

# ***Mail.XML Version 26.3***

## **System Messages Specification**

**Wednesday, March 20, 2024**

### **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 26.3?

With this release, the Mail.XML Messaging Protocol moves to Version 26.3. This release supports structure changes required by mailing industry and Postal Service.

Changes supported by Mail.XML 26.3 include:

- 2619 – Proposal to support the new promotions and add-ons for 2025 Mailing Promotions.
- 2622 - Proposal to support the types of Election Mail (Election Mail Official Ballot and Election Mail Non-Ballot Materials)
- 2623 - Proposal to support Protected Origin Mixed ADC pallet preparation level for Periodicals flats.

## 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/mailxml26.3/mailxml"`  
`targetNamespace="http://delivery-tech.org/Specs/mailxml26.3/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/mailxml26.3A/mailxml"` then at least one of the XSDs is at same version such as `filename='Mail.XML_26.3A.xsd' <- Errata A`
- If the wrapper version is labeled as `xmlns:mailxml="http://deliverytech.org/Specs/mailxml26.3B/mailxml"` then at least one of the XSDs is at same version such as `filename='Mail.XML_26.3B.xsd' <- Errata B`
- If the wrapper version is labeled as `xmlns:mailxml="http://deliverytech.`

org/Specs/mailxml26.3/mailxml" then at least one of the XSDs is at same version such as  
filename ='Mail.XML\_26.3.xsd' <- Major Version

## Mail.XML 26.3 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™ 26.3 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 26.3. Updates in this Specification are NOT backwardly compatible with previous versions. Other documents in the specification set include:

- Mail.XML™ 26.3: Transportation Messaging Specification documents all transportation messages
- Mail.XML™ 26.3: Mailing Messaging Specification documents all mailing messages
- Mail.XML™ 26.3: Informed Visibility Specification documents all informed visibility messages
- Mail.XML™ 26.3: Data Distribution Messaging Specification documents all data distribution messages
- Mail.XML™ 26.3: Identification Messaging Specification documents all identification messages
- Mail.XML™ 26.3: Supply Chain Messaging Specification documents all supply chain messages
- Mail.XML™ 26.3: System Messaging Specification documents all systems and fault messages
- Mail.XML™ 26.3: Simple Types Specification documents all simple types used across Mail.XML messages
- Mail.XML™ 26.3: Common Definitions Specification documents all shared elements and complex

- types.
- Mail.XML™ 26.3: 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 26.3? .....	3
About Mail.XML Schema Modularization .....	4
Mail.XML Module Versioning Rules .....	4
Mail.XML 26.3 XSD Modules .....	5
The Mail.XML™ 26.3 Messaging Documentation Set .....	5
Schema mailxml_defs_26.3.xsd .....	8

# Schema mailxml\_defs\_26.3.xsd

schema location: [..\XSDs\mailxml\\_defs\\_26.3.xsd](..\XSDs\mailxml_defs_26.3.xsd)  
attributeFormDefault: **qualified**  
elementFormDefault: **qualified**  
targetNamespace: [http://delivery-tech.org/Specs/mailxml26.3/mailxml\\_defs](http://delivery-tech.org/Specs/mailxml26.3/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](#)

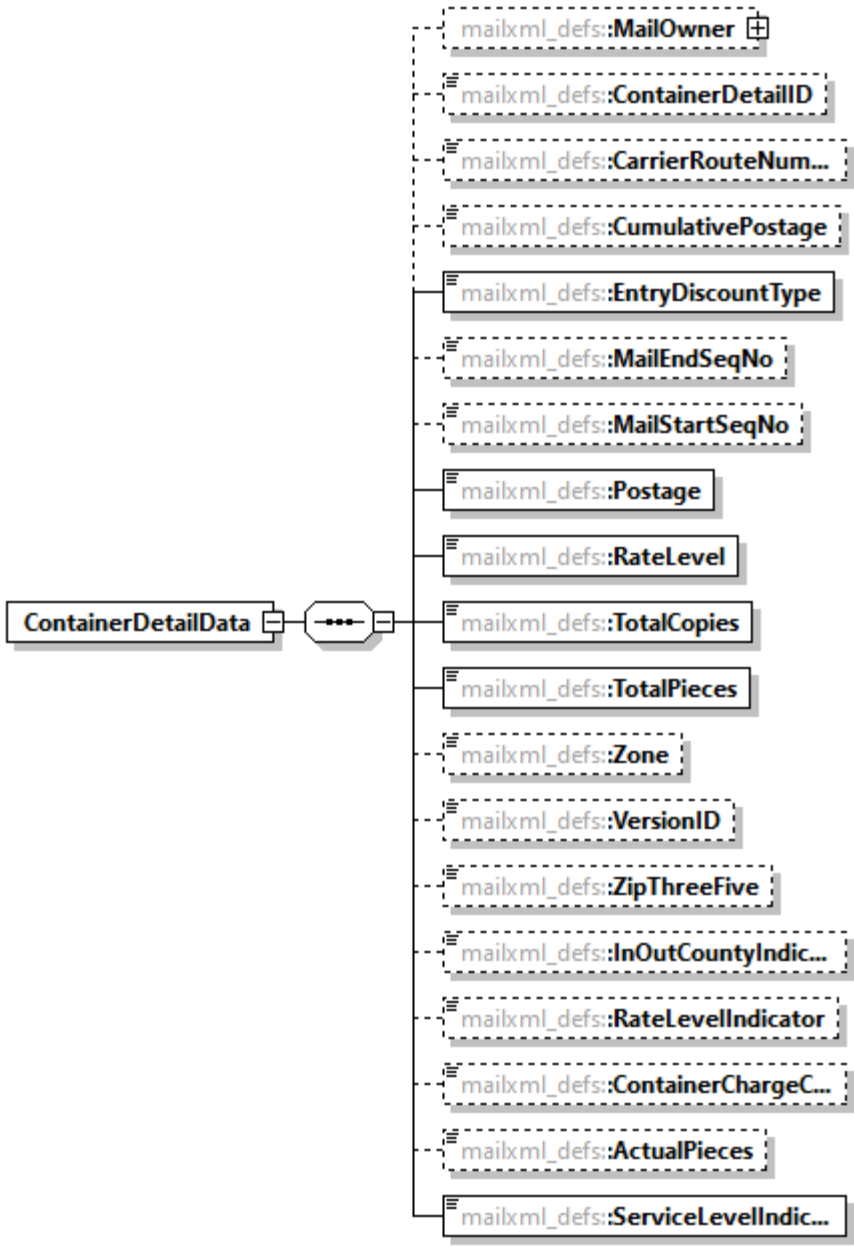
## Simple types

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



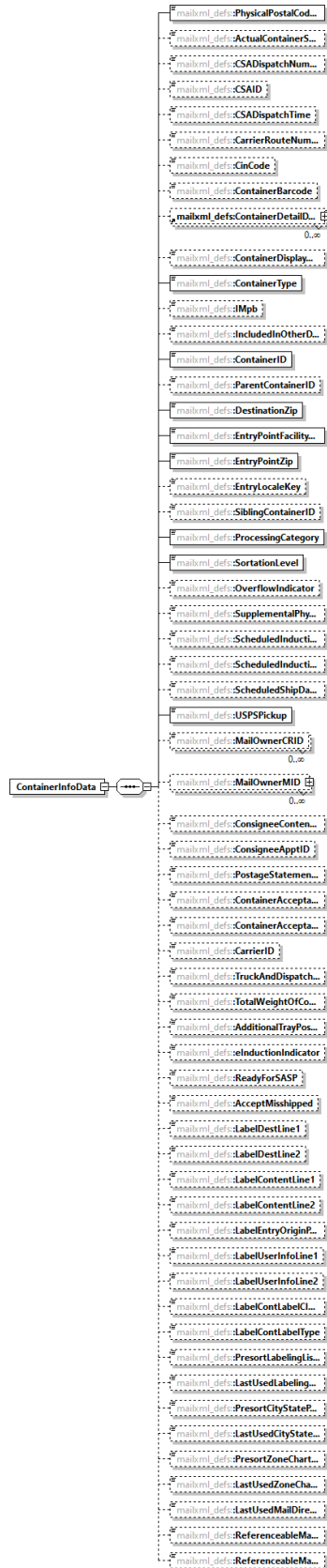
[mailxmlDetailType](#)  
[measurementType](#)  
[MIDType](#)  
[MPSRBlockType](#)  
[MPSVisIncludedInScanRecFlagType](#)  
[MPSVisResultOptionsType](#)  
[palletHandoffInfoType](#)  
[participantIDType](#)  
[permitPublicationDataType](#)  
[pickupApptBlockResponseType](#)  
[postageStatementSummaryType](#)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

