

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_dd_26.3.xsd	8

Schema mailxml_dd_26.3.xsd

schema location: [.\XSDs\mailxml_dd_26.3.xsd](#)
attributeFormDefault: **qualified**
elementFormDefault: **qualified**
targetNamespace: http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

Elements

[**AddressCorrection**](#)
[**AddressCorrectionDelivery**](#)
[**AddressCorrectionNotification**](#)
[**AddressCorrectionQueryRequest**](#)
[**AddressCorrectionQueryResponse**](#)
[**ByForConflictDelivery**](#)
[**ByForConflictNotification**](#)
[**ByForConflictQueryRequest**](#)
[**ByForConflictQueryResponse**](#)
[**ContainerVisibilityEntry**](#)
[**DataQualityVerificationReportDelivery**](#)
[**DataQualityVerificationReportNotification**](#)
[**DataQualityVerificationReportQueryRequest**](#)
[**DataQualityVerificationReportQueryResponse**](#)
[**DeliveryPointValidation**](#)
[**DeliveryPointValidationDelivery**](#)
[**DeliveryPointValidationNotification**](#)
[**DeliveryPointValidationQueryRequest**](#)
[**DeliveryPointValidationQueryResponse**](#)
[**IMbMailpieceScanData**](#)
[**NixieDetail**](#)
[**NixieDetailDelivery**](#)
[**NixieDetailNotification**](#)
[**NixieDetailQueryRequest**](#)
[**NixieDetailQueryResponse**](#)
[**NonComplianceDataWithPostageOwedQueryRequest**](#)
[**NonComplianceDataWithPostageOwedReportDelivery**](#)
[**NonComplianceDataWithPostageOwedReportNotification**](#)
[**NonComplianceDataWithPostageOwedReportQueryResponse**](#)

Complex types

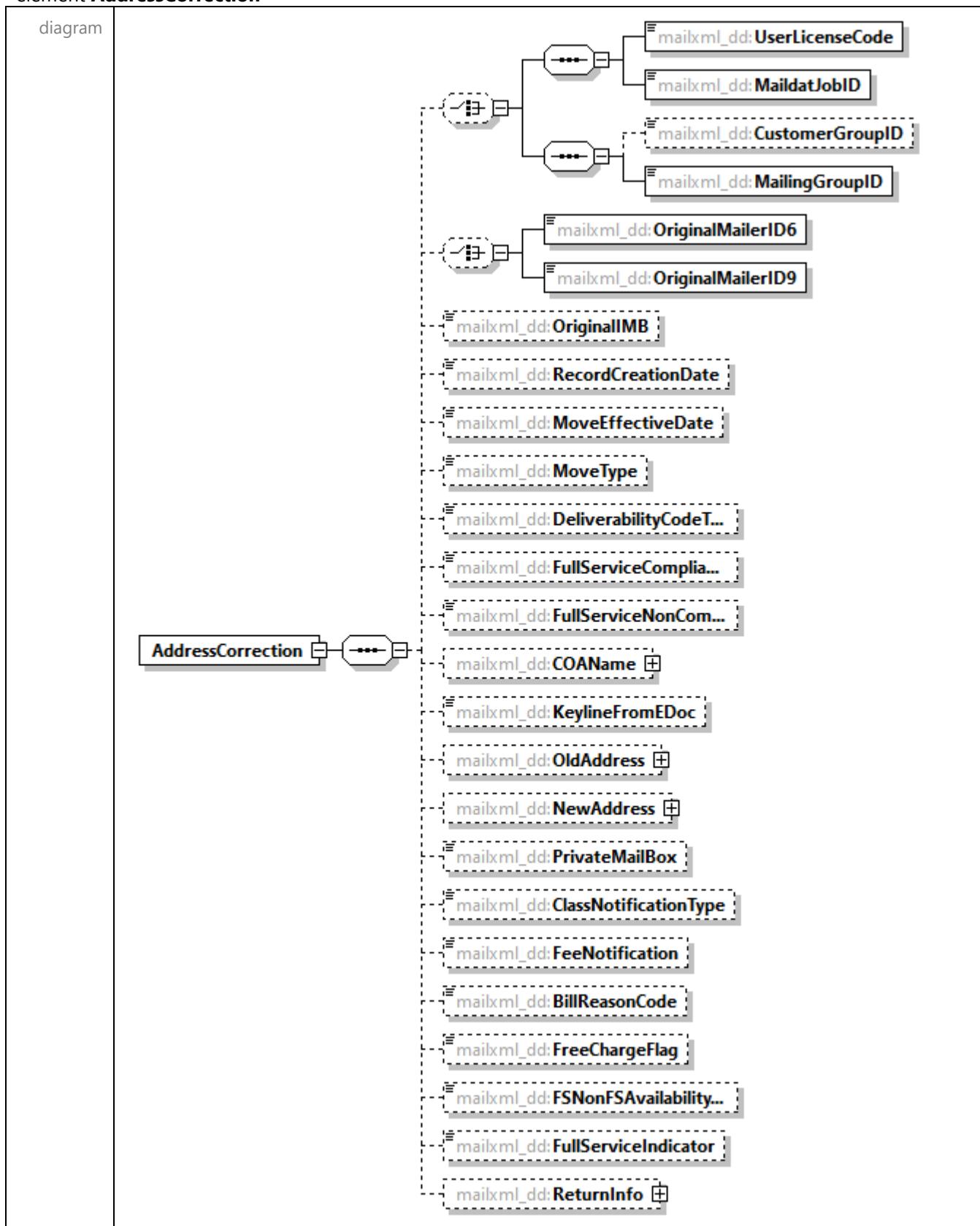
[**addressCorrectionAddressType**](#)
[**byForConflictType**](#)
[**clockStartedType**](#)
[**dqrContainerInfoType**](#)
[**foreignAddressType**](#)
[**manifestScanEventDetailType**](#)
[**manifestScanNotificationDataType**](#)
[**manifestScanQueryType**](#)
[**MPSNotificationDataType**](#)
[**MPSRResponseBlockType**](#)
[**MPSVisScanQueryType**](#)
[**newAddressCorrectionAddressType**](#)
[**nonComplianceDataWithPostageOwedReportType**](#)
[**PSRResponseBlockType**](#)
[**unManifestedScanEventDetailType**](#)
[**verificationErrorType**](#)

Simple types

[**addressCorrectionMoveType**](#)
[**addressTypeType**](#)
[**containerScanStateType**](#)
[**deliverabilityCodeType**](#)
[**eDocTypeType**](#)
[**primarySecondaryIndicatorType**](#)
[**verificationErrorTypeType**](#)
[**verificationWarningTypeType**](#)

