

Mail.XML Version 26.1

System Messages Specification

Monday, April 3, 2023

Working Group Chair

Shawn Baldwin, WindowBook

Technical Director

Shariq Mirza, DTAC Associate, Assurety Consulting & Solutions

Editor

Shariq Mirza, DTAC Associate, Assurety Consulting & Solutions

Copyright (c) 2022 – Delivery Technology Advocacy Council (“DTAC”). All Rights Reserved.

Mail.dat is a registered trademark of DTAC

Mail.XML is a trademark of DTAC



Copyright and Legal Notices

© 2022 Delivery Technology Advocacy Council. All Rights Reserved.

Copyright 2022 – Delivery Technology Advocacy Council (“DTAC”) is the “Copyright Owner” of “Mail.XML®”. All rights reserved by the Copyright Owner under the laws of the United States, Belgium, the European Economic Community, and all states, domestic and foreign. This document may be downloaded and copied provided that all copies retain and display the copyright and any other proprietary notices contained in this document. This document may not be sold, modified, edited, or taken out of context such that it creates a false or misleading statement or impression as to the purpose or use of the Mail.XML® specification, which is an open standard. Use of this Standard, in accord with the foregoing limited permission, shall not create for the user any rights in or to the copyright, which rights are exclusively reserved to the Copyright Owner.

DTAC and the members of the Mail.XML® Specifications - Committee (collectively and individually, "Presenters") make no representations or warranties, express or implied, including, but not limited to, warranties of merchantability, fitness, for a particular purpose, title, or non-infringement. The presenters do not make any representation or warranty that the contents of this document are free from error, suitable for any purpose of any user, or that implementation of such contents will not infringe any third-party patents, copyrights, trademarks or other rights. By making use of this document, the user assumes all risks and waives all claims against Presenters.

In no event shall Presenters be liable to user (or other person) for direct, indirect, special or consequential damages arising from or related to any use of this document, including, without limitation, lost profits, business interruption, loss of programs, or other data on your information handling system even if Presenters are expressly advised of the possibility of such damages.

Some states do not allow the disclaimer or limitation of damages, so the disclaimers set forth above apply to the maximum extent permitted under applicable law.

Abstract

This document describes the messaging protocol for use by mailers and their consignees. The Mail.XML™ Transaction Protocol defines the roles and responsibilities of Shippers and Consignees and defines the format and method for message exchange. This messaging protocol is designed to be XML and Web-Services compliant.

Mail.XML and Mail.dat are trademarks of DTAC.

About Mail.XML™

Mail.XML™ is bringing a paradigm change to the industry by increasing business function specific B2B (Business to Business) communication within the industry that supports automation and in the end enables cost avoidance and higher profits through improved competence and effectiveness of communication. Mail.XML is designed to increase efficiency and lower costs by removing many manual data entry processes and enabling quick near real time communication between business partners. Mail.XML currently supports container-based scheduling, pick up and drop off business processes, as well as identifying different business entities responsible for performing different services such as quality of mailing, address correction, and delivery confirmation on a mailing. The core focus of Mail.XML is communication between industry members and from industry to the final mail processing and delivery organization that delivers the mail to the end consumer, e.g., USPS. In the next few versions of Mail.XML the focus moves across mailing supply chain channels, and includes advanced functions such as payment; automated verification; enabling first, second, and third-party communication and incorporating presort planning, printing, and distribution processes.

What's New in Mail.XML Version 26.1?

With this release, the Mail.XML Messaging Protocol moves to Version 26.1. This release supports structure changes required by mailing industry and Postal Service.

Changes supported by Mail.XML 26.1 include:

- 2613 - Proposal to add support for Marriage Mail product
- 2612 - Proposal to add support for USPS Ground Advantage product
- 2614 - Proposal to support/Clarify Destination Hub for Parcel Select Destination Entry

About Mail.XML Schema Modularization

Today Mail.XML messages are grouped into 8 message types.

- Transportation Messages (TM)
- Mailing Messages (MM)
- Data Distribution Messages (DD)
- Dynamic Payment Template Messages
- Identification Messages (ID)
- Supply Chain Messages (SC)
- Informed Visibility (IV)
- System Messages
- Base: Shared simple types
- Definitions: Shared complex types and elements

The simple types shared across 2 or more modules are found in the Base schema. Likewise, the shared definitions module contains complex type definitions and elements that are shared across 2 or more modules.

Mail.XML Module Versioning Rules

The following versioning rules will be followed:

The Mail.XML wrapper schema^{**}.xsd) will always be given the next higher version number (or Errata designation) when any update is made to base, defs or any module. The name of the .xsd file will indicate the new version and the new version number will be used in the namespace and target declarations:
xmlns:mailxml="http://delivery-tech.org/Specs/mailxml26.1/mailxml"
targetNamespace="http://delivery-tech.org/Specs/mailxml26.1/mailxml"

- When updates are made, only those modules that are updated will be given the next higher version number (or Errata letter designation).
- If updates are made to the base or defs, then the base and defs xsds will be given the next higher version number (or Errata designation) and all modules that call to them will also be given the next higher version number (or Errata designation).

For example:

- If the wrapper version is labeled as xmlns:mailxml="http://deliverytech.org/Specs/mailxml26.1A/mailxml" then at least one of the XSDs is at same version such as filename = 'Mail.XML_26.1A.xsd' <- Errata A
- If the wrapper version is labeled as xmlns:mailxml="http://deliverytech.org/Specs/mailxml26.1B/mailxml" then at least one of the XSDs is at same version such as filename = 'Mail.XML_26.1B.xsd' <- Errata B

- If the wrapper version is labeled as xmlns:mailxml="http://deliverytech.org/Specs/mailxml26.1/mailxml" then at least one of the XSDs is at same version such as filename = 'Mail.XML_26.1.xsd' <- Major Version

Mail.XML 26.1 XSD Modules

The following Mail.XML XSD modules/namespaces are used:

- Mail.XML_tm.xsd: This module contains all the transportation (or FAST) messages and the attributes, elements and complex types that are unique to these messages. Namespace=Mail.XML_tm:
- Mail.XML_mm.xsd: This module contains all the mailing messages (eDoc) and the attributes, elements and complex types that are unique to these messages. Namespace=Mail.XML_mm:
- Mail.XML_iv.xsd: This module contains informed visibility messages and the attributes, elements and complex types that are unique to these messages. Namespace=Mail.XML_iv:
- Mail.XML_dd.xsd: This module contains all the data distribution messages and the attributes, elements and complex types that are unique to these messages. Namespace=Mail.XML_dd:
- Mail.XML_id.xsd: This module contains all the identification messages and the attributes, elements and complex types that are unique to these messages. Namespace=Mail.XML_id:
- Mail.XML_sc.xsd: This module contains all the supply chain messages and the attributes, elements and complex types that are unique to these messages. Namespace=Mail.XML_sc:
- Mail.XML_defs.xsd: This module contains all the common definitions of attributes, elements and complex types that are used across two or more message types. Namespace=Mail.XML_defs:
- Mail.XML_base.xsd: This module contains simple types that are shared across two or more modules that make up Mail.XML. These can be considered a building block for any message group. Namespace=Mail.XML_base:
- Mail.XML.xsd: This module contains the system messages of Mail.XML and is used to build custom profiles for Mail.XML. Namespace=Mail.XML:

The Mail.XML™ 26.1 Messaging Documentation Set

The Mail.XML Messaging Specification has been organized into a set of documents. This *Schemas Specification* is one document in a set of documents that make up the Mail.XML Specification 26.1. Updates in this Specification are NOT backwardly compatible with previous versions. Other documents in the specification set include:

- Mail.XML™ 26.1: Transportation Messaging Specification documents all transportation messages
- Mail.XML™ 26.1: Mailing Messaging Specification documents all mailing messages
- Mail.XML™ 26.1: Informed Visibility Specification documents all informed visibility messages
- Mail.XML™ 26.1: Data Distribution Messaging Specification documents all data distribution messages
- Mail.XML™ 26.1: Identification Messaging Specification documents all identification messages
- Mail.XML™ 26.1: Supply Chain Messaging Specification documents all supply chain messages
- Mail.XML™ 26.1: System Messaging Specification documents all systems and fault messages
- Mail.XML™ 26.1: Simple Types Specification documents all simple types used across Mail.XML

- messages
- Mail.XML™ 26.1: Common Definitions Specification documents all shared elements and complex types.
- Mail.XML™ 26.1: Schemas contains the .XSDs that make up the Mail.XML Messaging Specification

Table of Contents

Abstract	3
About Mail.XML™	3
What's New in Mail.XML Version 26.1?	3
About Mail.XML Schema Modularization	4
Mail.XML Module Versioning Rules	4
Mail.XML 26.1 XSD Modules	5
The Mail.XML™ 26.1 Messaging Documentation Set	5
Schema mailxml_defs_26.1.xsd	8

Schema mailxml_defs_26.1.xsd

schema location: [../XSDs/mailxml_defs_26.1.xsd](http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs)
attribute form default: **qualified**
element form default: **qualified**
targetNamespace: **http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs**

Elements

[ContainerDetailData](#)
[ContainerInfoData](#)
[DataRecipient](#)
[ExtraServiceData](#)
[LargeTransactionDivider](#)
[LargeTransactionDividerResult](#)
[QueryError](#)