element **ContainerDetailData**

diagram	 <pre> classDiagram     class ContainerDetailData {         MailOwner         ContainerDetailID         CarrierRouteNum...         CumulativePostage         EntryDiscountType         MailEndSeqNo         MailStartSeqNo         Postage         RateLevel         TotalCopies         TotalPieces         Zone         VersionID         ZipThreeFive         InOutCountyIndic...         RateLevelIndicator         ContainerChargeC...         ActualPieces         ServiceLevelIndic...     }         </pre> <p>The diagram illustrates the structure of the <b>ContainerDetailData</b> element. It is a container element, indicated by the square icon with a small circle on its right side. A dashed line connects the container to a sequence of child elements, represented by a cylinder icon with three dots. The child elements are listed as follows:</p> <ul style="list-style-type: none"> <li><b>mailxml_defs:MailOwner</b> (dashed border)</li> <li><b>mailxml_defs:ContainerDetailID</b> (dashed border)</li> <li><b>mailxml_defs:CarrierRouteNum...</b> (dashed border)</li> <li><b>mailxml_defs:CumulativePostage</b> (dashed border)</li> <li><b>mailxml_defs:EntryDiscountType</b> (solid border)</li> <li><b>mailxml_defs:MailEndSeqNo</b> (dashed border)</li> <li><b>mailxml_defs:MailStartSeqNo</b> (dashed border)</li> <li><b>mailxml_defs:Postage</b> (solid border)</li> <li><b>mailxml_defs:RateLevel</b> (solid border)</li> <li><b>mailxml_defs:TotalCopies</b> (solid border)</li> <li><b>mailxml_defs:TotalPieces</b> (solid border)</li> <li><b>mailxml_defs:Zone</b> (dashed border)</li> <li><b>mailxml_defs:VersionID</b> (dashed border)</li> <li><b>mailxml_defs:ZipThreeFive</b> (dashed border)</li> <li><b>mailxml_defs:InOutCountyIndic...</b> (dashed border)</li> <li><b>mailxml_defs:RateLevelIndicator</b> (dashed border)</li> <li><b>mailxml_defs:ContainerChargeC...</b> (dashed border)</li> <li><b>mailxml_defs:ActualPieces</b> (dashed border)</li> <li><b>mailxml_defs:ServiceLevelIndic...</b> (solid border)</li> </ul>
namespace	<a href="http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs">http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs</a>

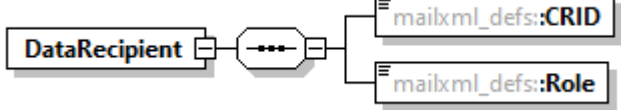
element **ContainerInfoData**

diagram

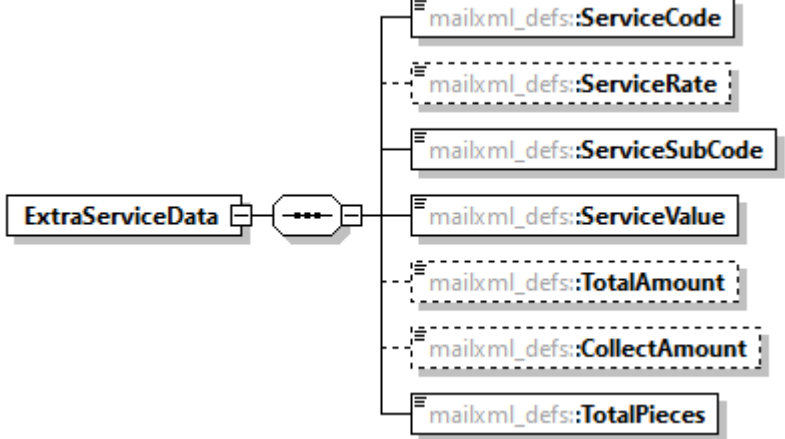


namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---


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/mailxml26.3/mailxml_defs

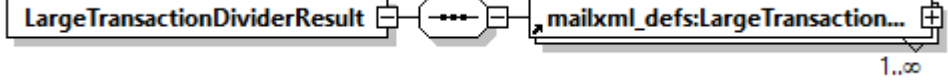
element **ExtraServiceData**

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/mailxml26.3/mailxml_defs

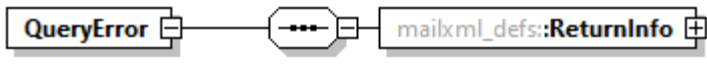
element **LargeTransactionDivider**

diagram	 <pre> graph LR     LTD[LargeTransactionDivider] --- C1(( ))     C1 --- TI[mailxml_defs::TrackingID] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

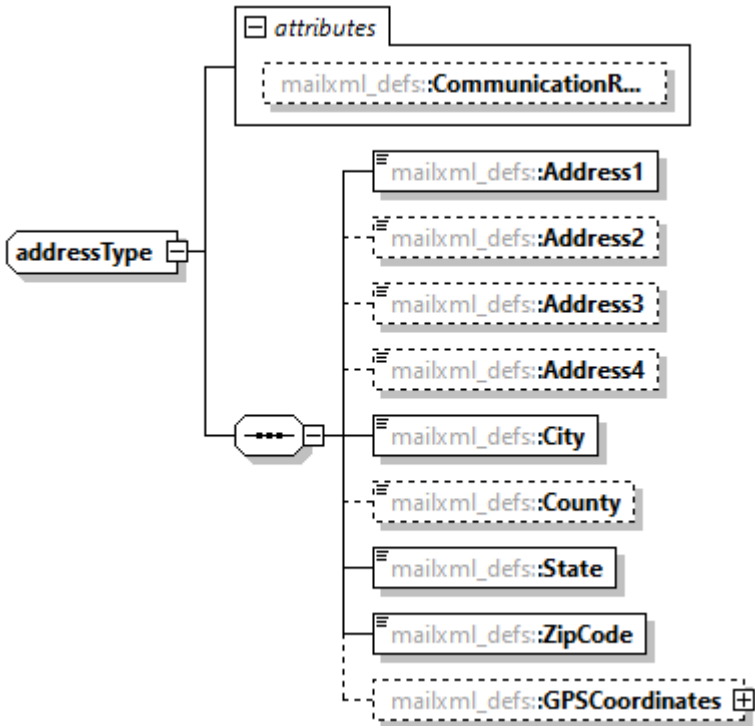
element **LargeTransactionDividerResult**

diagram	 <pre> graph LR     LTRD[LargeTransactionDividerResult] --- C1(( ))     C1 --- LTR[mailxml_defs::LargeTransaction...]     LTR -- 1..∞ --&gt; LTR </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