[ScanSTCCount](#)
[ScanSTCReconciliationDelivery](#)
[ScanSTCReconciliationNotification](#)
[ScanSTCReconciliationQueryRequest](#)
[ScanSTCReconciliationQueryResponse](#)
[StartTheClockBEMUBlock](#)
[StartTheClockDropShipOrOrigin](#)
[StartTheClockPlantLoadBlock](#)

element **AddressCorrection**



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

element **AddressCorrectionDelivery**

diagram	<pre> classDiagram class AddressCorrectionDelivery { grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage } class SubmittingParty class SubmittingSoftware class DataRecipient class PushMessageID class AddressCorrection AddressCorrectionDelivery "1..∞" --> SubmittingParty AddressCorrectionDelivery "1..∞" --> SubmittingSoftware AddressCorrectionDelivery "1..∞" --> DataRecipient AddressCorrectionDelivery "1..∞" --> PushMessageID AddressCorrectionDelivery "1..∞" --> AddressCorrection </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Delivery of address correction delivery data from USPS.</p>

element **AddressCorrectionNotification**

diagram	<pre> classDiagram class AddressCorrectionNotification { <<Notification from UPS that address correction data is ready to pick up.>> } class SubmittingParty class SubmittingSoftware class CRID class PushMessageID class UserLicenseCode class MaildatJobID class CustomerGroupID class MailingGroupID class FSNonFSAvailability... class NotificationDate class AvailableRecordCou... AddressCorrectionNotification "1" --> "1..>" SubmittingParty AddressCorrectionNotification "1" --> "1..>" SubmittingSoftware AddressCorrectionNotification "1" --> "1..>" CRID AddressCorrectionNotification "1" --> "1..>" PushMessageID PushMessageID "*" --> "1..>" UserLicenseCode PushMessageID "*" --> "1..>" MaildatJobID PushMessageID "*" --> "1..>" CustomerGroupID PushMessageID "*" --> "1..>" MailingGroupID PushMessageID "*" --> "1..>" FSNonFSAvailability... PushMessageID "*" --> "1..>" NotificationDate PushMessageID "*" --> "1..>" AvailableRecordCou... </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Notification from UPS that address correction data is ready to pick up.</p>

element **AddressCorrectionQueryRequest**

diagram	<pre> classDiagram class AddressCorrectionQueryReq... { <<Query request for full service address correction data.>> } class SubmittingParty class SubmittingSoftware class SubmitterTrackingID class OwningParty class PieceRequest class retrieveDataBy AddressCorrectionQueryReq... "1" --> "1..>" SubmittingParty AddressCorrectionQueryReq... "1" --> "1..>" SubmittingSoftware AddressCorrectionQueryReq... "1" --> "1..>" SubmitterTrackingID AddressCorrectionQueryReq... "1" --> "1..>" OwningParty AddressCorrectionQueryReq... "*" --> "1..>" PieceRequest PieceRequest "*" --> "1..>" retrieveDataBy </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Query request for full service address correction data.</p>

element **AddressCorrectionQueryResponse**

diagram	<p>The diagram illustrates the structure of the AddressCorrectionQueryResponse element. It consists of a main class box labeled AddressCorrectionQueryRespo... which contains a brief description: "Response to the query request for full service address correction data." A dependency relationship is shown from this class to another class box labeled attributes. Inside the attributes box, there is a group header grp mailxml_defs:LargeTransa... followed by five solid boxes: mailxml_defs:MessageGroupID, mailxml_defs:TotalMessageCount, mailxml_defs:MessageSerialNu..., mailxml_defs:TransmittedRecor..., and mailxml_defs:TotalRecordsAcro.... Below these is a dashed box mailxml_defs:LastMessage. From the attributes box, a dependency relationship leads to a class box labeled mailxml_dd:TrackingID, which is enclosed in a dashed box. This is followed by another dependency relationship to a class box labeled mailxml_dd:SubmitterTrackingID, also enclosed in a dashed box. Finally, a dependency relationship leads to two class boxes: mailxml_dd:QueryResults and mailxml_defs:QueryError, each enclosed in a dashed box. A note below mailxml_defs:QueryError states: "Error issued when the query data cannot be provided."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Response to the query request for full service address correction data.

element **ByForConflictDelivery**

diagram	<p>The diagram shows the structure of the ByForConflictDelivery element. It consists of a main class box labeled ByForConflictDelivery with a description: "Delivery of by / for conflict information from USPS." A dashed line connects it to a larger box labeled attributes. Inside the attributes box, there is a group box labeled grp mailxml_defs:LargeTransa... containing several attributes: MessageGroupID, TotalMessageCount, MessageSerialNu..., TransmittedRecor..., TotalRecordsAcro..., and LastMessage. Below this group is another group box labeled mailxml_dd:SubmittingParty. A multiplicity of 1..∞ is shown at the bottom right of the SubmittingParty box, indicating that multiple SubmittingParty instances can be associated with a single ByForConflictDelivery instance. Other attributes listed in the mailxml_dd group include SubmittingSoftware, DataRecipient, PushMessageID, and ByForConflict.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Delivery of by / for conflict information from USPS.