Complex types

[addressType](#)
[basicReturnInfoType](#)
[commonContactType](#)
[consigneeFacilityType](#)
[consolidatorCommunicationInfoType](#)
[contactIDType](#)
[ContainerDiscrepancyType](#)
[containerErrorWarningBlockType](#)
[containerIDType](#)
[ContainerInductedType](#)
[containerInfoDataType](#)
[containerKeysInfoType](#)
[containerPostInductionInfoType](#)
[containerPreInductionInfoType](#)
[containerReleaseInfoType](#)
[containerStatusInfoType](#)
[contentIDType](#)
[CSQContainerDetailDataType](#)
[CSQcontainerInfoDataType](#)
[CSQlinkingContainerIDType](#)
[documentVersionDataforCSQType](#)
[errorWarningType](#)
[fullContainerIDType](#)
[gPSCoordinates](#)
[IMcbAndIMtbPieceScanInfoType](#)
[IMcbPieceScanInfoType](#)
[IMtbPieceScanInfoType](#)
[inductionCloseoutInfoType](#)
[inductionIssuesType](#)
[InductionProblemType](#)
[intelligentMailBarcodeType](#)
[intelligentMailPackageBarcodeType](#)
[maildatContainerIDType](#)
[MaildatMPUInfoType](#)
[mailPieceIDType](#)
[MailXMLContainerIDType](#)
[mailxmlDetailType](#)
[measurementType](#)
[MIDType](#)

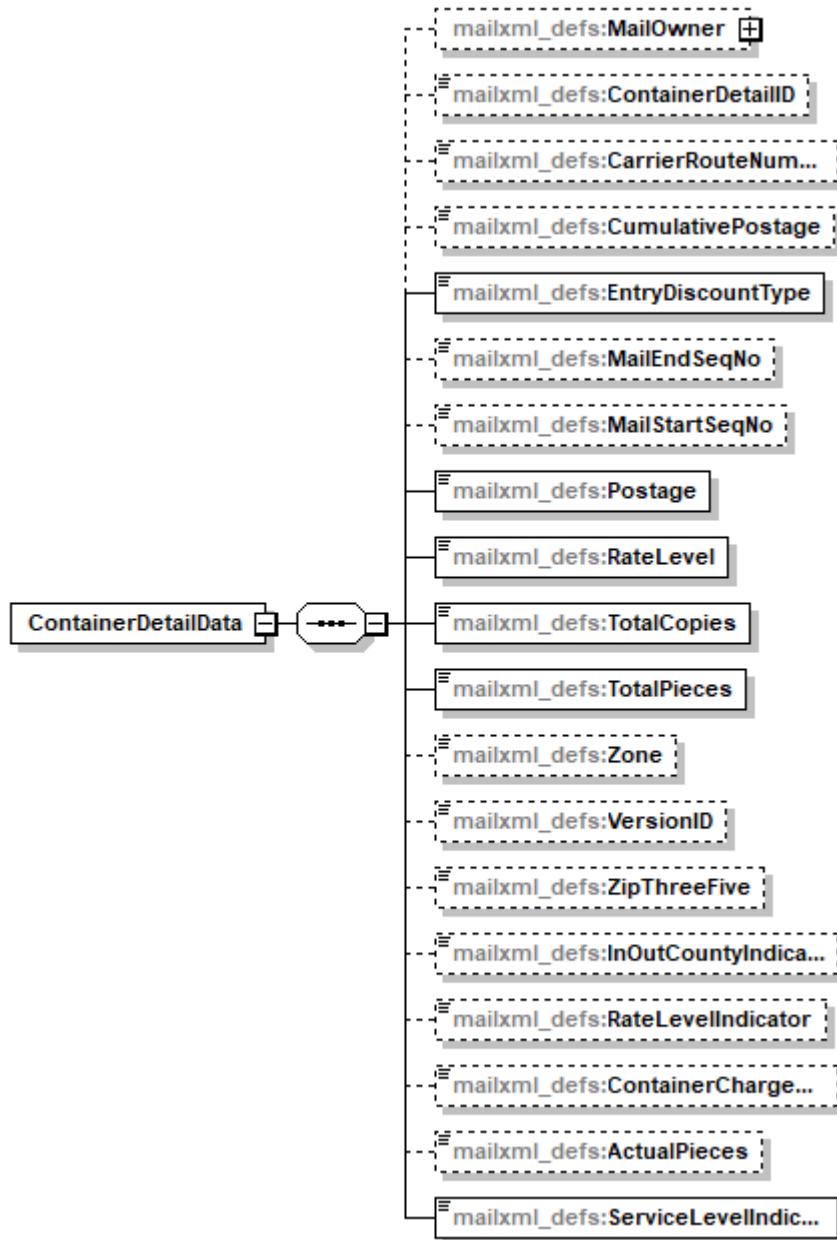
Simple types

[bundleScanTypeType](#)
[containerDiscrepancyCategoryType](#)
[containerScanStateType](#)
[containerStatusType](#)
[countTypeType](#)
[einductionDataSourceType](#)
[fullServiceComplianceIndicatorType](#)
[MPSQueryType](#)
[MPSSStateType](#)
[pieceScanEventTypeType](#)
[problemCategoryType](#)
[problemTypeType](#)
[reasonCodeType](#)
[retrieveDataByType](#)

[MPSRBlockType](#)
[MPSVisIncludedInScanRecFlagType](#)
[MPSVisResultOptionsType](#)
[palletHandoffInfoType](#)
[participantIDType](#)
[permitPublicationDataType](#)
[pickupApptBlockResponseType](#)
[postageStatementSummaryType](#)
[PSRBlockType](#)
[queryErrorType](#)
[returnInfoType](#)
[scanEventType](#)
[submittingSoftwareType](#)
[zipCode](#)

element **ContainerDetailData**

diagram



namespace http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

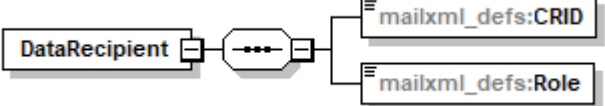
element ContainerInfoData

diagram

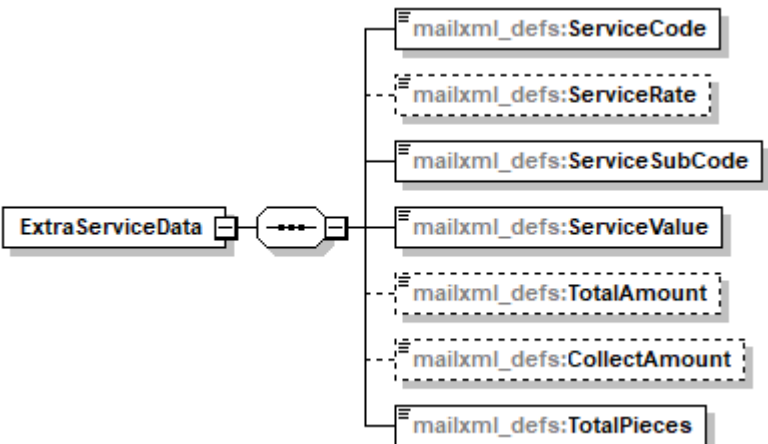


namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---


element **DataRecipient**

diagram	 <pre> graph LR DR[DataRecipient] --- C1(()) C1 --- CRID[mailxml_defs:CRID] C1 --- Role[mailxml_defs:Role] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

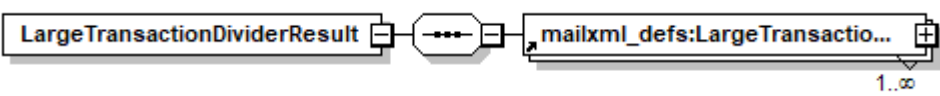
element **ExtraServiceData**

diagram	 <pre> graph LR ESD[ExtraServiceData] --- C1(()) C1 --- SC[mailxml_defs:ServiceCode] C1 --- SR[mailxml_defs:ServiceRate] C1 --- SSC[mailxml_defs:ServiceSubCode] C1 --- SV[mailxml_defs:ServiceValue] C1 --- TA[mailxml_defs:TotalAmount] C1 --- CA[mailxml_defs:CollectAmount] C1 --- TP[mailxml_defs:TotalPieces] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs


element **LargeTransactionDivider**

diagram	 <pre> graph LR LTD[LargeTransactionDivider] --- C1(()) C1 --- TID[mailxml_defs:TrackingID] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

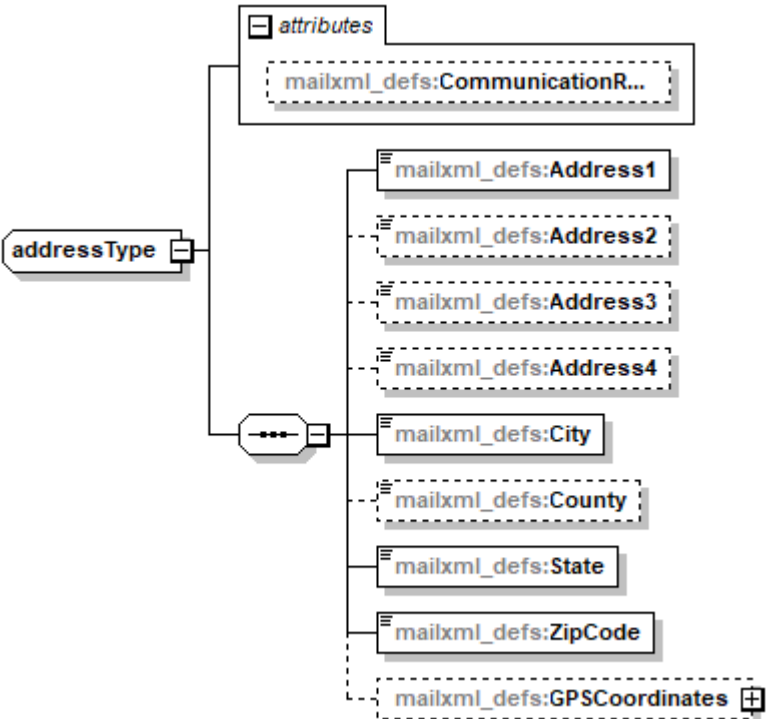
element **LargeTransactionDividerResult**

diagram	 <pre> graph LR LTRD[LargeTransactionDividerResult] --- C1(()) C1 --- LTR[mailxml_defs:LargeTransactio...] LTR -- 1..∞ --> Plus[+] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

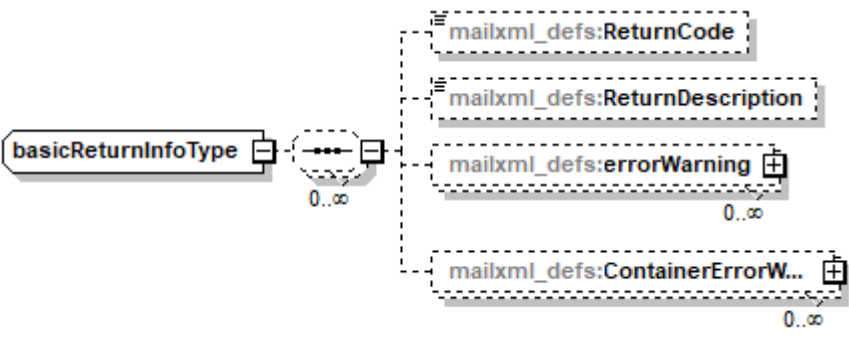
element **QueryError**

diagram	 <p>The diagram shows the QueryError element connected to the mailxml_defs:ReturnInfo element via a sequence connector. Below the diagram, the text reads: "Error issued when the query data cannot be provided."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
annotation	documentation Error issued when the query data cannot be provided.