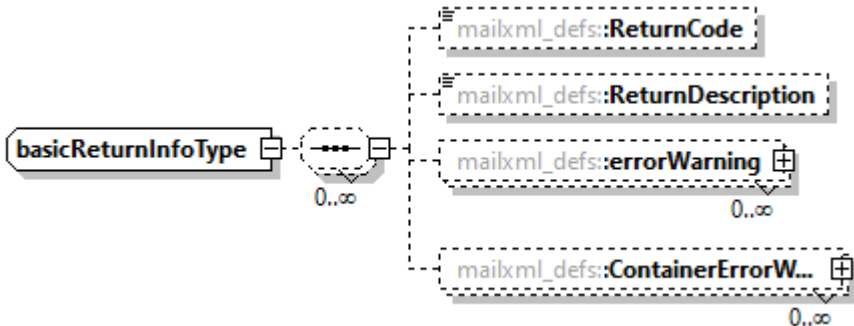
## element **QueryError**

diagram	 <p>The diagram shows the <b>QueryError</b> element connected to the <b>mailxml_defs::ReturnInfo</b> element via a dashed line with a circle containing three dots. The <b>QueryError</b> element has a small square icon, and the <b>mailxml_defs::ReturnInfo</b> element has a small square icon with a plus sign.</p> <p>Error issued when the query data cannot be provided.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
annotation	<p>documentation</p> <p>Error issued when the query data cannot be provided.</p>

## complexType **addressType**

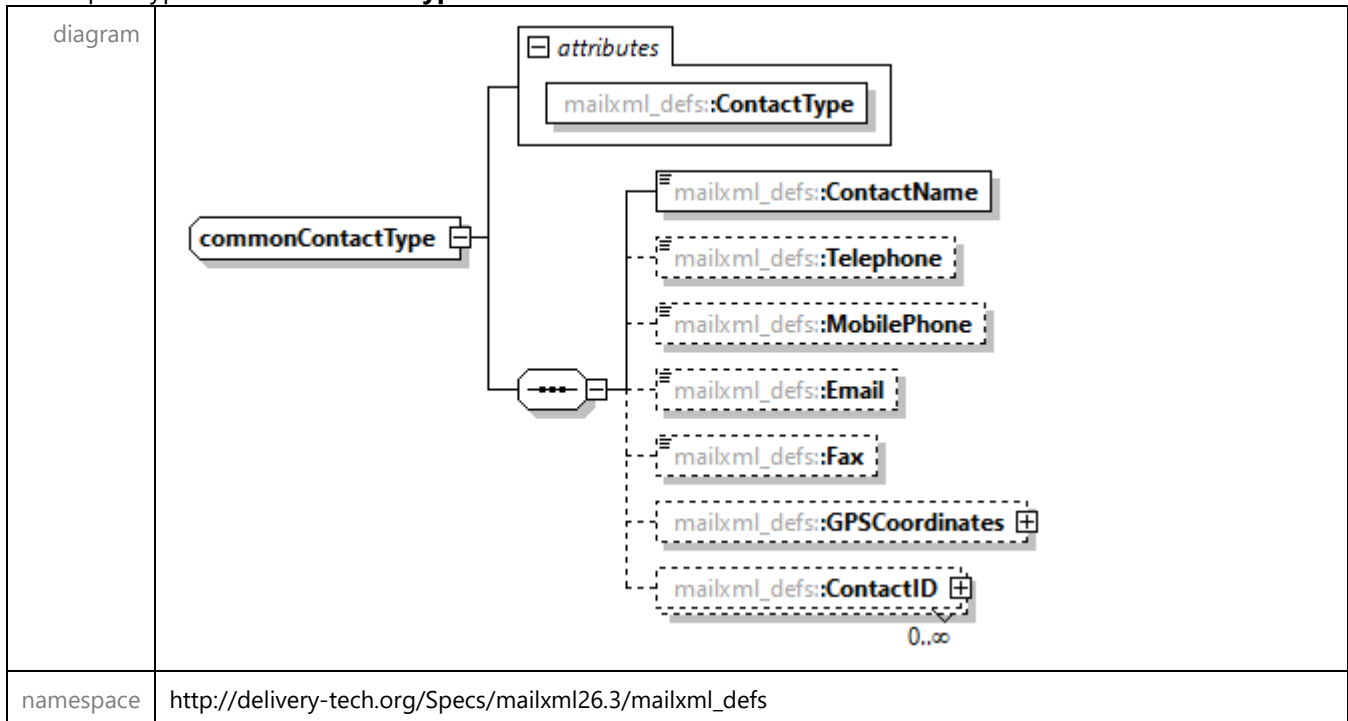
diagram	 <p>The diagram shows the <b>addressType</b> complexType. It has an <b>attributes</b> section containing <b>mailxml_defs::CommunicationR...</b>. The main body of the complexType is connected to a dashed line with a circle containing three dots, which then branches into several elements: <b>mailxml_defs::Address1</b>, <b>mailxml_defs::Address2</b>, <b>mailxml_defs::Address3</b>, <b>mailxml_defs::Address4</b>, <b>mailxml_defs::City</b>, <b>mailxml_defs::County</b>, <b>mailxml_defs::State</b>, <b>mailxml_defs::ZipCode</b>, and <b>mailxml_defs::GPSCoordinates</b>. The <b>GPSCoordinates</b> element has a small square icon with a plus sign.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

## complexType **basicReturnInfoType**

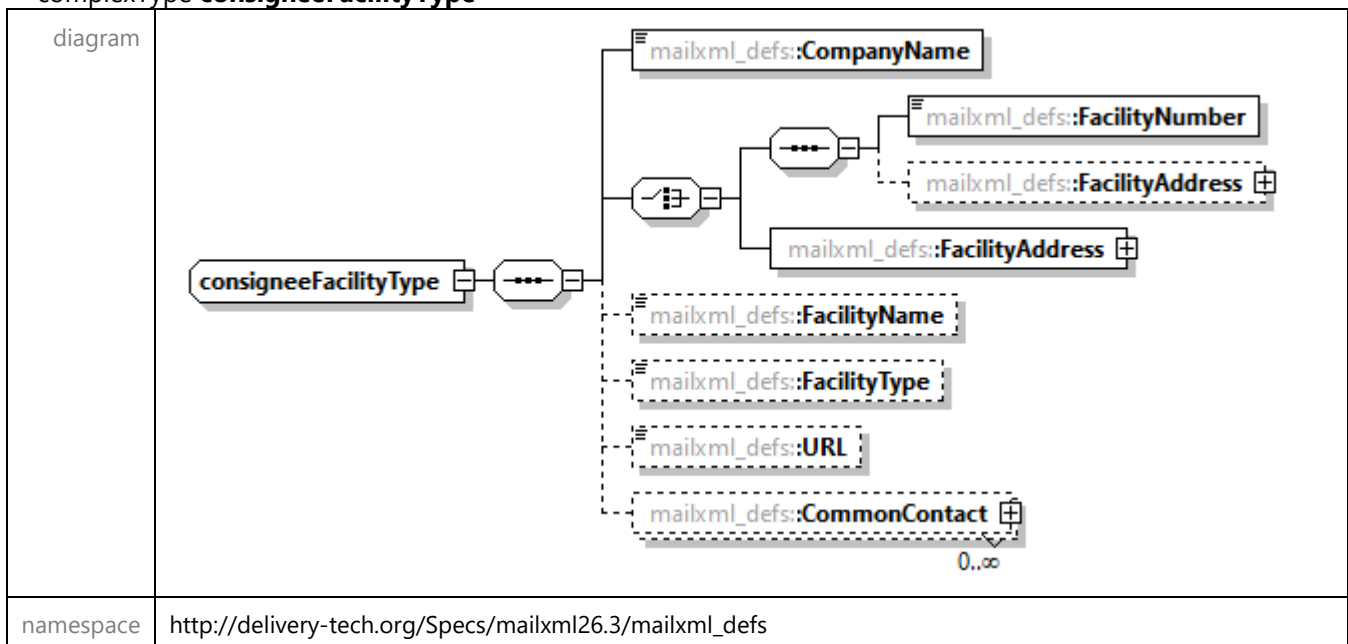
diagram	 <p>The diagram shows the <b>basicReturnInfoType</b> complexType. It is connected to a dashed line with a circle containing three dots, which then branches into four elements: <b>mailxml_defs::ReturnCode</b>, <b>mailxml_defs::ReturnDescription</b>, <b>mailxml_defs::errorWarning</b>, and <b>mailxml_defs::ContainerErrorW...</b>. The <b>errorWarning</b> and <b>ContainerErrorW...</b> elements have small square icons with plus signs. The cardinality <b>0..∞</b> is shown below the dashed line and at the bottom right of the diagram.</p>
---------	---