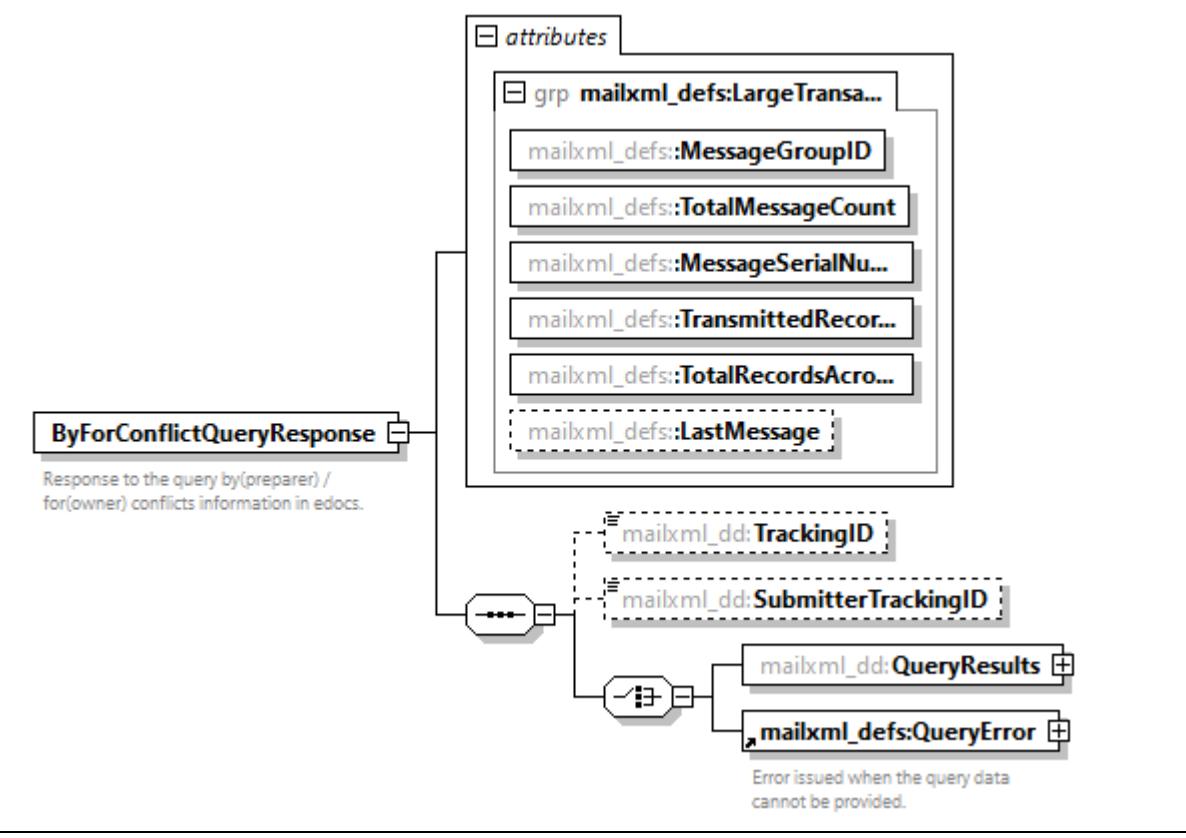
element **ByForConflictNotification**

diagram	<pre> sequenceDiagram participant BCN as ByForConflictNotification participant SP as SubmittingParty participant SS as SubmittingSoftware participant PMI as PushMessageID participant UL as UserLicenseCode participant MJID as MaildatJobID participant CGID as CustomerGroupID participant MGID as MailingGroupID participant FNFSAV as FSNonFSAvailability participant ND as NotificationDate participant ARC as AvailableRecordCount BCN->>SP: BCN->>SS: BCN->>PMI: BCN->>UL: BCN->>MJID: BCN->>CGID: BCN->>MGID: BCN->>FNFSAV: BCN->>ND: BCN->>ARC: </pre> <p>Notification from USPS that a by / for conflict resolution is ready for pickup.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Notification from USPS that a by / for conflict resolution is ready for pickup.</p>

element **ByForConflictQueryRequest**

diagram	<pre> sequenceDiagram participant BCQR as ByForConflictQueryRequest participant SP as SubmittingParty participant SS as SubmittingSoftware participant STID as SubmitterTrackingID participant UL as UserLicenseCode participant MJID as MaildatJobID participant CGID as CustomerGroupID participant MGID as MailingGroupID participant RD as retrieveDataBy BCQR->>SP: BCQR->>SS: BCQR->>STID: BCQR->>UL: BCQR->>MJID: BCQR->>CGID: BCQR->>MGID: BCQR->>RD: </pre> <p>Query request for by(preparer) / for(owner) conflicts information in edocs.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Query request for by(preparer) / for(owner) conflicts information in edocs.</p>

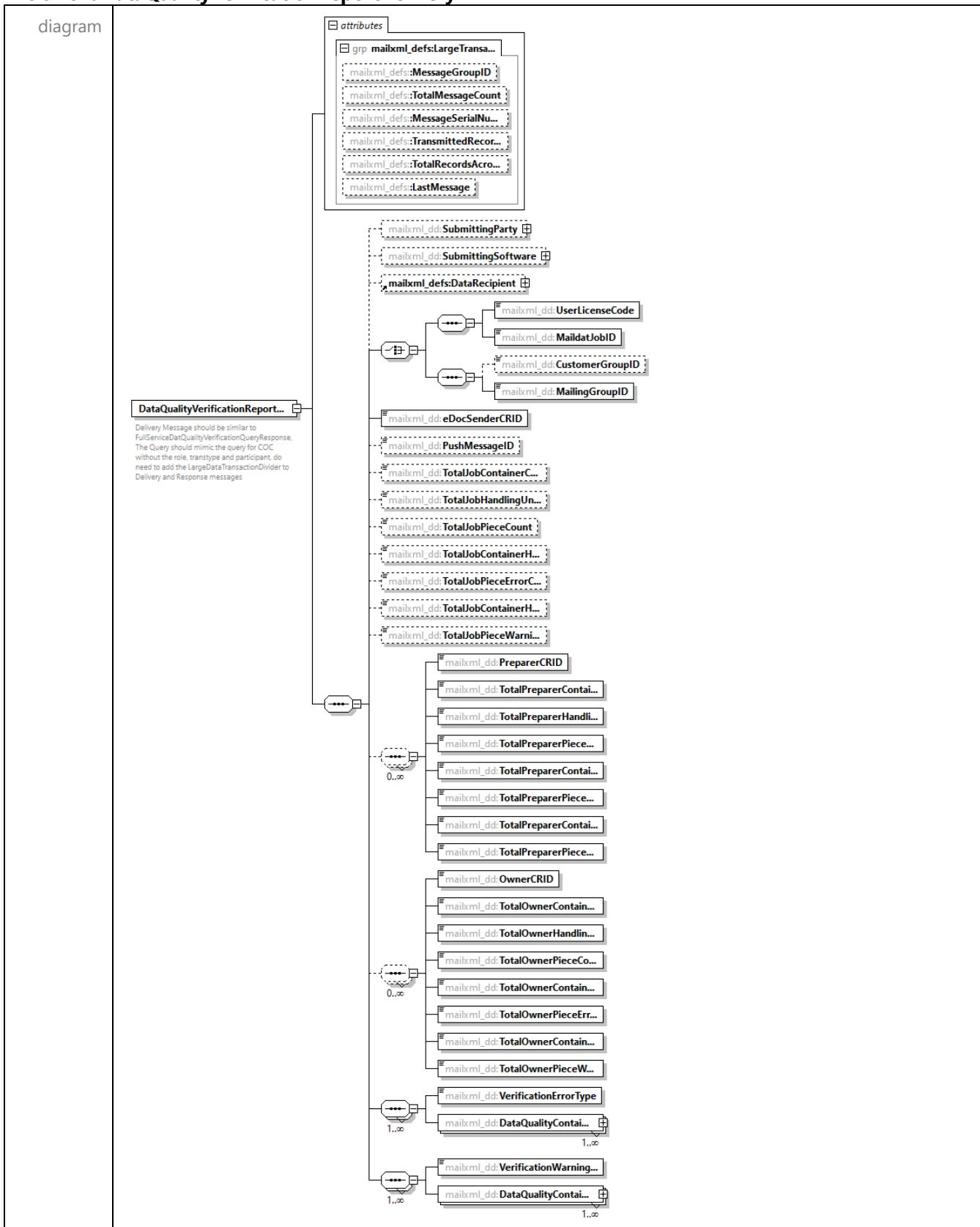
element **ByForConflictQueryResponse**

diagram	 <pre> classDiagram class ByForConflictQueryResponse { <<Response to the query by(preparer) / for(owner) conflicts information in edocs.>> attributes grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage } ByForConflictQueryResponse "1" --> "1" mailxml_dd:TrackingID ByForConflictQueryResponse "1" --> "1" mailxml_dd:SubmitterTrackingID mailxml_dd:SubmitterTrackingID "*" --> "1" mailxml_dd:QueryResults mailxml_dd:SubmitterTrackingID "*" --> "1" mailxml_defs:QueryError } class mailxml_defs { LargeTransa... MessageGroupID TotalMessageCount MessageSerialNu... TransmittedRecor... TotalRecordsAcro... LastMessage } class mailxml_dd { TrackingID SubmitterTrackingID QueryResults QueryError } </pre> <p>The diagram illustrates the structure of the ByForConflictQueryResponse element. It features a main class box labeled ByForConflictQueryResponse with a note below it: "Response to the query by(preparer) / for(owner) conflicts information in edocs.". A line connects this class to a box labeled "attributes". Inside the "attributes" box are several items: grp mailxml_defs:LargeTransa..., mailxml_defs:MessageGroupID, mailxml_defs:TotalMessageCount, mailxml_defs:MessageSerialNu..., mailxml_defs:TransmittedRecor..., mailxml_defs:TotalRecordsAcro..., and mailxml_defs:LastMessage. Below the attributes is another line connecting to a dashed box labeled "mailxml_dd". Inside this dashed box are TrackingID and SubmitterTrackingID. From SubmitterTrackingID, two lines lead to QueryResults and QueryError.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Response to the query by(preparer) / for(owner) conflicts information in edocs.</p>