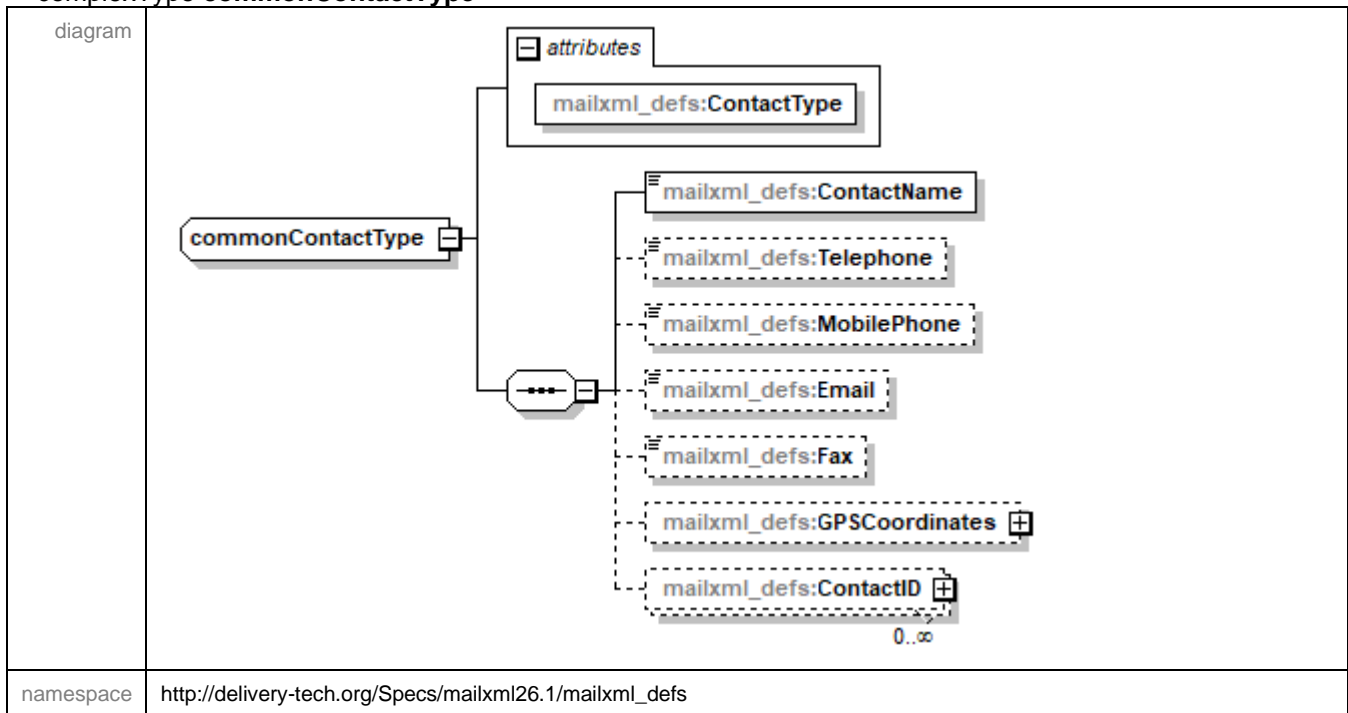
complexType **addressType**

diagram	 <p>The diagram shows the addressType complexType. It has an attributes section containing mailxml_defs:CommunicationR.... The main body of the complexType is a sequence of elements: mailxml_defs:Address1, mailxml_defs:Address2, mailxml_defs:Address3, mailxml_defs:Address4, mailxml_defs:City, mailxml_defs:County, mailxml_defs:State, mailxml_defs:ZipCode, and mailxml_defs:GPSCoordinates. The GPSCoordinates element is optional, indicated by a dashed border and a plus icon.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

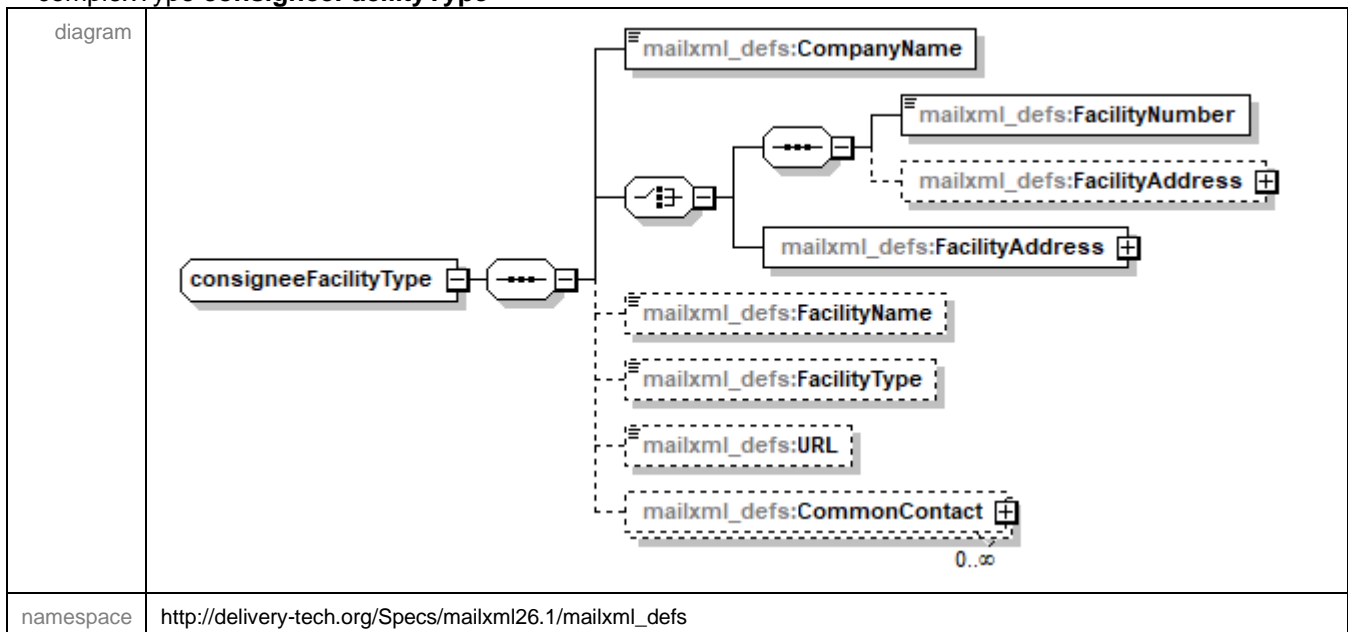
complexType **basicReturnInfoType**

diagram	 <p>The diagram shows the basicReturnInfoType complexType. It contains a sequence of elements: mailxml_defs:ReturnCode, mailxml_defs:ReturnDescription, mailxml_defs:errorWarning, and mailxml_defs:ContainerErrorW.... The errorWarning and ContainerErrorW... elements are optional, indicated by dashed borders and plus icons. Multiplicity values of 0..∞ are shown for the optional elements.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **commonContactType**



complexType **consigneeFacilityType**



complexType consolidatorCommunicationInfoType

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

complexType **contactIDType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

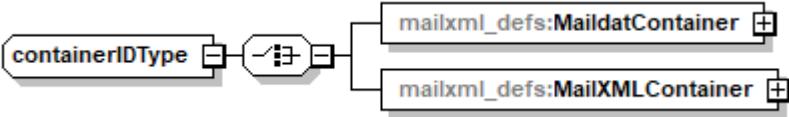
complexType **ContainerDiscrepancyType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

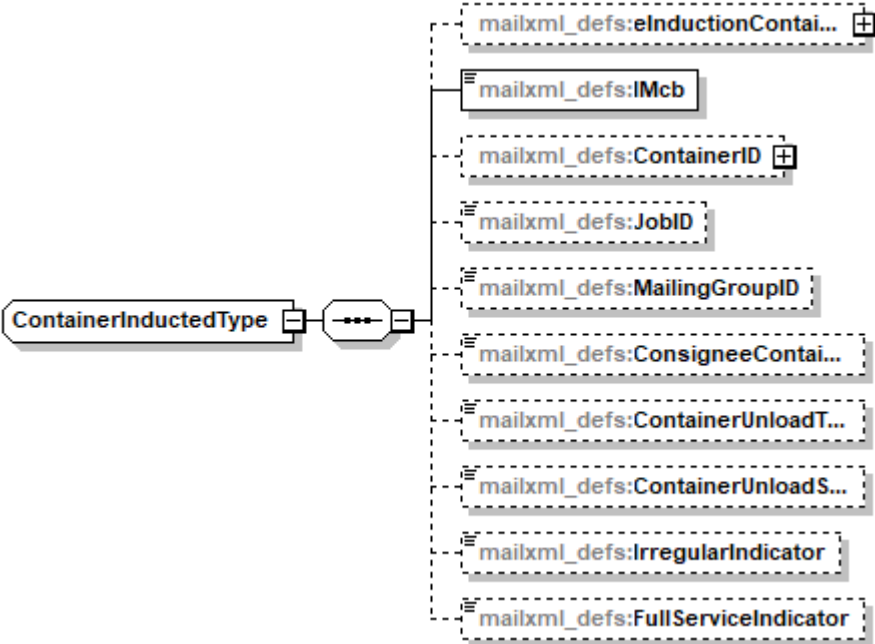
complexType **containerErrorWarningBlockType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType containerIDType