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

### complexType **commonContactType**



### complexType **consigneeFacilityType**



complexType **consolidatorCommunicationInfoType**

diagram

consolidatorCommunicationInf...

```

<?xml version="1.0" encoding="UTF-8" ?>
<consolidatorCommunicationInfoType>
  <?xml version="1.0" encoding="UTF-8" ?>
  <PalletID>
    <?xml version="1.0" encoding="UTF-8" ?>
  </PalletID>
  <ContainerBarcode>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ContainerBarcode>
  <InterrollID>
    <?xml version="1.0" encoding="UTF-8" ?>
  </InterrollID>
  <BillingNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </BillingNumber>
  <OriginFacilityInfo>
    <?xml version="1.0" encoding="UTF-8" ?>
  </OriginFacilityInfo>
  <ConsolidatorFacility>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ConsolidatorFacility>
  <PalletCount>
    <?xml version="1.0" encoding="UTF-8" ?>
  </PalletCount>
  <FacilityType>
    <?xml version="1.0" encoding="UTF-8" ?>
  </FacilityType>
  <DestinationCity>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DestinationCity>
  <DestinationState>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DestinationState>
  <DestinationZipCode>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DestinationZipCode>
  <LocalKey>
    <?xml version="1.0" encoding="UTF-8" ?>
  </LocalKey>
  <FirstPickupDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </FirstPickupDate>
  <ActualWarehouse>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ActualWarehouse>
  <DueAtConsolidator>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DueAtConsolidator>
  <RecdAtConsolidator>
    <?xml version="1.0" encoding="UTF-8" ?>
  </RecdAtConsolidator>
  <ConsolidatedInvoice>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ConsolidatedInvoice>
  <ShippedQuantity>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ShippedQuantity>
  <ShipmentComplete>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ShipmentComplete>
  <ConsolidatorShip>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ConsolidatorShip>
  <BeginDeliveryWindow>
    <?xml version="1.0" encoding="UTF-8" ?>
  </BeginDeliveryWindow>
  <EndDeliveryWindow>
    <?xml version="1.0" encoding="UTF-8" ?>
  </EndDeliveryWindow>
  <ApptDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ApptDate>
  <ActualDeliveryDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ActualDeliveryDate>
  <CoPalPoolDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </CoPalPoolDate>
  <DeliveryType>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DeliveryType>
  <PurchaseOrderNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </PurchaseOrderNumber>
  <SackTrayCount>
    <?xml version="1.0" encoding="UTF-8" ?>
  </SackTrayCount>
  <PieceCount>
    <?xml version="1.0" encoding="UTF-8" ?>
  </PieceCount>
  <Weight>
    <?xml version="1.0" encoding="UTF-8" ?>
  </Weight>
  <ContainerStatus>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ContainerStatus>
  <LastPickupDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </LastPickupDate>
  <LastPickupType>
    <?xml version="1.0" encoding="UTF-8" ?>
  </LastPickupType>
  <CurrentZone>
    <?xml version="1.0" encoding="UTF-8" ?>
  </CurrentZone>
  <CurrentLocation>
    <?xml version="1.0" encoding="UTF-8" ?>
  </CurrentLocation>
  <AppointmentDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </AppointmentDate>
  <AppointmentNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </AppointmentNumber>
  <CustomerName>
    <?xml version="1.0" encoding="UTF-8" ?>
  </CustomerName>
  <InvoiceNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </InvoiceNumber>
  <BillTo>
    <?xml version="1.0" encoding="UTF-8" ?>
  </BillTo>
  <MailPreparerJobID>
    <?xml version="1.0" encoding="UTF-8" ?>
  </MailPreparerJobID>
  <StatComment>
    <?xml version="1.0" encoding="UTF-8" ?>
  </StatComment>
  <ServiceRequestID>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ServiceRequestID>
  <ServiceItem>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ServiceItem>
  <ReferenceNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ReferenceNumber>
  <VersionNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </VersionNumber>
  <Method>
    <?xml version="1.0" encoding="UTF-8" ?>
  </Method>
  <OutboundShipping>
    <?xml version="1.0" encoding="UTF-8" ?>
  </OutboundShipping>
  <MidDayOffTransit>
    <?xml version="1.0" encoding="UTF-8" ?>
  </MidDayOffTransit>
  <ActualTransit>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ActualTransit>
  <HoldDays>
    <?xml version="1.0" encoding="UTF-8" ?>
  </HoldDays>
  <DeliveryConfirmation>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DeliveryConfirmation>
  <ScheduledPickup>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ScheduledPickup>
  <ScheduledPickup>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ScheduledPickup>
  <PS8125Received>
    <?xml version="1.0" encoding="UTF-8" ?>
  </PS8125Received>
  <SegmentTapeCode>
    <?xml version="1.0" encoding="UTF-8" ?>
  </SegmentTapeCode>
  <BillingVersionNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </BillingVersionNumber>
  <MailPreparerJobNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </MailPreparerJobNumber>
  <MailPreparerJobNumber>
    <?xml version="1.0" encoding="UTF-8" ?>
  </MailPreparerJobNumber>
  <StartFacilityDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </StartFacilityDate>
  <EndFacilityDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </EndFacilityDate>
  <DSMSStatus>
    <?xml version="1.0" encoding="UTF-8" ?>
  </DSMSStatus>
  <ReleasedBy>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ReleasedBy>
  <ReleasedDate>
    <?xml version="1.0" encoding="UTF-8" ?>
  </ReleasedDate>
</consolidatorCommunicationInfoType>

```



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

complexType **contactIDType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

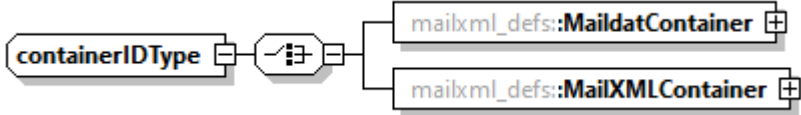
complexType **ContainerDiscrepancyType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