element **ContainerVisibilityEntry**

diagram	<p>The diagram illustrates the structure of the ContainerVisibilityEntry element. It consists of several components:</p> <ul style="list-style-type: none">ContainerVisibilityEntry: The main container, represented by a rectangle with a boundary.Associations: Lines connecting the main container to various sub-elements. There are two primary association points on the right side of the main container's boundary.Sub-elements: These are grouped into dashed boxes and include:<ul style="list-style-type: none">UserLicenseCode, MaildataJobID, CustomerGroupID, and MailingGroupID (under the top dashed box).ConsigneeApptID (under the second dashed box).LogicalIndicator (under the third dashed box).CSAID (under the fourth dashed box).ContainerID, ParentContainerID, and SiblingContainerID (under the fifth dashed box).ContainerType (under the sixth dashed box).ScanEvent (under the seventh dashed box, preceded by a plus sign, indicating it's an optional element).IMcb, IMtb, and IMpb (under the eighth dashed box, preceded by a plus sign).FullServiceComplia..., FullServiceNonCom..., FSNonFSAvailability..., and FullServiceIndicator (under the ninth dashed box).
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

element DataQualityVerificationReportDelivery



namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Delivery Message should be similar to FullServiceDatQualityVerificationQueryResponse, The Query should mimic the query for COC without the role, transtype and participant, do need to add the LargeDataTransactionDivider to Delivery and Response messages</p>

element DataQualityVerificationReportNotification

diagram	<pre> classDiagram class DataQualityVerificationReportNotification { <<Notification from USPS that a full service data quality verification report is ready for pickup.>> } class SubmittingParty class SubmittingSoftware class PushMessageID class UserLicenseCode class MaildatJobID class CustomerGroupID class eDocSenderCRID class VerificationErrorType class VerificationWarning class FSNonFSAvailability class NotificationDate class AvailableRecordCou DataQualityVerificationReportNotification "1..∞" --> SubmittingParty DataQualityVerificationReportNotification "1..∞" --> SubmittingSoftware DataQualityVerificationReportNotification "1..∞" --> PushMessageID DataQualityVerificationReportNotification "1..∞" --> UserLicenseCode DataQualityVerificationReportNotification "1..∞" --> MaildatJobID DataQualityVerificationReportNotification "1..∞" --> CustomerGroupID DataQualityVerificationReportNotification "1..∞" --> eDocSenderCRID DataQualityVerificationReportNotification "1..∞" --> VerificationErrorType DataQualityVerificationReportNotification "1..∞" --> VerificationWarning DataQualityVerificationReportNotification "1..∞" --> FSNonFSAvailability DataQualityVerificationReportNotification "1..∞" --> NotificationDate DataQualityVerificationReportNotification "1..∞" --> AvailableRecordCou </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Notification from USPS that a full service data quality verification report is ready for pickup.</p>

element **DataQualityVerificationReportQueryRequest**

diagram	<p>The diagram illustrates the structure of the DataQualityVerificationReportQueryRequest element. It consists of the following components:</p> <ul style="list-style-type: none">DataQualityVerificationReport...: The main class, represented by a rectangle with a dashed border.SubmittingParty, SubmittingSoftware, and SubmitterTrackingID: Associated with DataQualityVerificationReport... via dashed lines.UserLicenseCode, MaildatJobID, CustomerGroupID, and MailingGroupID: Associated with DataQualityVerificationReport... via dashed lines.eDocSenderCRID: Associated with DataQualityVerificationReport... via a dashed line.VerificationErrorType and VerificationWarning...: Associated with DataQualityVerificationReport... via dashed lines.retrieveDataBy: Associated with DataQualityVerificationReport... via a dashed line. <p>Relationships are indicated by dashed lines connecting the main class to its attributes. Some attributes have multiplicity markers (e.g., 1..∞) indicating they can appear multiple times.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Query request for full service data quality reports.

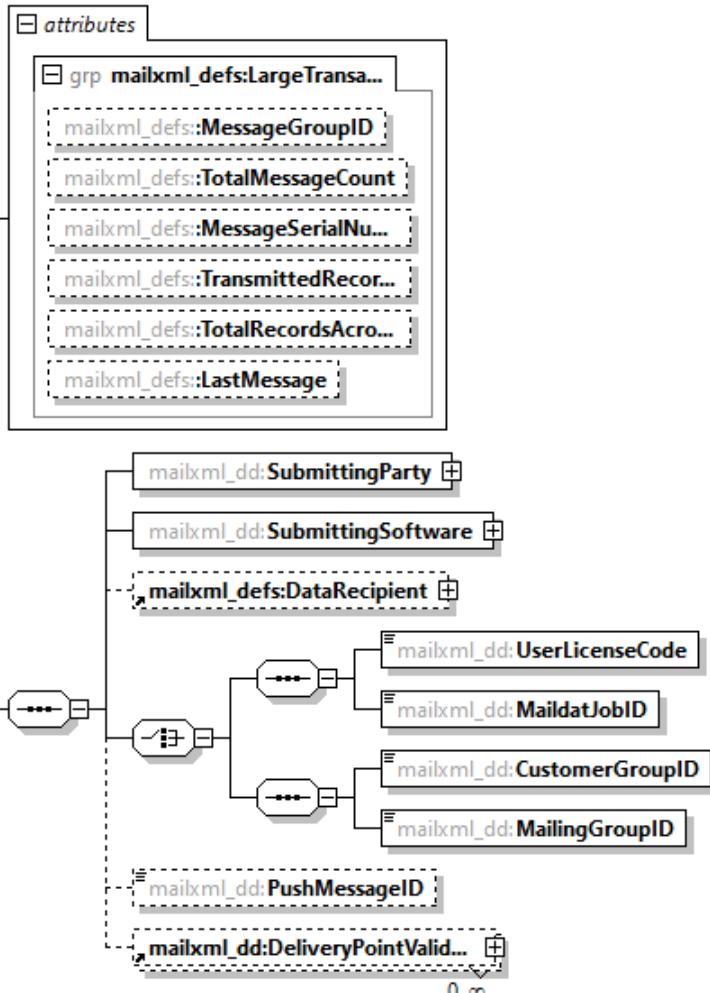
element **DataQualityVerificationReportQueryResponse**

diagram	<pre> classDiagram class DataQualityVerificationReportQueryResponse { grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage } DataQualityVerificationReportQueryResponse "Response to the Request for Full Service Data Quality Verification Reports" DataQualityVerificationReportQueryResponse < --> mailxml_dd:TrackingID DataQualityVerificationReportQueryResponse < --> mailxml_dd:SubmitterTrackingID mailxml_dd:TrackingID < --> mailxml_dd:QueryResults mailxml_dd:SubmitterTrackingID < --> mailxml_defs:QueryError mailxml_defs:QueryError "Error issued when the query data cannot be provided." </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Response to the Request for Full Service Data Quality Verification Reports</p>