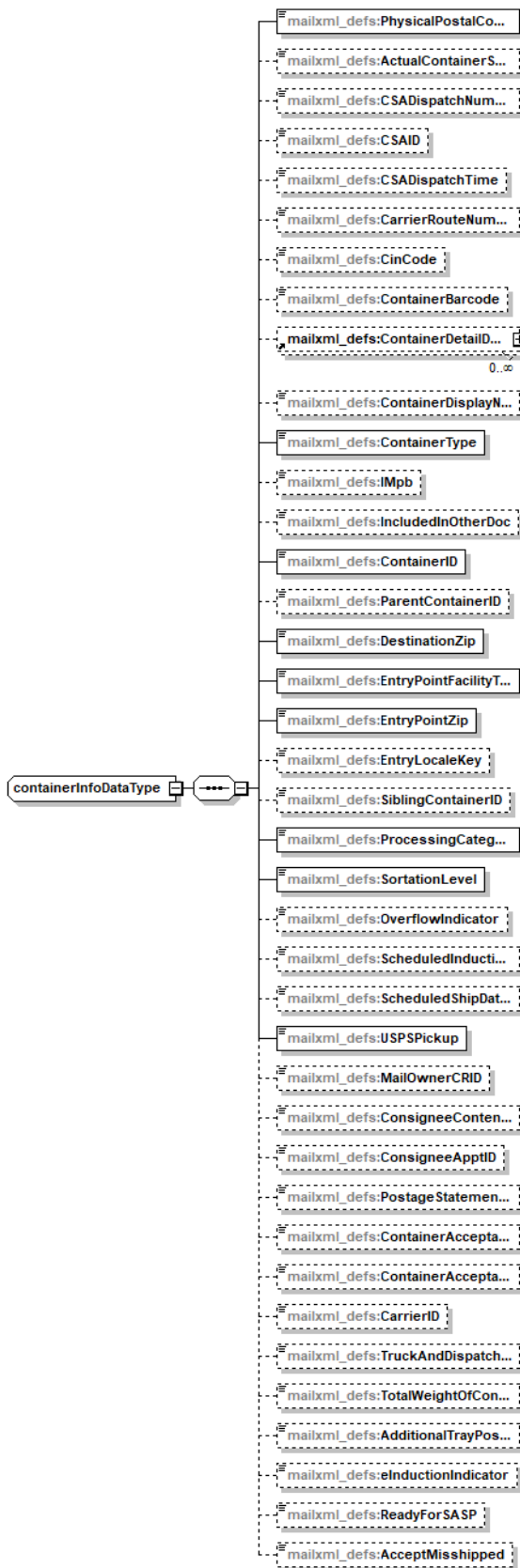
diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType ContainerInductedType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **containerInfoDataType**

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

complexType containerKeysInfoType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType containerPostInductionInfoType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType containerPreInductionInfoType

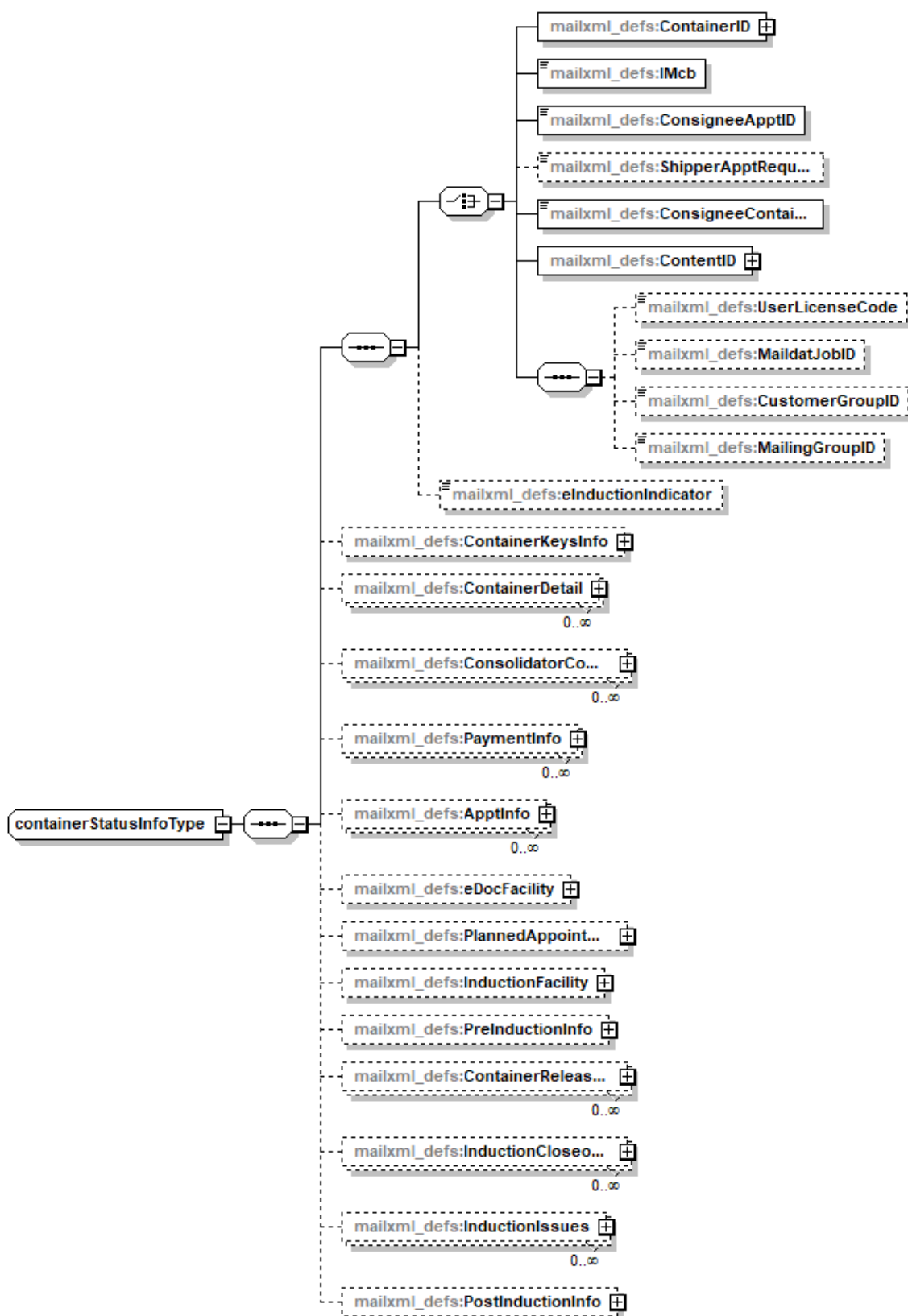
diagram	<pre> xsd:sequence base="containerPreInductionInfoType" xmlns:mailxml_defs="http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs"> mailxml_defs:BarcodeFormatVal... mailxml_defs:UniqueBarcodeVal... mailxml_defs:EPDValidation mailxml_defs:SingleAppointme... mailxml_defs:IMcbToAppointme... mailxml_defs:PaymentValidation mailxml_defs:LabelingListValida... mailxml_defs:TransportedMailA... mailxml_defs:ContainerManifes... mailxml_defs:NewZoneValidation mailxml_defs:PreInductionStatus mailxml_defs:ReturnInfo? </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType containerReleaseInfoType

diagram	<pre> xsd:sequence base="containerReleaseInfoType" xmlns:mailxml_defs="http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs"> mailxml_defs:TransactionID mailxml_defs:MessageGenerat... mailxml_defs:MessageSentTarg... mailxml_defs:MessageType </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType containerStatusInfoType