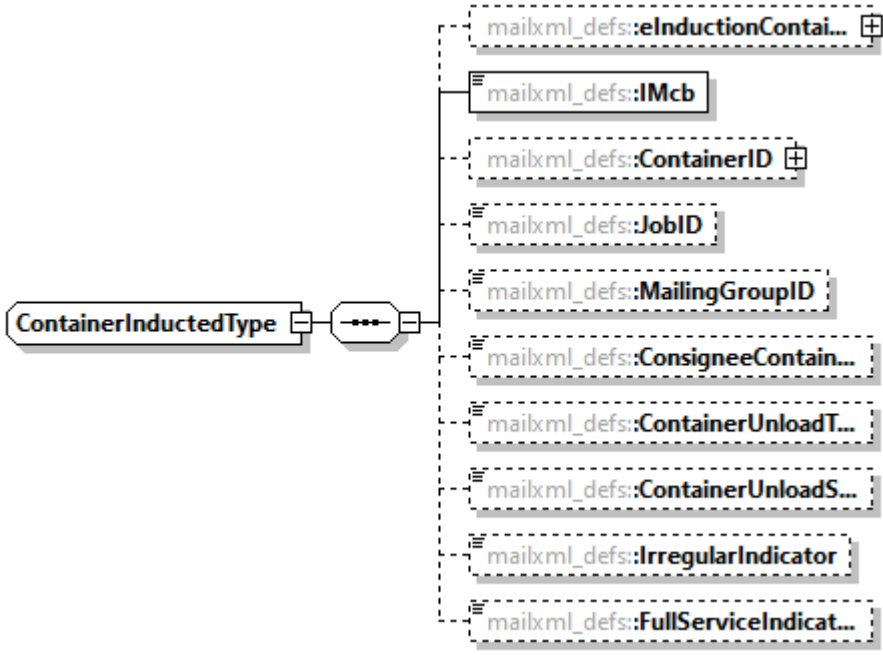
complexType **containerErrorWarningBlockType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **containerIDType**

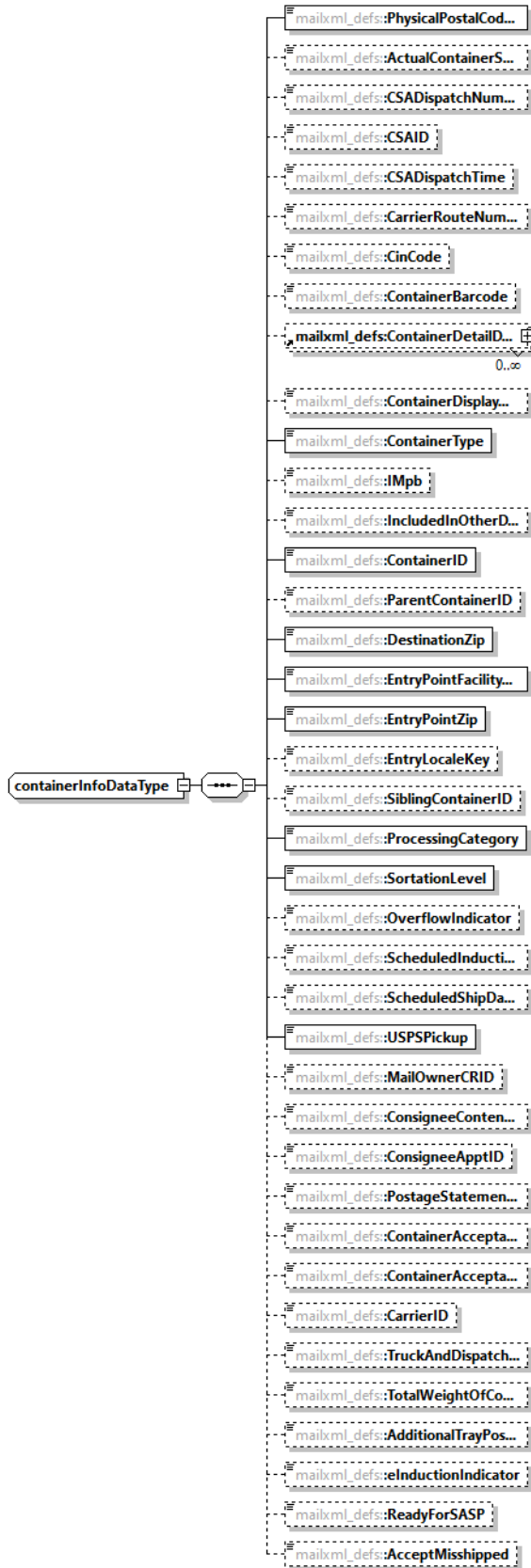
diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **ContainerInductedType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **containerInfoDataType**

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

complexType **containerKeysInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **containerPostInductionInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **containerPreInductionInfoType**

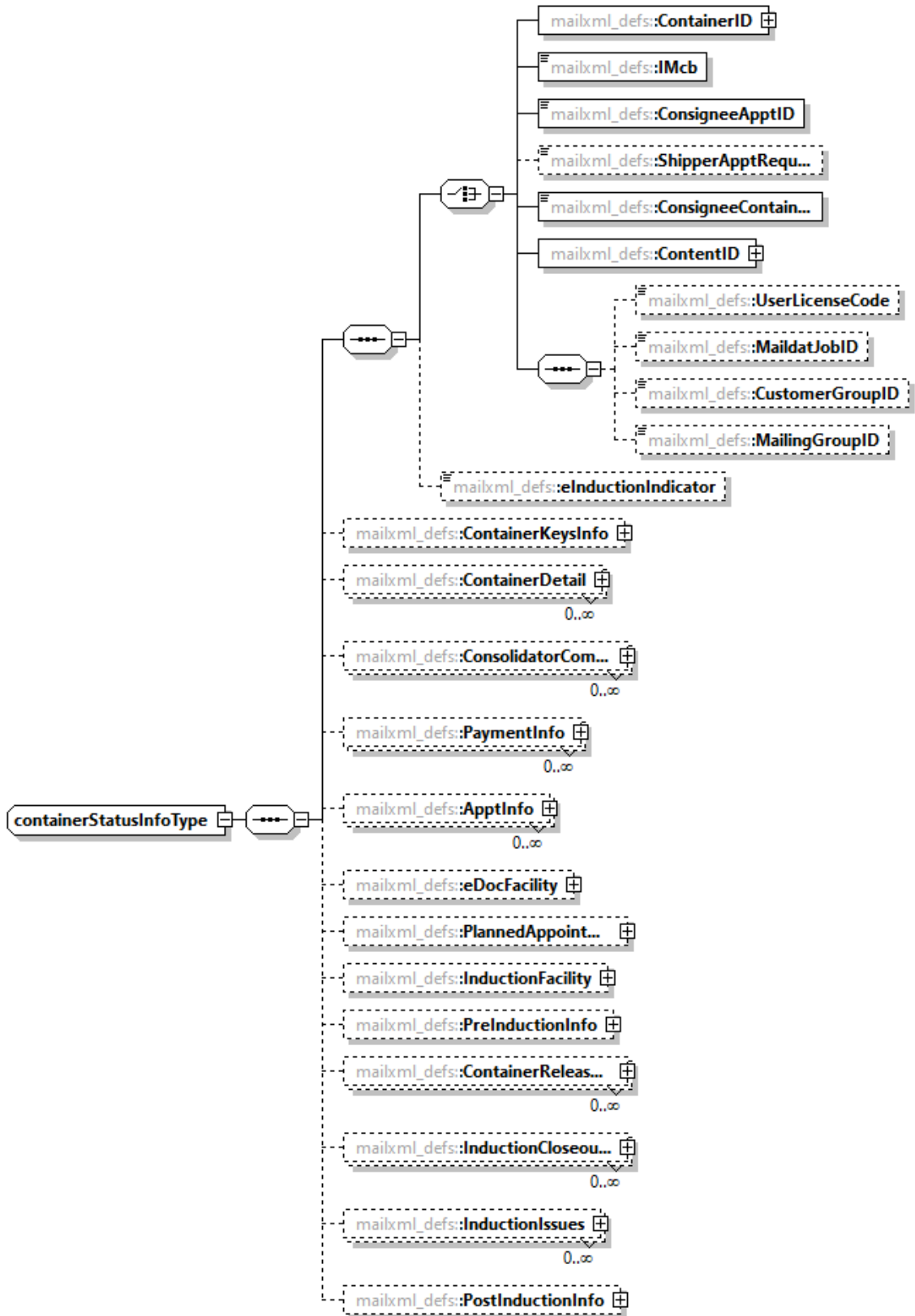
diagram	
namespace	<a href="http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs">http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs</a>

complexType **containerReleaseInfoType**

diagram	
namespace	<a href="http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs">http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs</a>