element **DeliveryPointValidation**

diagram	<pre>graph TD; DPV[DeliveryPointValidation] --> O1[OriginalMailerID6]; DPV --> O2[OriginalMailerID9]; O1 --- ULC[UserLicenseCode]; O2 --- MJID[MaildatJobID]; O2 --- CGID[CustomerGroupID]; O2 --- MGID[MailingGroupID]; ULC --- IDP[InvalidDeliveryPoint...]; ULC --- PSI[PrimarySecondary...]; ULC --- DPVDR[DPVDataRecipientC...]; ULC --- FSC[FullServiceComplia...]; ULC --- FSFSA[FullServiceNonCom...]; subgraph DashedBox []; FSIndicator[FullServiceIndicator]; FSNonFSA[FSNonFSAvailability...]; end; ULC --- RI[ReturnInfo]</pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

element **DeliveryPointValidationDelivery**

diagram	 <pre> classDiagram class DeliveryPointValidationDelivery { grp mailxml_defs:LargeTrans mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage } class SubmittingParty class SubmittingSoftware class DataRecipient class UserLicenseCode class MaildatJobID class CustomerGroupID class MailingGroupID class PushMessageID class DeliveryPointValid... DeliveryPointValidationDelivery "1" -- "0..∞" SubmittingParty DeliveryPointValidationDelivery "1" -- "0..∞" SubmittingSoftware DeliveryPointValidationDelivery "1" -- "0..∞" DataRecipient DataRecipient "*" -- "0..∞" UserLicenseCode DataRecipient "*" -- "0..∞" MaildatJobID DataRecipient "*" -- "0..∞" CustomerGroupID DataRecipient "*" -- "0..∞" MailingGroupID DataRecipient "*" -- "0..∞" PushMessageID DataRecipient "*" -- "0..∞" DeliveryPointValid... </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Delivery of Delivery Point Validation by USPS.</p>

element **DeliveryPointValidationNotification**

diagram	<p>The diagram illustrates the structure of the DeliveryPointValidationNotification element. It begins with a central node labeled DeliveryPointValidationNotific.... This node has two outgoing connections: one to the left and one to the right. The left connection leads to a dashed box containing SubmittingParty, SubmittingSoftware, and CRID. The right connection leads to another dashed box containing PushMessageID. From this point, two more dashed boxes branch out: one for UserLicenseCode and MaildatJobID, and another for CustomerGroupID and MailingGroupID. Finally, three solid boxes at the bottom represent FSNonFSAvailability..., NotificationDate, and AvailableRecordCou....</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Notification by USPS that Delivery Point Validation is ready for pickup.

element **DeliveryPointValidationQueryRequest**

diagram	<p>The diagram illustrates the structure of the DeliveryPointValidationQueryRequest element. It begins with a central node labeled DeliveryPointValidationQueryR.... This node has two outgoing connections: one to the left and one to the right. The left connection leads to a dashed box containing SubmittingParty, SubmittingSoftware, SubmitterTrackingID, and RequestorCRID. The right connection leads to another dashed box containing OriginalMailerID6 and OriginalMailerID9. From this point, two more dashed boxes branch out: one for UserLicenseCode and MaildatJobID, and another for CustomerGroupID and MailingGroupID. Finally, three solid boxes at the bottom represent DateRange and retrieveDataBy.</p>
---------	---

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Query request for USPS for delivery point validation.</p>

element **DeliveryPointValidationQueryResponse**

diagram	<pre> classDiagram class DeliveryPointValidationQueryR { <<Response to the Query request for USPS for delivery point validation.>> } class grp { mailxml_defs:LargeTransa... } class mailxml_defs { MessageGroupID TotalMessageCount MessageSerialNu... TransmittedRecor... TotalRecordsAcro... LastMessage } class mailxml_dd { TrackingID SubmitterTrackingID QueryResults QueryError } DeliveryPointValidationQueryR "1" -- "1" grp : DeliveryPointValidationQueryR "1" -- "1" mailxml_defs : DeliveryPointValidationQueryR "*" -- "1" mailxml_dd : mailxml_defs "1" -- "1" mailxml_dd : mailxml_dd "1" -- "1" QueryResults : mailxml_dd "1" -- "1" QueryError : </pre> <p>The diagram illustrates the structure of the DeliveryPointValidationQueryResponse element. It consists of the following components:</p> <ul style="list-style-type: none"> DeliveryPointValidationQueryR: The main element, described as "Response to the Query request for USPS for delivery point validation." grp: A group containing attributes for large transactions. mailxml_defs: A group containing attributes for message identification and tracking. mailxml_dd: A group containing attributes for tracking and submission tracking. QueryResults: An association with the mailxml_dd group. QueryError: An association with the mailxml_dd group. <p>Associations are shown between DeliveryPointValidationQueryR and grp, DeliveryPointValidationQueryR and mailxml_defs, DeliveryPointValidationQueryR and mailxml_dd, mailxml_defs and mailxml_dd, and mailxml_dd and QueryResults.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Response to the Query request for USPS for delivery point validation.</p>

element **IMbMailpieceScanData**

diagram	<p>The diagram illustrates the structure of the IMbMailpieceScanData element. It starts with a class box labeled IMbMailpieceScanData, which has a multiplicity of *** (three asterisks) at its association end. This connects to a dashed box containing four attributes: UserLicenseCode, MaildataJobID, CustomerGroupID, and MailingGroupID. From this dashed box, another association line leads to a class box labeled MailBundleCount. This association also has a multiplicity of *** at its end. From MailBundleCount, two association lines branch out to two more class boxes: MPSCount and IMcbAndIMtbPiec.... Both of these have a multiplicity of 1..∞ at their ends. From IMcbAndIMtbPiec..., three association lines lead to three separate class boxes: IMcbPieceScanInfo, IMtbPieceScanInfo, and IMbScanRec. Each of these three class boxes also has a multiplicity of 1..∞ at its end.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

element **NixieDetail**

diagram	<pre> classDiagram class NixieDetail { <<Full service nixie detail.>> UserLicenseCode MaildatJobID CustomerGroupID MailingGroupID OriginalMailerID6 OriginalMailerID9 OriginalIMB RecordCreationDate ActionCode ParsedAddressOnPi... OnPieceCityStateZip ReturnedToAddress ReturnedToCityStat... ReasonCode KeylineFromEDoc ClassNotificationType FeeNotification FullServiceComplia... FullServiceNonCom... BillReasonCode FreeChargeFlag FSNonFSAvailability... FullServiceIndicator ReturnInfo } class FullServiceNixieDetail { <<Full service nixie detail.>> } NixieDetail < -- FullServiceNixieDetail </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

annotation	documentation Full service nixie detail.
------------	---

element NixieDetailDelivery

diagram	<pre> classDiagram class NixieDetailDelivery { grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage } class SubmittingParty class SubmittingSoftware class DataRecipient class PushMessageID class NixieDetail NixieDetailDelivery "1..∞" --> SubmittingParty NixieDetailDelivery "1..∞" --> SubmittingSoftware NixieDetailDelivery "1..∞" --> DataRecipient NixieDetailDelivery "1..∞" --> PushMessageID NixieDetailDelivery "1..∞" --> NixieDetail </pre> <p>Delivery from USPS of full service nixie detail.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Delivery from USPS of full service nixie detail.