diagram

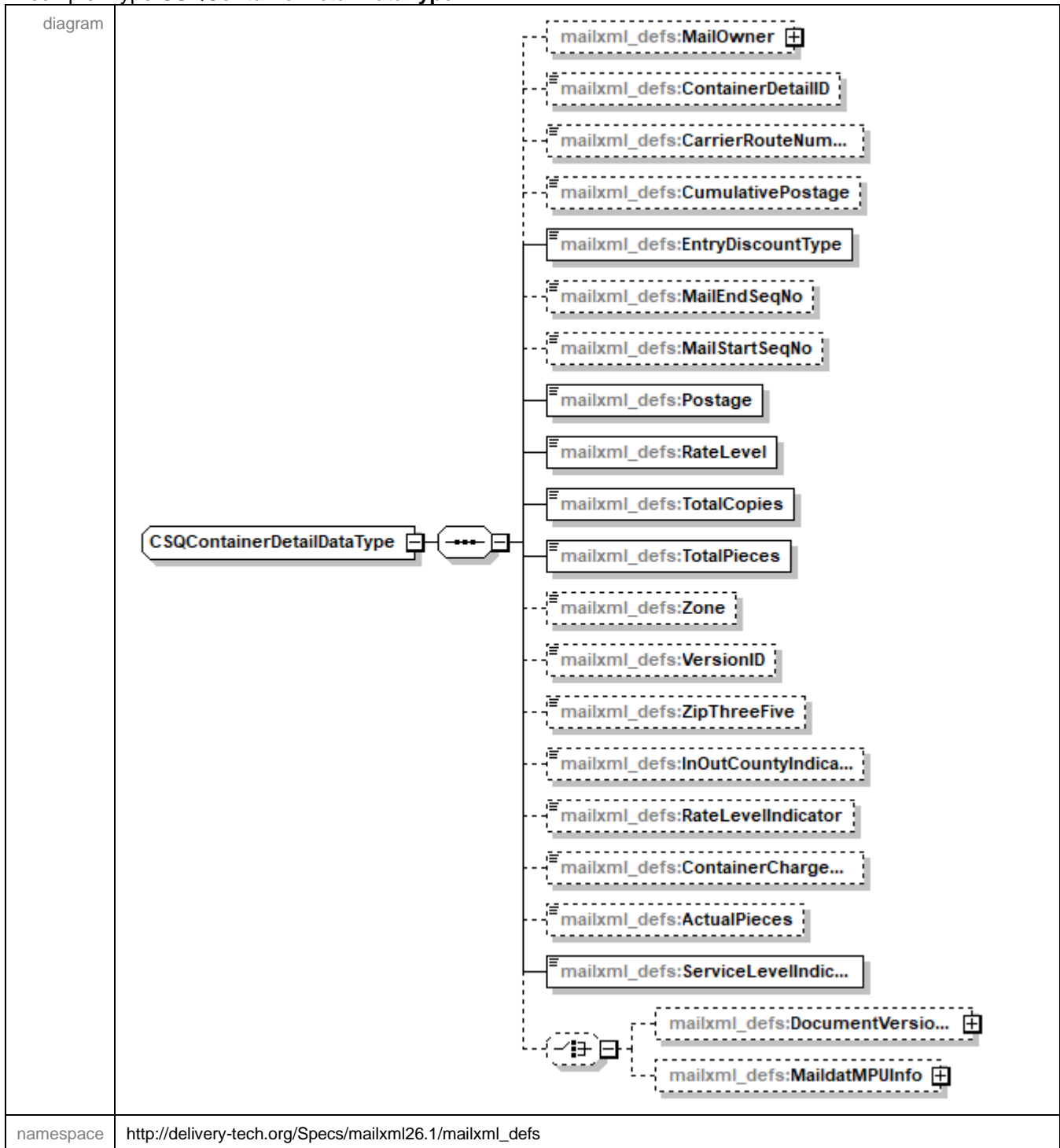


namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

complexType **contentIDType**

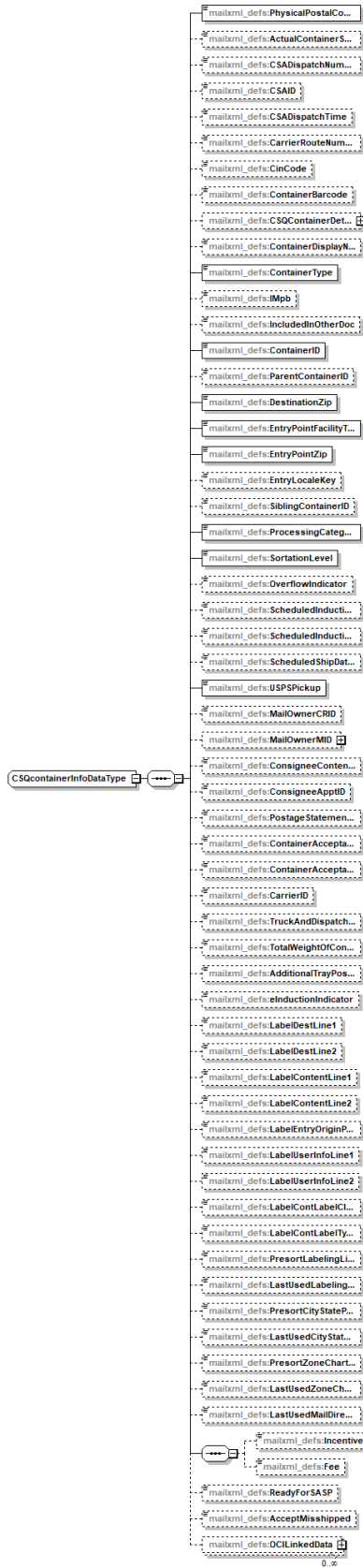
diagram	<pre> classDiagram class contentIDType { +attributes +mailxml_defs:ConsigneeConten... +mailxml_defs:SchedulerContent... } </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType CSQContainerDetailDataType



complexType **CSQcontainerInfoDataType**

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

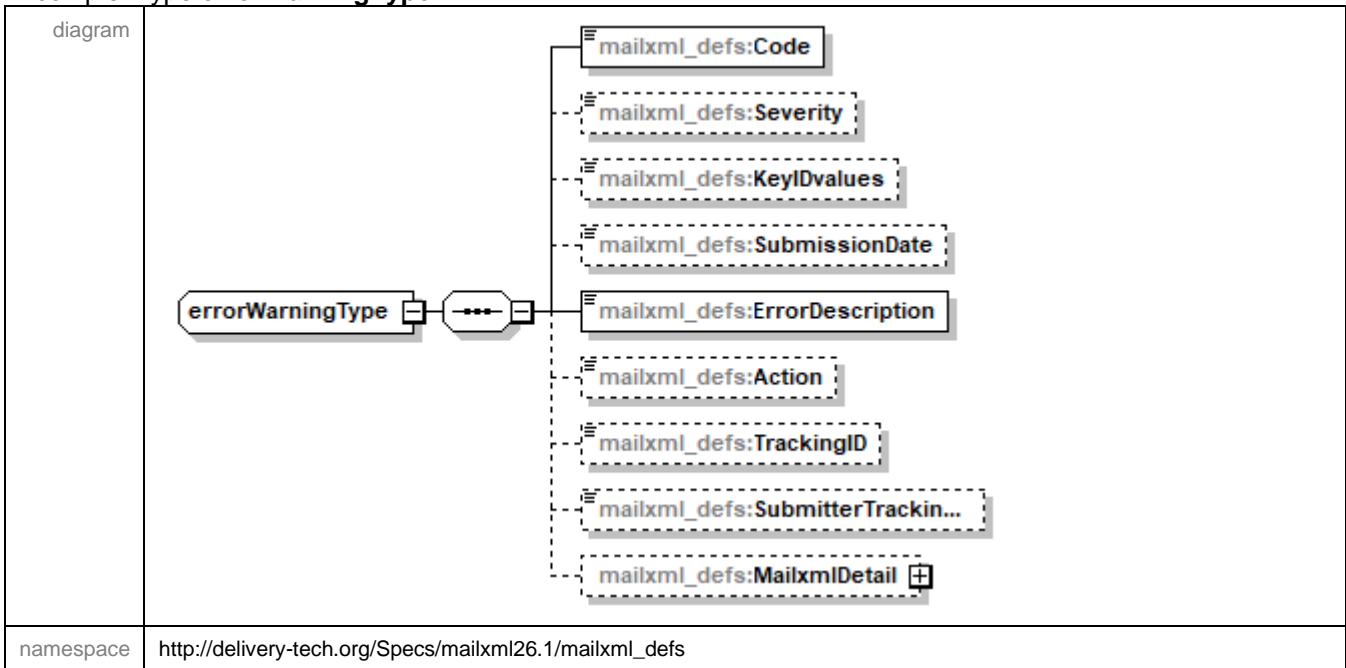
complexType **CSQlinkingContainerIDType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

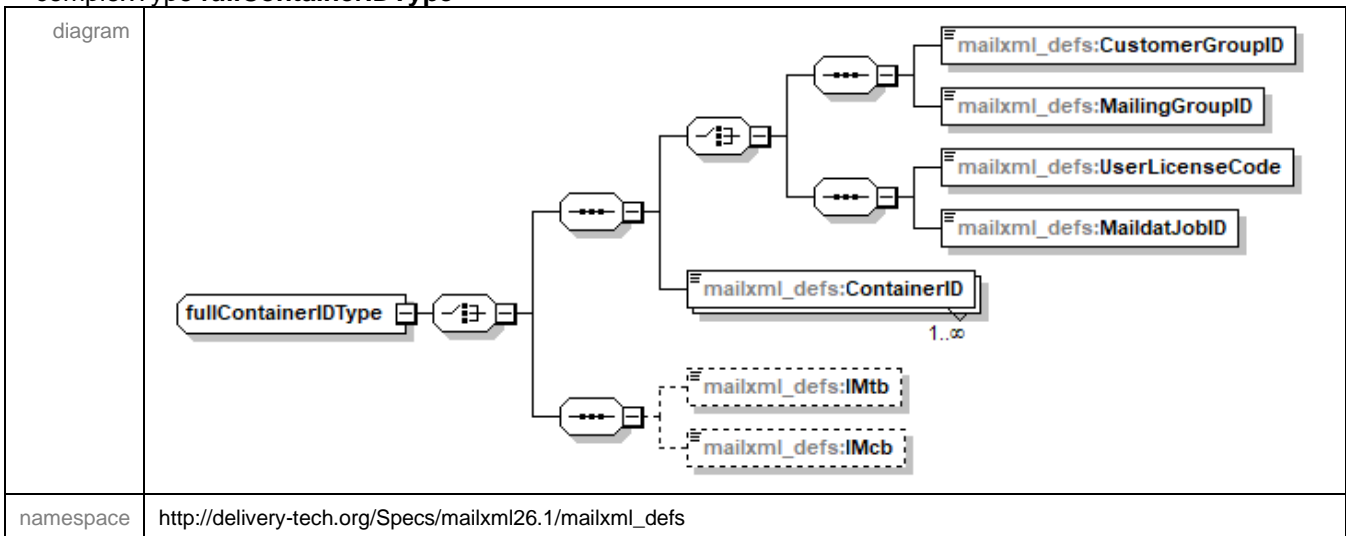
complexType **documentVersionDataforCSQType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType errorWarningType



complexType fullContainerIDType



complexType **gPSCoordinates**

diagram	<pre> xsd:complexType base="base" name="gPSCoordinates"> <div> attributes <div> mailxml_defs:GPSSystem </div> </div> <div> mailxml_defs:Latitude mailxml_defs:Longitude mailxml_defs:Height </div> </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **IMcbAndIMtbPieceScanInfoType**

diagram	<pre> xsd:complexType base="base" name="IMcbAndIMtbPieceScanInfoType"> <div> mailxml_defs:IMcb mailxml_defs:IMtbPieceScanInfo </div> </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **IMcbPieceScanInfoType**

diagram	<pre> xsd:complexType base="base" name="IMcbPieceScanInfoType"> <div> mailxml_defs:IMcb mailxml_defs:IMbScanRec </div> </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **IMtbPieceScanInfoType**

diagram	<pre> xsd:complexType base="base" name="IMtbPieceScanInfoType"> <div> mailxml_defs:IMtb mailxml_defs:IMbScanRec </div> </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

