standard.de</a>
EMSG	<i>eXtra Standardnachrichten, Schnittstellenbeschreibung</i> , zu finden unter <a href="http://www.extra-standard.de">www.extra-standard.de</a>
IFACE	<i>eXtra Transport Schnittstellenbeschreibung</i> , zu finden unter <a href="http://www.extra-standard.de">www.extra-standard.de</a>
IMPL	<i>eXtra Implementierung</i>
KOMP	<i>eXtra Kompendium</i> , zu finden unter <a href="http://www.extra-standard.de">www.extra-standard.de</a>
RFC2119	<i>Request for Comments: 2119</i> , S. Bradner, Harvard University, March 1997, <a href="http://www.ietf.org/rfc/rfc2119.txt">http://www.ietf.org/rfc/rfc2119.txt</a>
PROF	<i>eXtra Profilierung</i> , zu finden unter <a href="http://www.extra-standard.de">www.extra-standard.de</a>
XENC	<i>XML Encryption</i> , <a href="http://www.w3.org/TR/xmlenc-core/">http://www.w3.org/TR/xmlenc-core/</a>
XML	<i>XML Recommendation 1.0, 3rd Edition</i> , <a href="http://www.w3.org/XML">http://www.w3.org/XML</a>
XSD	<i>XML Schema Definition</i> , <a href="http://www.w3.org/TR/xmlschema-0/">http://www.w3.org/TR/xmlschema-0/</a>
XSIG	<i>XML Signature</i> , <a href="http://www.w3.org/TR/xmldsig-core/">http://www.w3.org/TR/xmldsig-core/</a>
XSL	<i>XML Stylesheet Language</i> , <a href="http://www.w3.org/TR/1999/REC-xslt-19991116">http://www.w3.org/TR/1999/REC-xslt-19991116</a> , <a href="http://www.w3.org/TR/xslt20/">http://www.w3.org/TR/xslt20/</a>