element **NixieDetailNotification**

diagram	<pre> graph LR ND[NixieDetailNotification] --- NP(()) NP --- NP_C[...] NP_C --- SP[mailxml_dd:SubmittingParty] NP_C --- SS[mailxml_dd:SubmittingSoftware] NP_C --- CRID[mailxml_dd:CRID] NP_C --- PMID[mailxml_dd:PushMessageID] NP_C --- NP_C2[...] NP_C2 --- UL[mailxml_dd:UserLicenseCode] NP_C2 --- MJID[mailxml_dd:MaildatJobID] NP_C2 --- CGID[mailxml_dd:CustomerGroupID] NP_C2 --- MGID[mailxml_dd:MailingGroupID] NP_C2 --- FSNA[mailxml_dd:FSNonFSAvailability...] NP_C2 --- ND[mailxml_dd:NotificationDate] NP_C2 --- ARCC[mailxml_dd:AvailableRecordCou...] </pre> <p>NixieDetailNotification</p> <p>Notification by USPS that full service nixie detail is ready for pickup.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Notification by USPS that full service nixie detail is ready for pickup.</p>

element **NixieDetailQueryRequest**

diagram	<pre> graph LR NDQ[NixieDetailQueryRequest] --- NP(()) NP --- NP_C[...] NP_C --- SP[mailxml_dd:SubmittingParty] NP_C --- SS[mailxml_dd:SubmittingSoftware] NP_C --- STID[mailxml_dd:SubmitterTrackingID] NP_C --- RCID[mailxml_dd:RequestorCRID] NP_C --- NP_C2[...] NP_C2 --- OMID6[mailxml_dd:OriginalMailerID6] NP_C2 --- OMID9[mailxml_dd:OriginalMailerID9] NP_C2 --- PR[PieceRequest] PR --- RD[retrieveDataBy] PR --- RD1["1..∞"] </pre> <p>NixieDetailQueryRequest</p> <p>Query request for full service nixie detail.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Query request for full service nixie detail.</p>

element NixieDetailQueryResponse

diagram	<p>The diagram illustrates the structure of the NixieDetailQueryResponse element. It features a main class box labeled NixieDetailQueryResponse with a brief description: "Response to the Query request for full service nixie detail." Below the class box is a detailed view of its attributes. The attributes are grouped under a header "attributes". Inside this group, there is a sub-group "grp mailxml_defs:LargeTransa..." containing several attributes: MessageGroupID, TotalMessageCount, MessageSerialNu..., TransmittedRecor..., TotalRecordsAcro..., and LastMessage. Below this group, there is another sub-group "mailxml_dd: TrackingID" containing TrackingID and SubmitterTrackingID. A dashed line connects the TrackingID attribute to a sequence of three association points. From the second point, a solid line leads to a box labeled QueryResults. From the third point, a dashed line leads to a box labeled QueryError. A note below QueryError states: "Error issued when the query data cannot be provided."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Response to the Query request for full service nixie detail.

element **NonComplianceDataWithPostageOwedQueryRequest**

diagram	<p>NonComplianceDataWithPostageOwedQueryRequest Query request for full service Non Compliance Data reports.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation Query request for full service Non Compliance Data reports.

element **NonComplianceDataWithPostageOwedReportDelivery**

diagram	<pre> classDiagram class NonComplianceDataWithPostageOwedReportDelivery { <<Delivery of by / for conflict information from USPS.>> attributes grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage mailxml_dd:SubmittingParty mailxml_dd:SubmittingSoftware mailxml_defs:DataRecipient mailxml_dd:PushMessageID mailxml_dd:NonComplianceDa... mailxml_dd:FSNonFSAvailability... mailxml_dd:FullServiceIndicator operations } </pre> <p>The diagram shows the structure of the NonComplianceDataWithPostageOwedReportDelivery element. It consists of several attributes and a collection of NonComplianceData objects.</p> <ul style="list-style-type: none"> Attributes: <ul style="list-style-type: none"> grp mailxml_defs:LargeTransa... (grouped) <ul style="list-style-type: none"> mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage mailxml_dd:SubmittingParty mailxml_dd:SubmittingSoftware mailxml_defs:DataRecipient mailxml_dd:PushMessageID mailxml_dd:NonComplianceDa... (with multiplicity 1..∞) mailxml_dd:FSNonFSAvailability... mailxml_dd:FullServiceIndicator Operations: None <p>Delivery of by / for conflict information from USPS.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Delivery of by / for conflict information from USPS.</p>

element **NonComplianceDataWithPostageOwedReportNotification**

diagram	<p>The diagram illustrates the structure of the NonComplianceDataWithPostageOwedReportNotification element. It begins with a central node labeled "NonComplianceDataWithPostageOwedReportNotification". A dashed line connects this node to a group of nested nodes. This group contains "SubmittingParty", "SubmittingSoftware", and "PushMessageID". Another dashed line connects "PushMessageID" to a sequence of nodes: "UserLicenseCode", "MaildatJobID", "CustomerGroupID", and "MailingGroupID". A third dashed line connects "CustomerGroupID" to "eDocSenderCRID", "FSNonFSAvailability...", "NotificationDate", and "AvailableRecordCou...". Below the main structure, a note states: "Notification from USPS that a full service non compliance data report is ready for pickup."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation Notification from USPS that a full service non compliance data report is ready for pickup.</p>

element **NonComplianceDataWithPostageOwedReportQueryResponse**

diagram	<pre> graph LR ND[NonComplianceDataWithPostageOwedReportQueryResponse] --- Attributes[attributes] Attributes --- LargeTransactions[grp mailxml_defs:LargeTransactions] LargeTransactions --- MGID[mailxml_defs:MessageGroupID] LargeTransactions --- TMC[mailxml_defs:TotalMessageCount] LargeTransactions --- MSN[mailxml_defs:MessageSerialNumber] LargeTransactions --- TRR[mailxml_defs:TransmittedRecords] LargeTransactions --- TRA[mailxml_defs:TotalRecordsAcrossAll] LargeTransactions --- LM[mailxml_defs:LastMessage] ND --- TrackingID[mailxml_dd:TrackingID] ND --- SubmitterTrackingID[mailxml_dd:SubmitterTrackingID] TrackingID --- QueryResults[mailxml_dd:QueryResults] SubmitterTrackingID --- QueryResults QueryResults --- QR[mailxml_defs:QueryResults] QueryResults --- QE[mailxml_defs:QueryError] </pre> <p>The diagram illustrates the structure of the NonComplianceDataWithPostageOwedReportQueryResponse element. It consists of two main parts: a set of attributes and a response body. The attributes include grp mailxml_defs:LargeTransactions, which contains mailxml_defs:MessageGroupID, mailxml_defs:TotalMessageCount, mailxml_defs:MessageSerialNumber, mailxml_defs:TransmittedRecords, mailxml_defs:TotalRecordsAcrossAll, and mailxml_defs:LastMessage. The response body contains mailxml_dd:TrackingID and mailxml_dd:SubmitterTrackingID. The mailxml_dd:SubmitterTrackingID node branches into mailxml_dd:QueryResults and mailxml_defs:QueryError. A note indicates that mailxml_defs:QueryError is issued when query data cannot be provided.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Response to the Request for Full Service Non Compliance Data With Postage Owed Report Query Request</p>