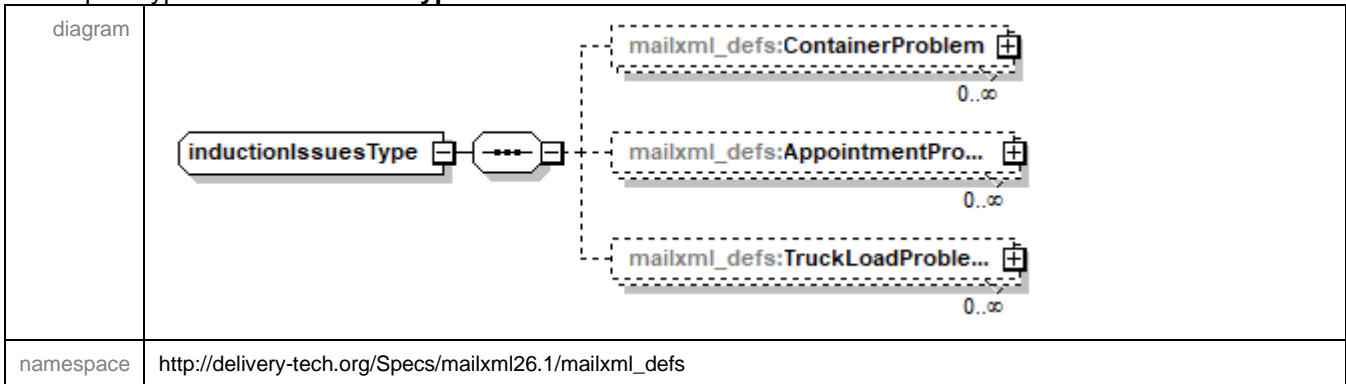
complexType inductionCloseoutInfoType

diagram

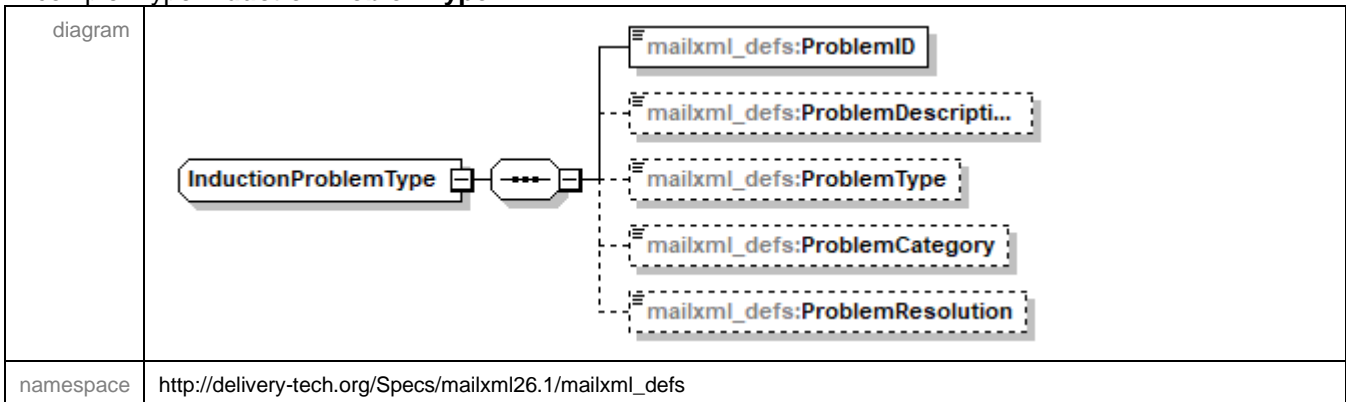


namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

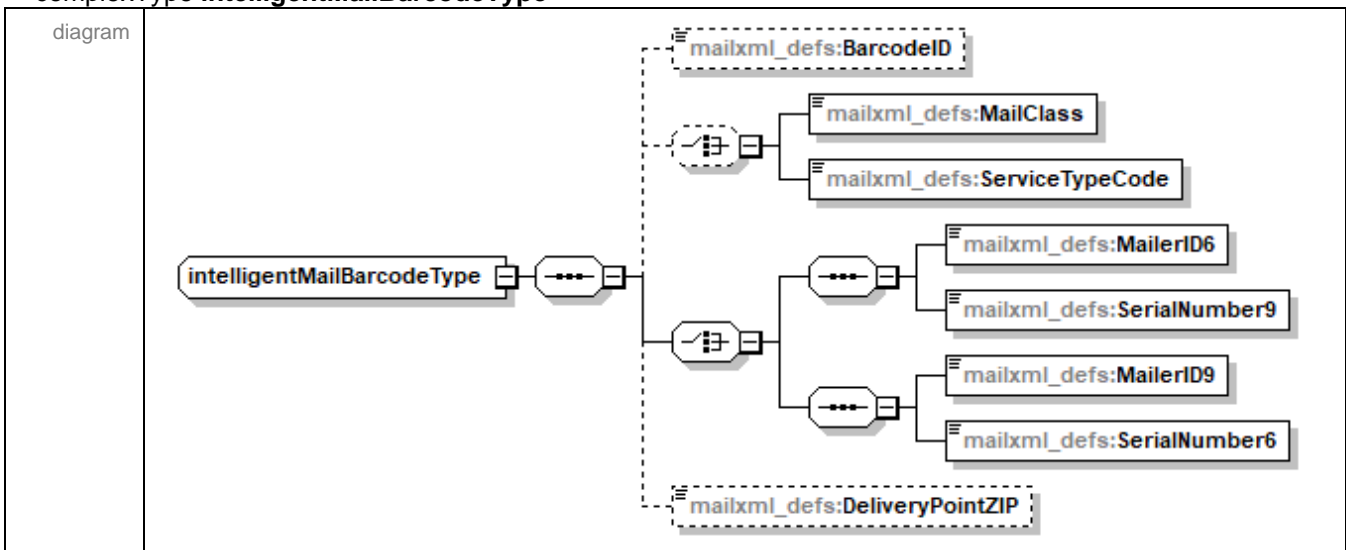
complexType inductionIssuesType



complexType InductionProblemType



complexType intelligentMailBarcodeType



namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

complexType **intelligentMailPackageBarcodeType**

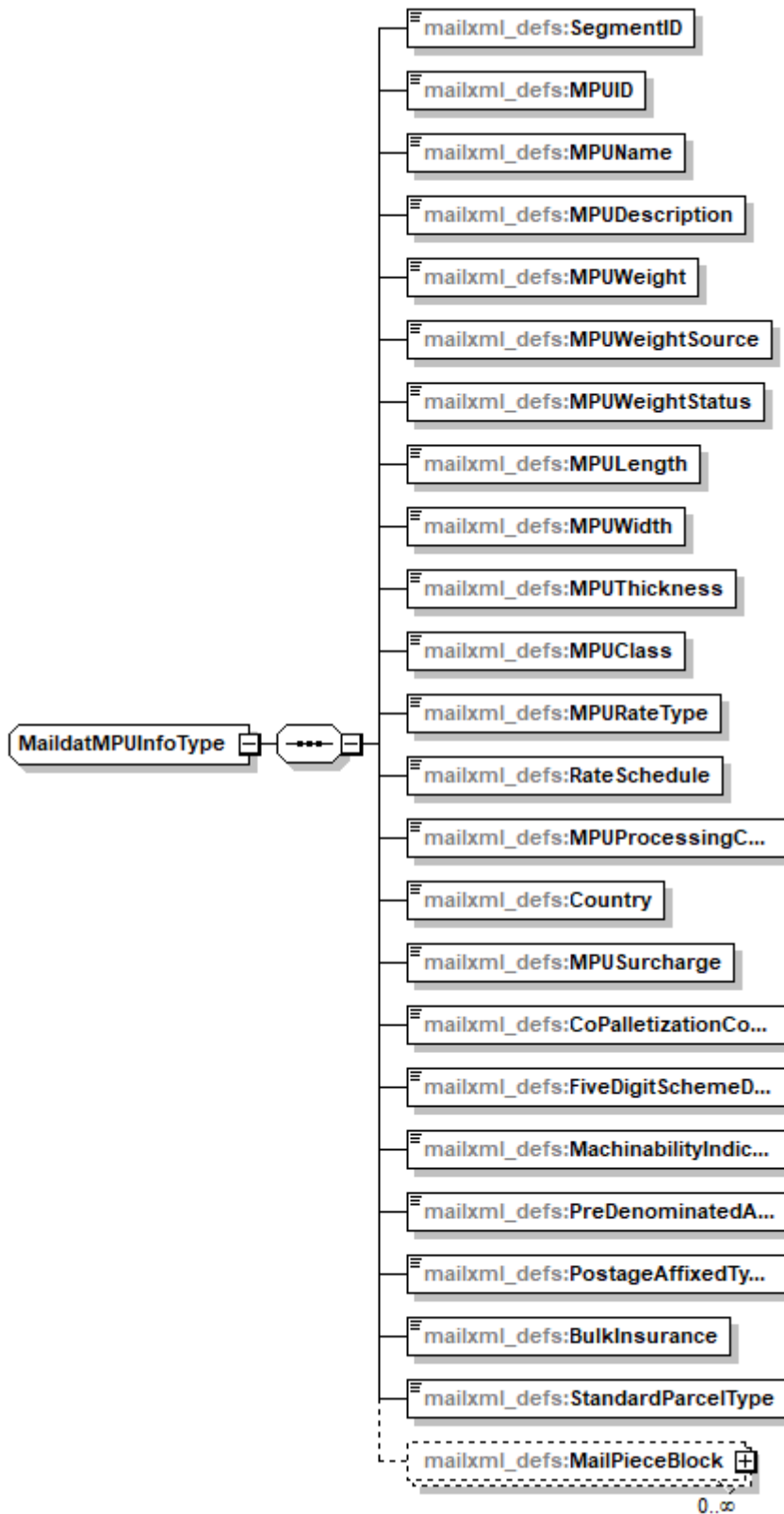
diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **maildatContainerIDType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType MaildatMPUInfoType

diagram



namespace http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType mailPieceIDType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType MailXMLContainerIDType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType mailxmlDetailType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType measurementType

diagram	
---------	--

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
-----------	---

complexType MIDType

diagram	<pre> sequenceDiagram MIDType --> Choice1or3 Choice1or3 --> MID6[mailxml_defs:MID6] Choice1or3 --> MID9[mailxml_defs:MID9] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType MPSRBlockType

diagram	<pre> sequenceDiagram MPSRBlockType --> Choice1or3 Choice1or3 --> ContainerID[mailxml_defs:ContainerID 1..∞] Choice1or3 --> PSRBlock[mailxml_defs:PSRBlock 1..∞] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