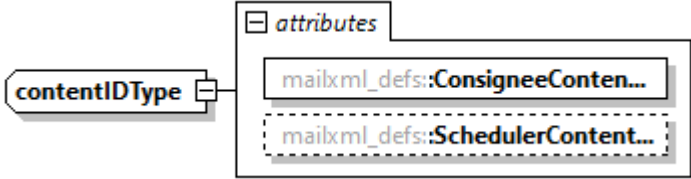
complexType **containerStatusInfoType**

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

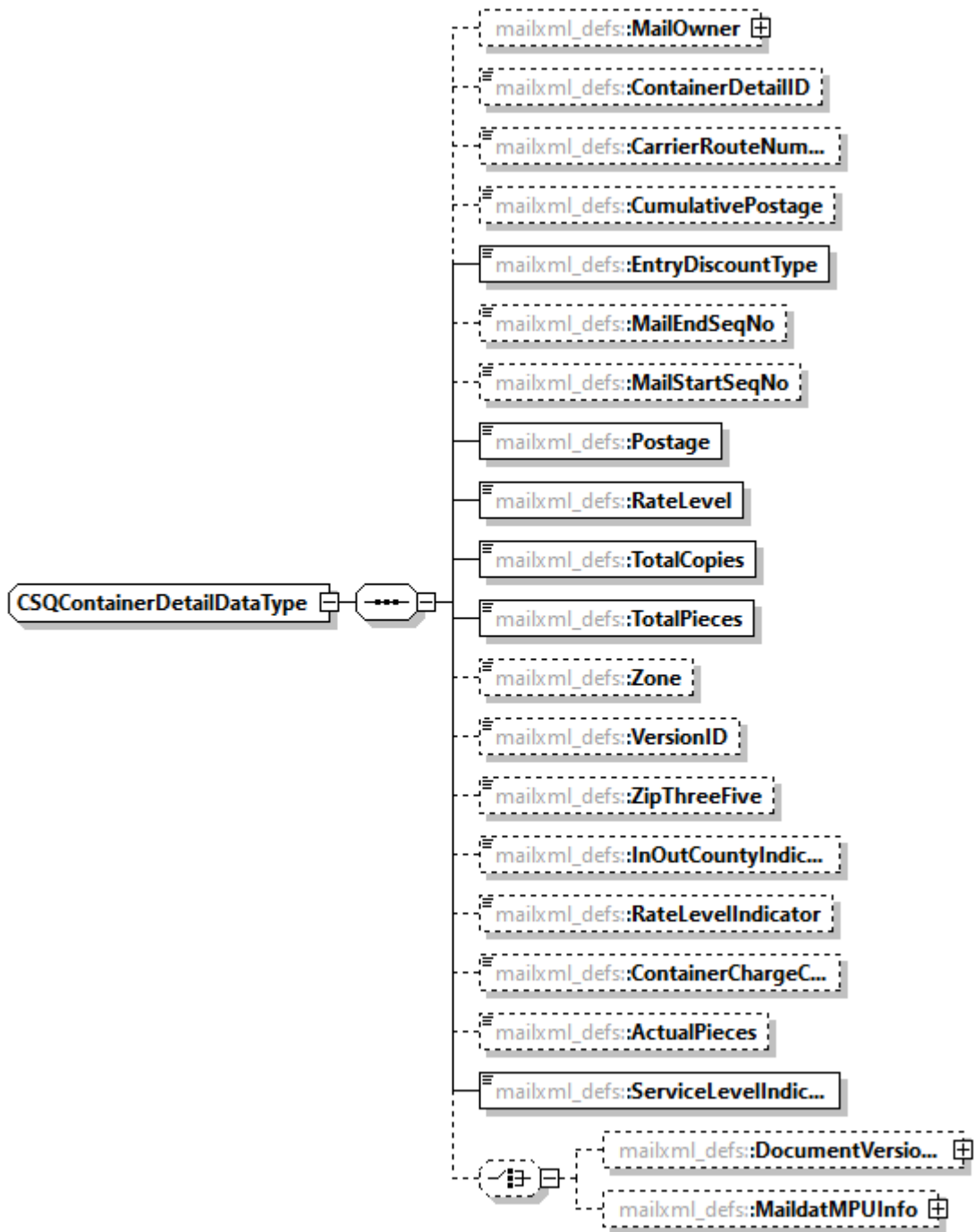
complexType **contentIDType**

diagram	 <p>The diagram shows a complex type named <b>contentIDType</b>. It has a tab labeled <i>attributes</i>. Inside the type, there are two elements: <code>mailxml_defs::ConsigneeContent...</code> and <code>mailxml_defs::SchedulerContent...</code>. The <code>SchedulerContent...</code> element is enclosed in a dashed border, indicating it is optional.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs



complexType **CSQContainerDetailDataType**

diagram



namespace

[http://delivery-tech.org/Specs/mailxml26.3/mailxml\\_defs](http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs)

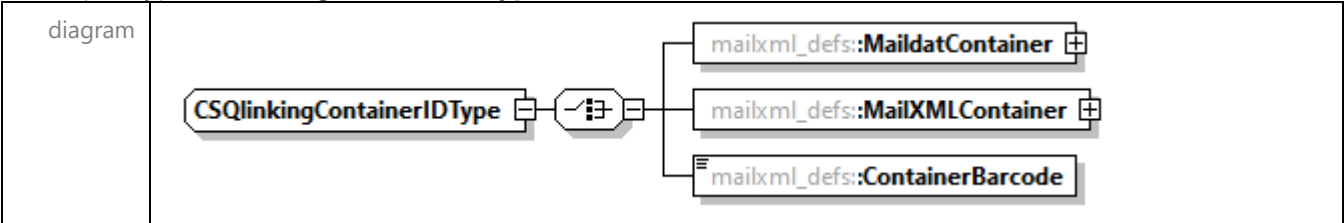
complexType **CSQcontainerInfoDataType**

diagram



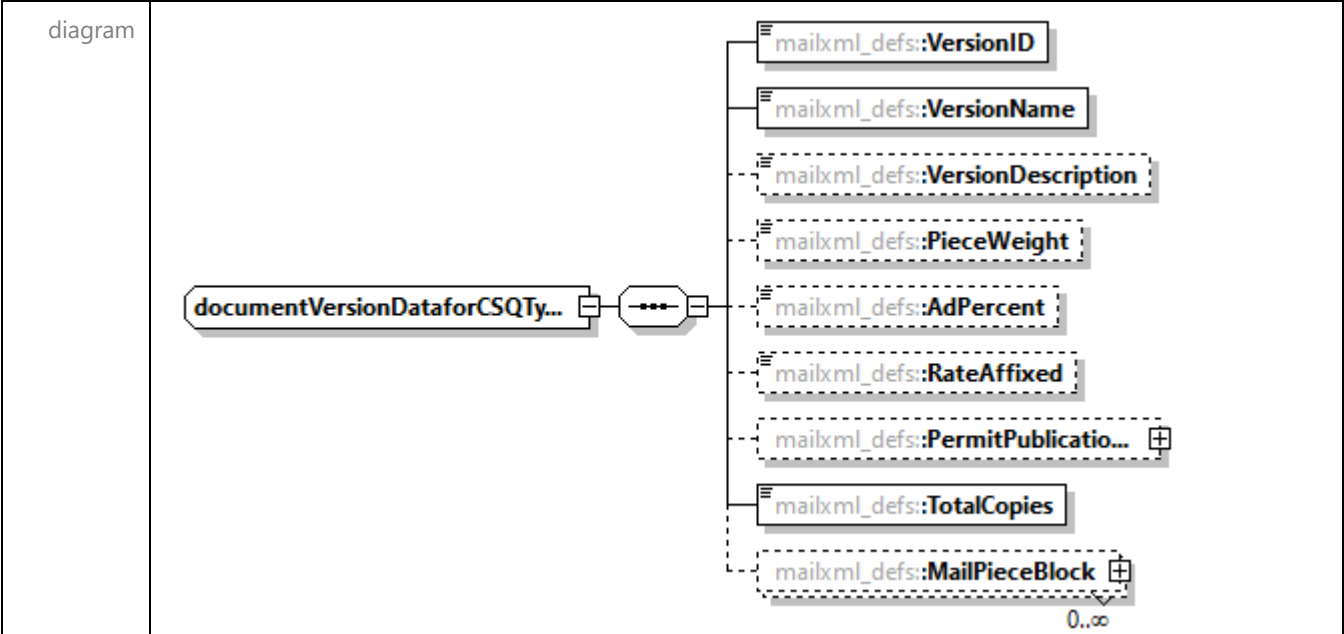
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