element **ScanSTCCount**

diagram	<p>The diagram illustrates the structure of the ScanSTCCount element. It begins with a lifeline labeled "ScanSTCCount". This lifeline sends a message to another lifeline, which then branches into two parallel regions. The top region contains four objects: UserLicenseCode, MaildataJobID, CustomerGroupID, and MailingGroupID. The bottom region contains two objects: ConsigneeApptID and LogicalIndicator. Both regions converge back onto the original lifeline. From this convergence point, the lifeline sends a message to a third region, which contains five objects: McbCount, IMtbCount, IMpbCount, IMbCount, and STCCount. Finally, the lifeline sends a message to a fourth region, which contains three objects: PlannedCount, PaidCount, and ScannedCount.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	documentation ScanandSPMCountfor thridpartyreconciliation

element **ScanSTCReconciliationDelivery**

diagram	<pre> classDiagram class ScanSTCReconciliationDelivery { grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage } class SubmittingParty class SubmittingSoftware class DataRecipient class eDocSenderCRID class PushMessageID class ScanSTCCount ScanSTCReconciliationDelivery "1..∞" --> SubmittingParty : ScanSTCReconciliationDelivery "1..∞" --> SubmittingSoftware : ScanSTCReconciliationDelivery "1..∞" --> DataRecipient : ScanSTCReconciliationDelivery "1..∞" --> eDocSenderCRID : ScanSTCReconciliationDelivery "1..∞" --> PushMessageID : ScanSTCReconciliationDelivery "1..∞" --> ScanSTCCount : </pre> <p>ScanSTCReconciliationDelivery Response to the Query request for full service container visibility information.</p> <p>attributes</p> <ul style="list-style-type: none"> grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage <p>Relationships</p> <ul style="list-style-type: none"> SubmittingParty SubmittingSoftware DataRecipient eDocSenderCRID PushMessageID ScanSTCCount <p>ScanandSPMCountfor thirdpartyreconciliation</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Response to the Query request for full service container visibility information.</p>

element **ScanSTCReconciliationNotification**

diagram	<pre> graph LR A[ScanSTCReconciliationNotifica...] A --- B[...] B --- C[...] C --- D[...] C --- E[...] C --- F[...] C --- G[...] C --- H[...] C --- I[...] C --- J[...] C --- K[...] C --- L[...] C --- M[...] C --- N[...] C --- O[...] C --- P[...] C --- Q[...] C --- R[...] C --- S[...] C --- T[...] C --- U[...] C --- V[...] C --- W[...] C --- X[...] C --- Y[...] C --- Z[...] </pre> <p>Notification sent by USPS that full service container visibility information is ready for pickup.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Notification sent by USPS that full service container visibility information is ready for pickup.</p>

element **ScanSTCReconciliationQueryRequest**

diagram	<pre> graph LR A[ScanSTCReconciliationQueryR...] A --- B[...] B --- C[...] C --- D[...] C --- E[...] C --- F[...] C --- G[...] C --- H[...] C --- I[...] C --- J[...] C --- K[...] C --- L[...] C --- M[...] C --- N[...] C --- O[...] C --- P[...] C --- Q[...] C --- R[...] C --- S[...] C --- T[...] C --- U[...] C --- V[...] C --- W[...] C --- X[...] C --- Y[...] C --- Z[...] </pre> <p>Query request for full service container visibility information.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Query request for full service container visibility information.</p>

element **ScanSTCReconciliationQueryResponse**

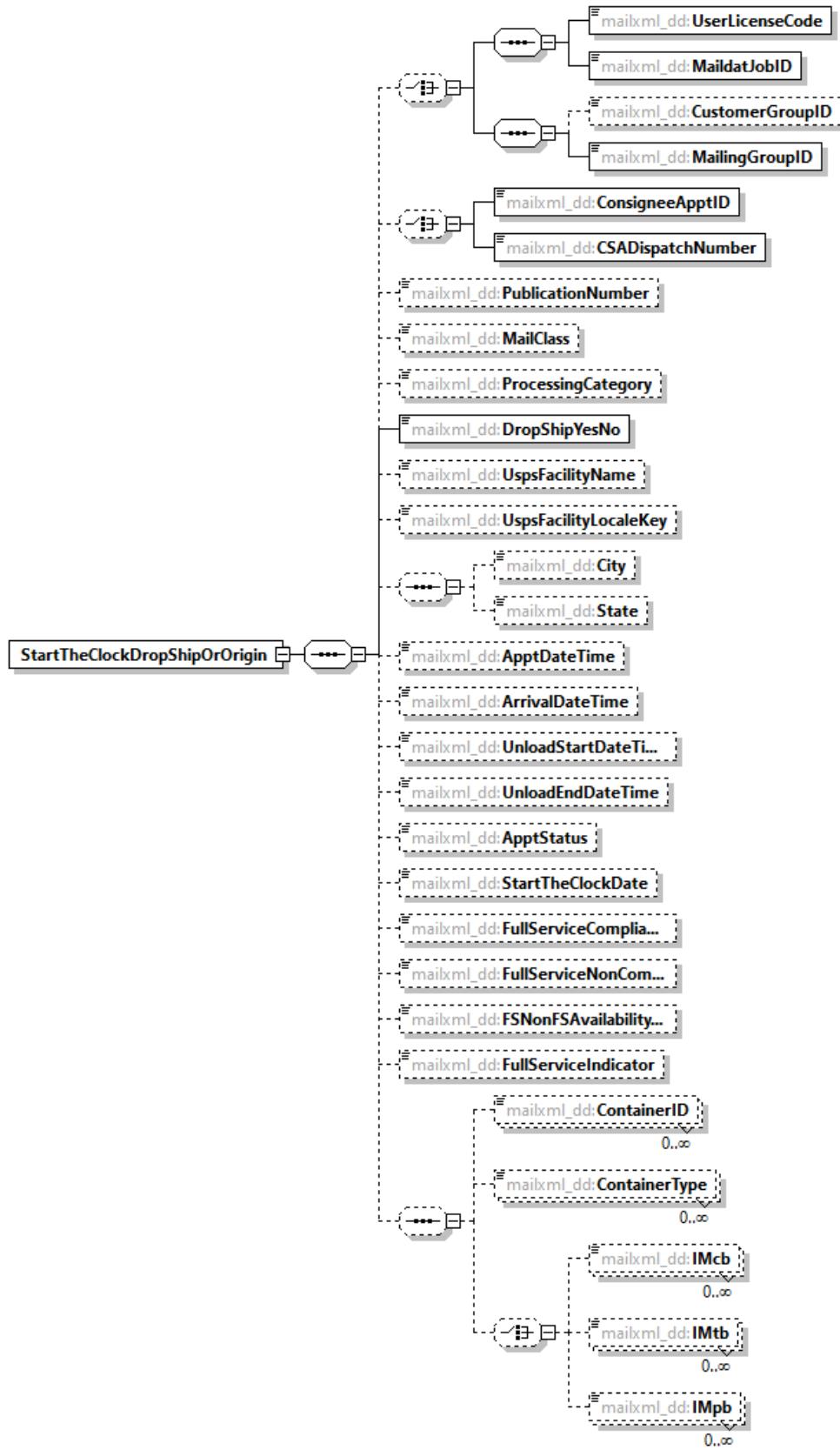
diagram	<pre> classDiagram class ScanSTCReconciliationQueryR { <<Response to the Query request for for full service container visibility information.>> } class mailxml_defs { <<grp mailxml_defs:LargeTransa...>> MessageGroupID TotalMessageCount MessageSerialNu... TransmittedRecor... TotalRecordsAcro... LastMessage } class mailxml_dd { TrackingID SubmitterTrackingID QueryResults QueryError } ScanSTCReconciliationQueryR "1" -- "1" mailxml_defs : ScanSTCReconciliationQueryR "1" -- "1" mailxml_dd : mailxml_defs "1..>" -- "1..>" mailxml_dd : mailxml_dd "1..>" -- "1..>" mailxml_defs : </pre> <p>ScanSTCReconciliationQueryR... Response to the Query request for for full service container visibility information.</p> <p>attributes</p> <ul style="list-style-type: none"> grp mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage <p>mailxml_dd:</p> <ul style="list-style-type: none"> TrackingID SubmitterTrackingID QueryResults QueryError <p>Error issued when the query data cannot be provided.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
annotation	<p>documentation</p> <p>Response to the Query request for for full service container visibility information.</p>