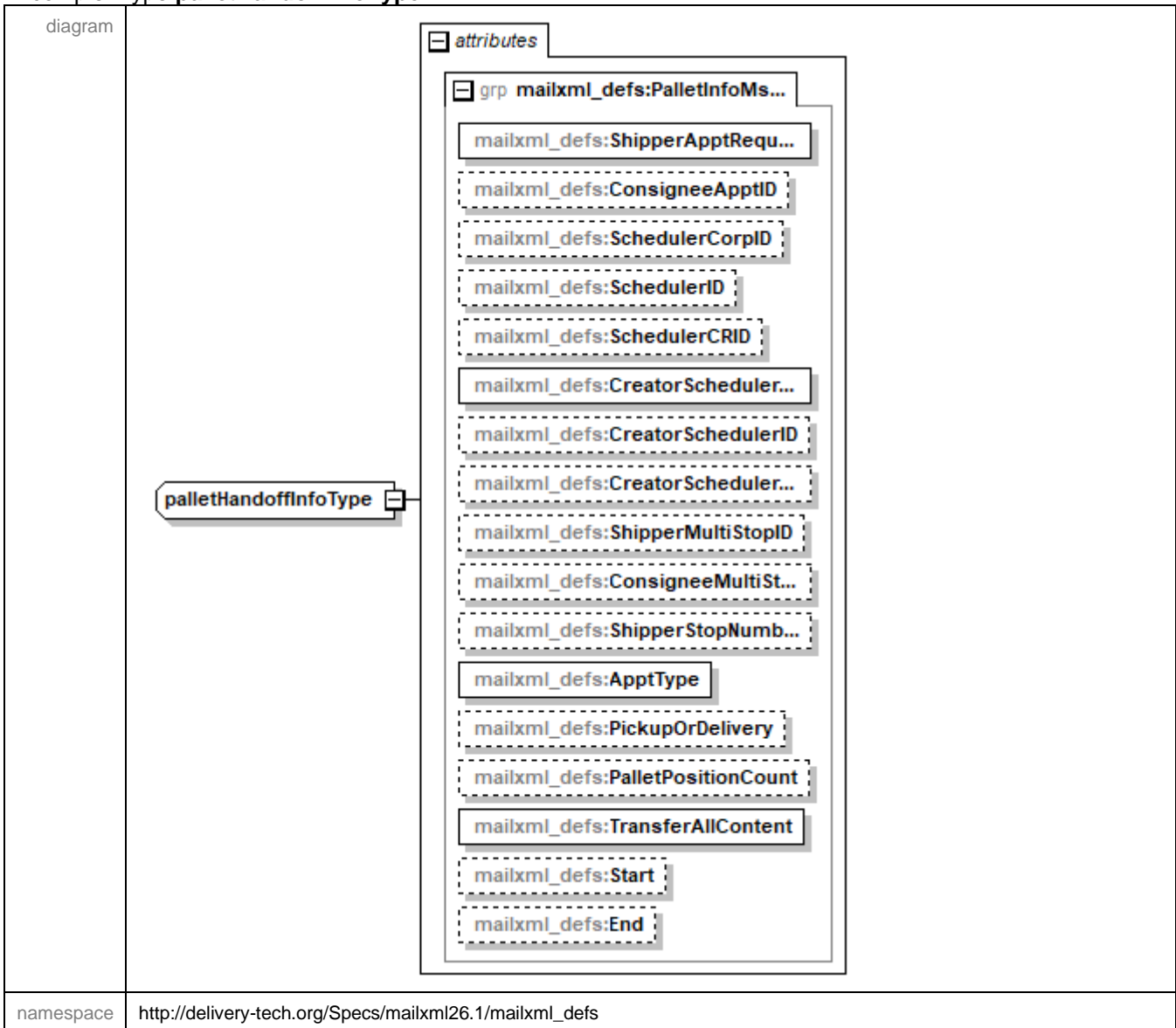
complexType MPSVisIncludedInScanRecFlagType

diagram	<pre> sequenceDiagram MPSVisIncludedInScanRecFlagT... --> Choice1or3 Choice1or3 --> IMcb[mailxml_defs:IMcb] Choice1or3 --> IMtb[mailxml_defs:IMtb] Choice1or3 --> BundleID[mailxml_defs:BundleID] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

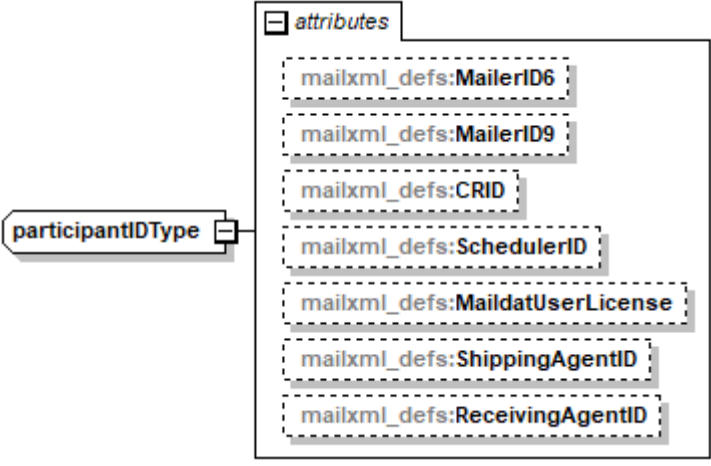
complexType MPSVisResultOptionsType

diagram	<pre> sequenceDiagram MPSVisResultOptionsType --> Choice1or3 Choice1or3 --> IMcb[mailxml_defs:IMcb] Choice1or3 --> IMtb[mailxml_defs:IMtb] Choice1or3 --> IMB[mailxml_defs:IMB] Choice1or3 --> IMpb[mailxml_defs:IMpb] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

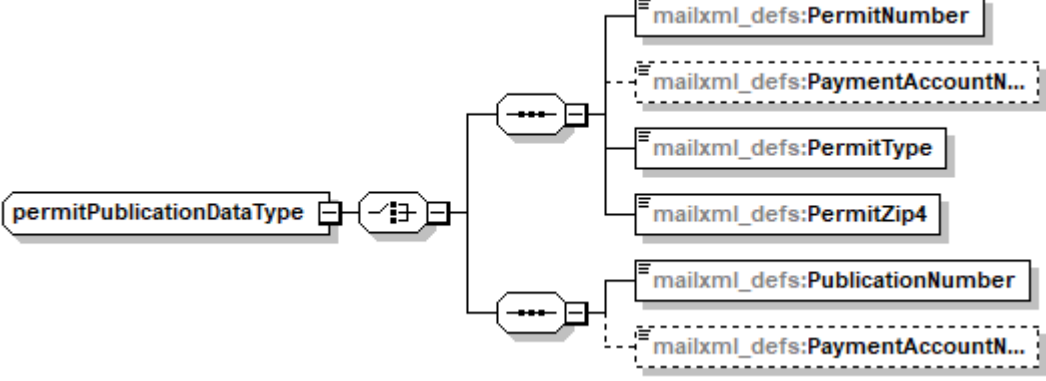
complexType **palletHandoffInfoType**



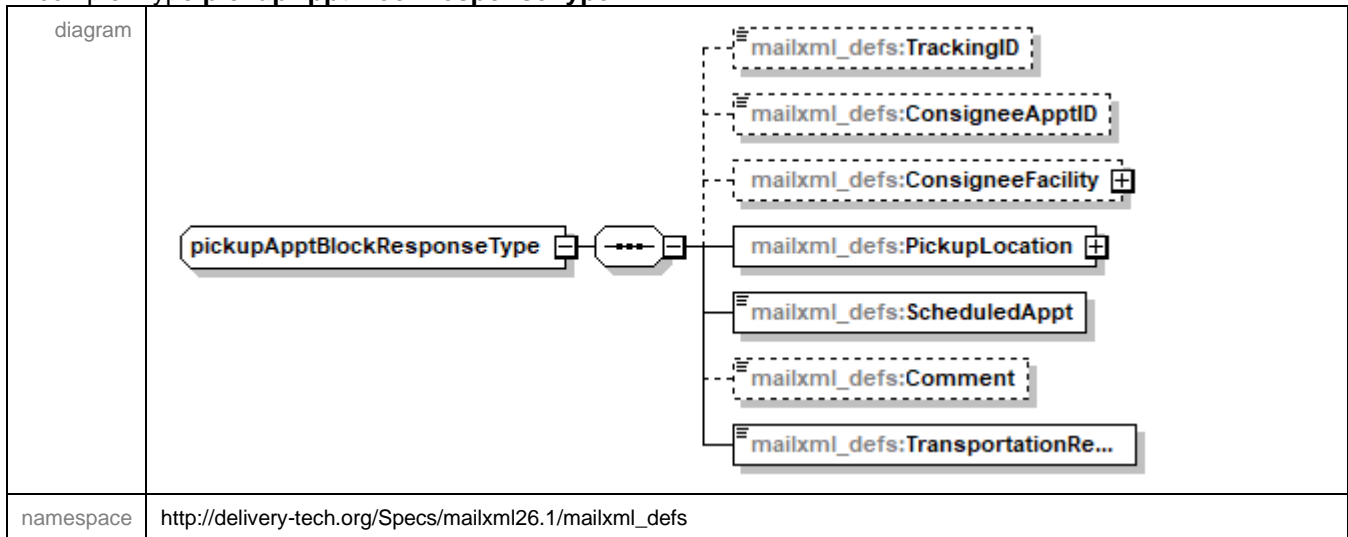
complexType participantIDType

diagram	 <p>The diagram shows the participantIDType complex type. It has a single child element named attributes, which is a container for seven attributes. Each attribute is represented by a dashed box containing the text <code>mailxml_defs:</code> followed by the attribute name. The attributes are: MailerID6, MailerID9, CRID, SchedulerID, MaildatUserLicense, ShippingAgentID, and ReceivingAgentID.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