complexType **CSQlinkingContainerIDType**



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

complexType **documentVersionDataforCSQType**



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

# complexType **errorWarningType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

# complexType **fullContainerIDType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **gPSCoordinates**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **IMcbAndIMtbPieceScanInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **IMcbPieceScanInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **IMtbPieceScanInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **inductionCloseoutInfoType**

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

complexType **inductionIssuesType**

diagram	<pre> graph LR     A[inductionIssuesType] --- B(( ))     B --- C[mailxml_defs:ContainerProblem 0..∞]     B --- D[mailxml_defs:AppointmentPro... 0..∞]     B --- E[mailxml_defs:TruckLoadProblem 0..∞] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **InductionProblemType**

diagram	<pre> graph LR     A[InductionProblemType] --- B(( ))     B --- C[mailxml_defs:ProblemID]     B --- D[mailxml_defs:ProblemDescripti...]     B --- E[mailxml_defs:ProblemType]     B --- F[mailxml_defs:ProblemCategory]     B --- G[mailxml_defs:ProblemResolution] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs



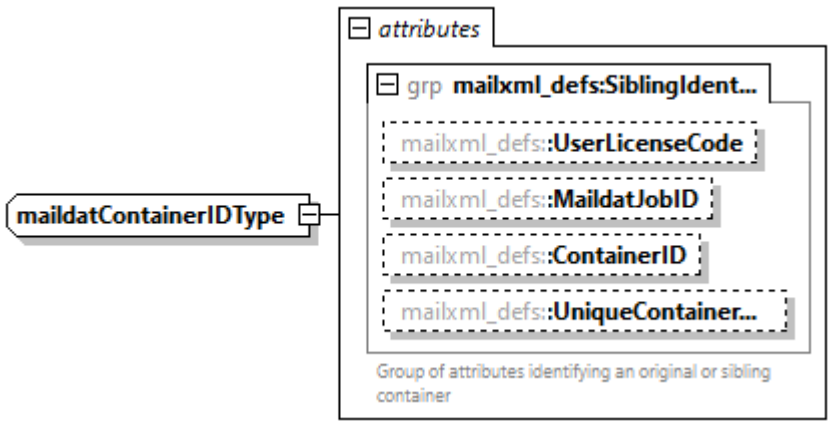
# complexType **intelligentMailBarcodeType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

# complexType **intelligentMailPackageBarcodeType**

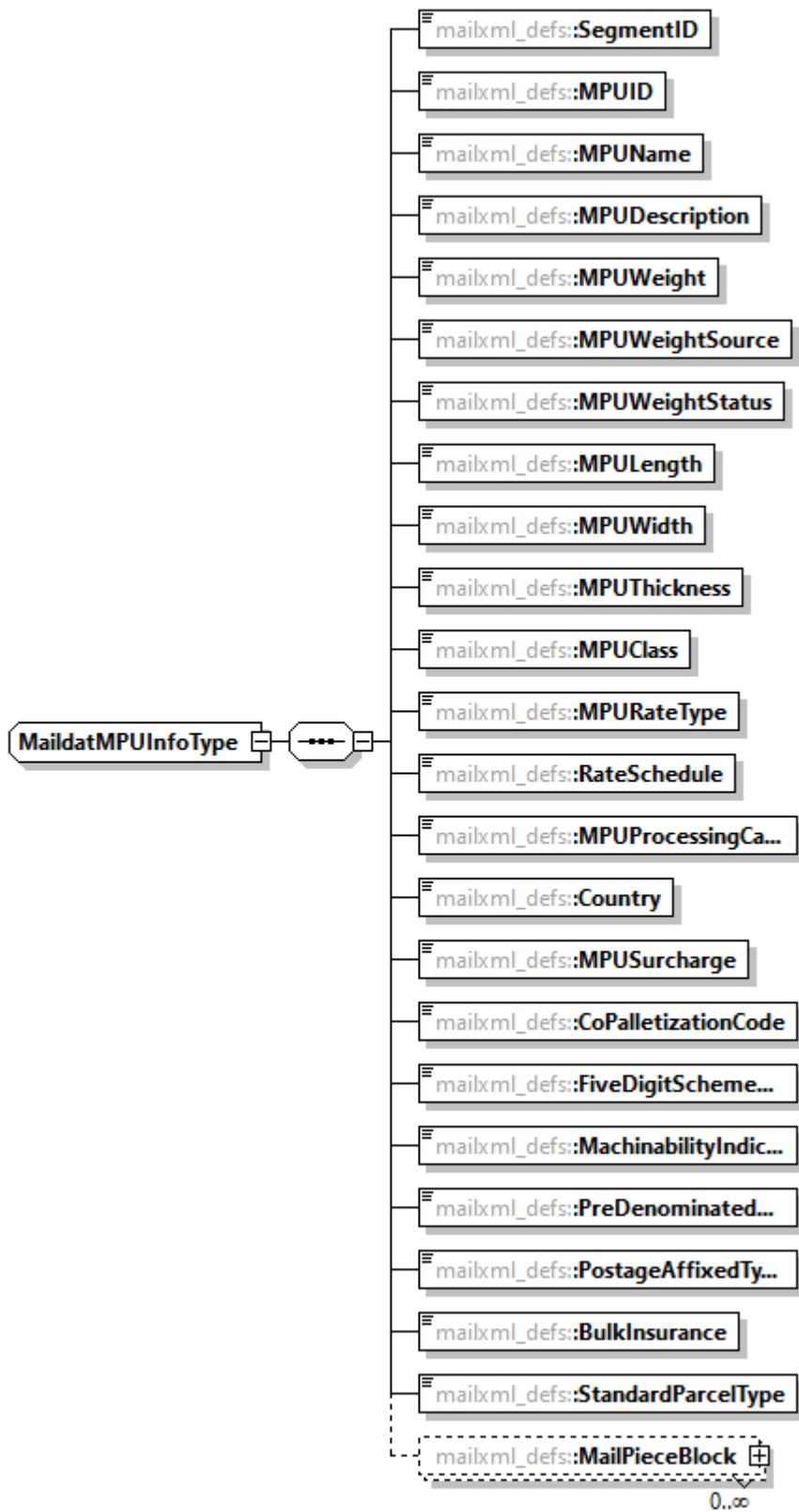
diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **maildatContainerIDType**

diagram	 <p>The diagram illustrates the structure of the <b>maildatContainerIDType</b> complex type. It is represented by a rounded rectangle with a small square handle on its right side. A line connects this handle to a larger box representing the type's content. This box is divided into two main sections. The top section is labeled <b>attributes</b> and contains a group of four attributes, each in a dashed box: <b>mailxml_defs:UserLicenseCode</b>, <b>mailxml_defs:MaildatJobID</b>, <b>mailxml_defs:ContainerID</b>, and <b>mailxml_defs:UniqueContainer...</b>. The bottom section contains the text: "Group of attributes identifying an original or sibling container".</p>
namespace	<a href="http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs">http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs</a>