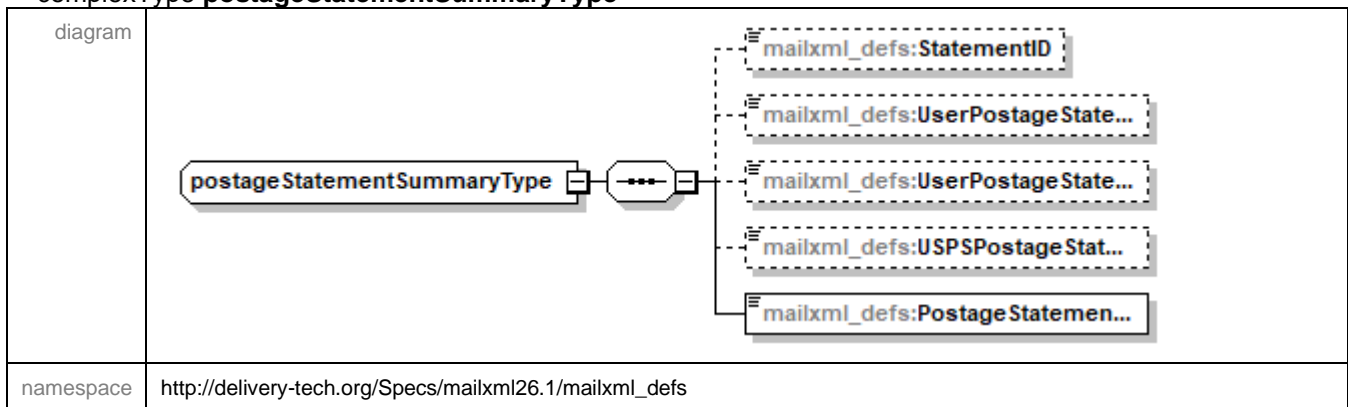
complexType permitPublicationDataType

diagram	 <p>The diagram shows the permitPublicationDataType complex type. It has a single child element, which is a container for two groups. Each group is represented by a dashed box containing the text <code>mailxml_defs:</code> followed by the group name. The groups are: PermitNumber and PaymentAccountN.... Each group contains two attributes: PermitType and PermitZip4.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

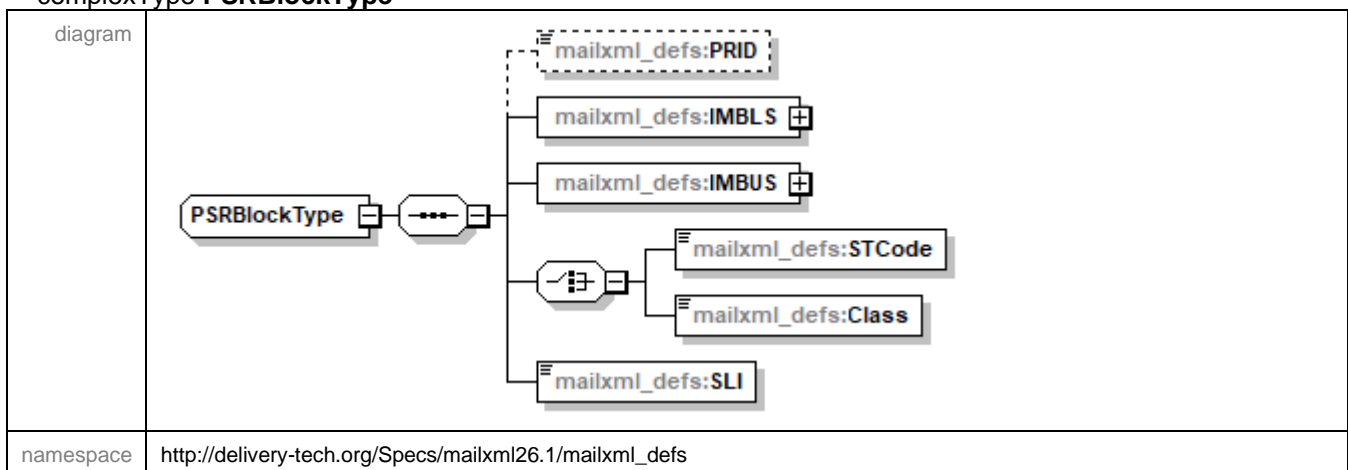
complexType pickupApptBlockResponseType




complexType postageStatementSummaryType



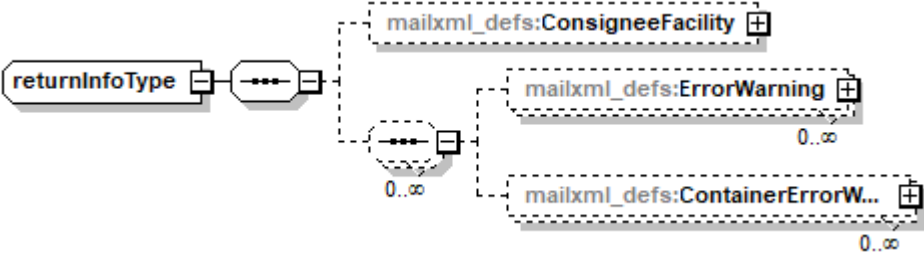
complexType PSRBlockType



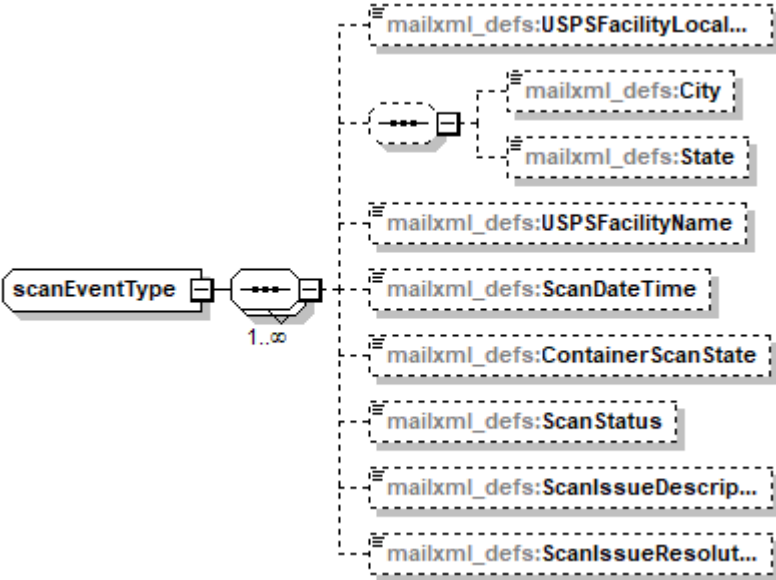
complexType queryErrorType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

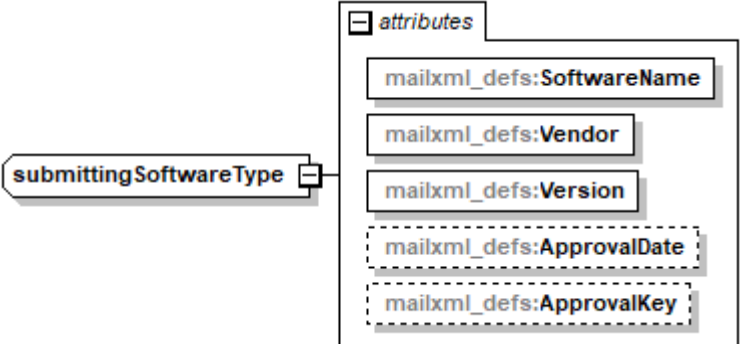
complexType returnInfoType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs


complexType scanEventType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **submittingSoftwareType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs

complexType **zipCode**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	extension of mailxml_base:ns09

simpleType **bundleScanTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **containerDiscrepancyCategoryType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **containerScanStateType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string
annotation	documentation Scan data for Container Scan States

simpleType **containerStatusType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **countTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **einductionDataSourceType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **fullServiceComplianceIndicatorType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	mailxml_base:yesNo

simpleType **MPSQueryType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string
annotation	documentation Mail Piece Scan Query Type

simpleType **MPSStateType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string
annotation	documentation Scan States for Mail Piece Scan Data

simpleType **pieceScanEventTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **problemCategoryType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **problemTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **reasonCodeType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

simpleType **retrieveDataByType**

namespace	http://delivery-tech.org/Specs/mailxml26.1/mailxml_defs
type	restriction of xs:string

XML Schema documentation generated by [XMLSpy](http://www.altova.com/xmlspy) Schema Editor <http://www.altova.com/xmlspy>