complexType MaildatMPUInfoType

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
-----------	---

complexType **mailPieceldType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **MailXMLContainerIDType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **mailxmlDetailType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **measurementType**

diagram	<pre> graph LR     A[measurementType] --&gt; B[...]     B --&gt; C[choice]     C --&gt; D[mailxml_defs::Value]     C --&gt; E[mailxml_defs::RangeMin]     C --&gt; F[mailxml_defs::RangeMax]             </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **MIDType**

diagram	<pre> graph LR     A[MIDType] --&gt; B[...]     B --&gt; C[choice]     C --&gt; D[mailxml_defs::MID6]     C --&gt; E[mailxml_defs::MID9]             </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **MPSRBlockType**

diagram	<pre> graph LR     A[MPSRBlockType] --&gt; B[...]     B --&gt; C[choice]     C --&gt; D[mailxml_defs::ContainerID 1..∞]     C --&gt; E[mailxml_defs::PSRBlock 1..∞]             </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **MPSVisIncludedInScanRecFlagType**

diagram	<pre> graph LR     A[MPSVisIncludedInScanRecFlagT...] --&gt; B[...]     B --&gt; C[choice]     C --&gt; D[mailxml_defs::IMcb]     C --&gt; E[mailxml_defs::IMtb]     C --&gt; F[mailxml_defs::BundleID]             </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

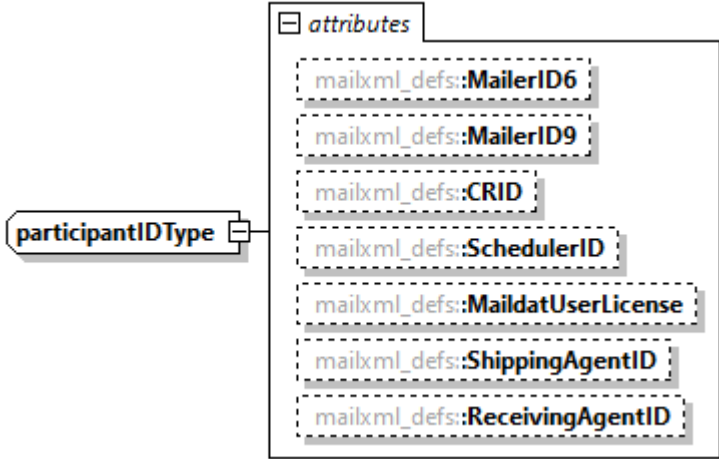
complexType **MPSVisResultOptionsType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

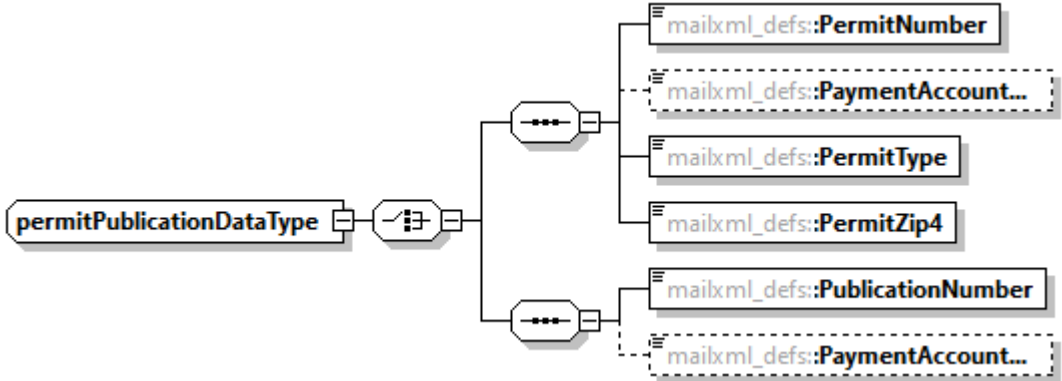
complexType **palletHandoffInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

# complexType **participantIDType**

diagram	 <p>The diagram shows a complex type named <b>participantIDType</b>. It has a list of attributes under the heading <b>attributes</b>. The attributes are:</p> <ul style="list-style-type: none"> <li>mailxml_defs:MailerID6</li> <li>mailxml_defs:MailerID9</li> <li>mailxml_defs:CRID</li> <li>mailxml_defs:SchedulerID</li> <li>mailxml_defs:MaildatUserLicense</li> <li>mailxml_defs:ShippingAgentID</li> <li>mailxml_defs:ReceivingAgentID</li> </ul>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

# complexType **permitPublicationDataType**

diagram	 <p>The diagram shows a complex type named <b>permitPublicationDataType</b>. It has a sequence of elements. The first element is a choice between two groups:</p> <ul style="list-style-type: none"> <li>Group 1 (indicated by a dashed line): <ul style="list-style-type: none"> <li>mailxml_defs:PermitNumber</li> <li>mailxml_defs:PaymentAccount...</li> </ul> </li> <li>Group 2 (indicated by a solid line): <ul style="list-style-type: none"> <li>mailxml_defs:PermitType</li> <li>mailxml_defs:PermitZip4</li> </ul> </li> </ul> <p>The second element is a choice between two groups:</p> <ul style="list-style-type: none"> <li>Group 3 (indicated by a dashed line): <ul style="list-style-type: none"> <li>mailxml_defs:PublicationNumber</li> <li>mailxml_defs:PaymentAccount...</li> </ul> </li> <li>Group 4 (indicated by a solid line): <ul style="list-style-type: none"> <li>mailxml_defs:PublicationNumber</li> <li>mailxml_defs:PaymentAccount...</li> </ul> </li> </ul>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **pickupApptBlockResponseType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **postageStatementSummaryType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

complexType **PSRBlockType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs



### complexType **queryErrorType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

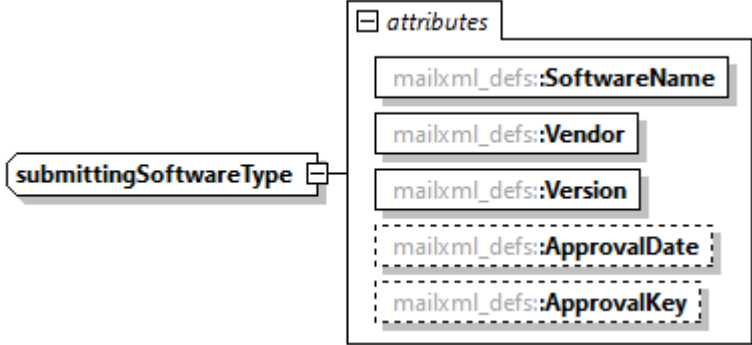
### complexType **returnInfoType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs


### complexType **scanEventType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **submittingSoftwareType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs

### complexType **zipCode**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	extension of <b>mailxml_base:ns09</b>

### simpleType **bundleScanTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

### simpleType **containerDiscrepancyCategoryType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

### simpleType **containerScanStateType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>
annotation	documentation Scan data for Container Scan States

### simpleType **containerStatusType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **countTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **einductionDataSourceType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **fullServiceComplianceIndicatorType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	<b>mailxml_base:yesNo</b>

simpleType **MPSQueryType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>
annotation	documentation Mail Piece Scan Query Type

simpleType **MPSSStateType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>
annotation	documentation Scan States for Mail Piece Scan Data

simpleType **pieceScanEventTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **problemCategoryType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **problemTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **reasonCodeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>

simpleType **retrieveDataByType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_defs
type	restriction of <b>xs:string</b>