element **StartTheClockBEMUBlock**

diagram	<pre> sequenceDiagram participant A as StartTheClockBEMUBlock participant B as mailxml_dd:StartTheClockBEMU A->>B: activate B B-->>A: deactivate B 1..>A </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

element **StartTheClockDropShipOrOrigin**

diagram



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

element **StartTheClockPlantLoadBlock**

diagram	<pre> graph LR Start[StartTheClockPlantLoadBlock] --- Seq1[...] Seq1 --- Plan[mailxml_dd:StartTheClockPlan...] Plan -- "1..∞" --> Plan </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

complexType **addressCorrectionAddressType**

diagram	<pre> graph TD Root[addressCorrectionAddressType] --- Seq1[...] Seq1 --- Type[mailxml_dd:AddressType] Type --- Urban[mailxml_dd:UrbanizationName] Urban --- Num[mailxml_dd:PrimaryNumber] Num --- Dir1[mailxml_dd:PreDirectional] Dir1 --- StName[mailxml_dd:StreetName] StName --- Suffix[mailxml_dd:StreetSuffix] Suffix --- Dir2[mailxml_dd:PostDirectional] Dir2 --- Unit[mailxml_dd:UnitDesignator] Unit --- SecNum[mailxml_dd:SecondaryNumber] SecNum --- City[mailxml_dd:City] City --- State[mailxml_dd:State] State --- Zip[mailxml_dd:ZipCode] Zip --- Plus4[mailxml_dd:ZipCodePlus4] Plus4 --- D11[mailxml_dd:DeliveryPoint11Dig...] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

complexType **byForConflictType**

diagram	<p>The diagram illustrates the structure of the byForConflictType complex type. It starts with a base class byForConflictType at the top left. An association line connects it to a dashed box containing several XML elements: mailxml_dd:UserLicenseCode, mailxml_dd:MaildatJobID, mailxml_dd:CustomerGroupID, mailxml_dd:MailingGroupID, mailxml_dd:MailOwnerConflict (with multiplicity 0..∞), mailxml_dd:MailPreparerConflict (with multiplicity 0..∞), mailxml_dd:FSNonFSAvailability..., mailxml_dd:FullServiceIndicator, and mailxml_dd:ReturnInfo.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

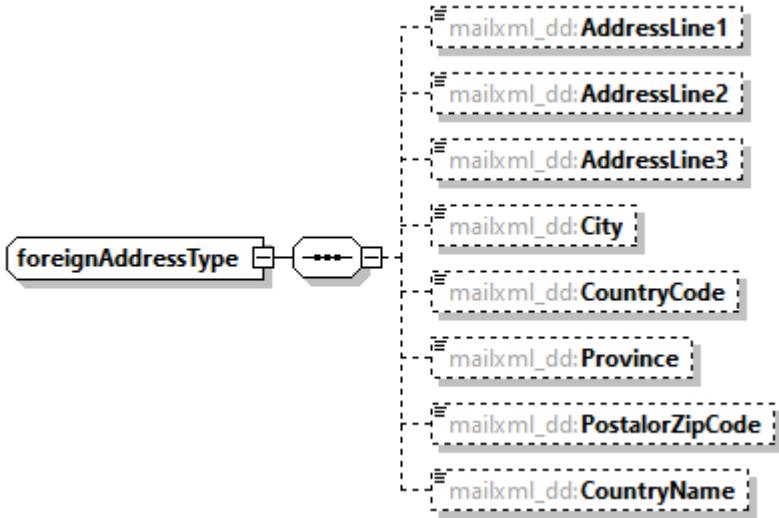
complexType **clockStartedType**

diagram	<p>The diagram illustrates the structure of the clockStartedType complex type. It starts with a base class clockStartedType at the top left. An association line connects it to a dashed box containing three XML elements: mailxml_dd:StartTheClockBEM... (with multiplicity 0..∞), mailxml_dd:StartTheClockDro... (with multiplicity 0..∞), and mailxml_dd:StartTheClockPlan... (with multiplicity 0..∞).</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

complexType **dqrContainerInfoType**

diagram	<pre> classDiagram class dqrContainerInfoType { mailxml_dd:ContainerID mailxml_dd:SiblingRefContainer... mailxml_dd:ParentContainerID mailxml_dd:ProcessingCategory mailxml_dd:MailClass mailxml_dd:USPSFacilityK... mailxml_dd:City mailxml_dd:State mailxml_dd:MailingDate mailxml_dd:ContainerType mailxml_dd:IMcb mailxml_dd:IMtb mailxml_dd:CSAID mailxml_dd:ConsigneeAppTID mailxml_dd:FSNonFSAvailability... mailxml_dd:FullServiceIndicator mailxml_dd:VerificationError mailxml_dd:PreparerCRID mailxml_dd:OwnerCRID mailxml_dd:PieceInfo mailxml_dd:PieceRangeInfo } </pre> <p>The diagram illustrates the structure of the dqrContainerInfoType complex type. It consists of the following components:</p> <ul style="list-style-type: none"> Attributes: <ul style="list-style-type: none"> <code>mailxml_dd:ContainerID</code> <code>mailxml_dd:SiblingRefContainer...</code> <code>mailxml_dd:ParentContainerID</code> <code>mailxml_dd:ProcessingCategory</code> <code>mailxml_dd:MailClass</code> <code>mailxml_dd:USPSFacilityK...</code> <code>mailxml_dd:City</code> <code>mailxml_dd:State</code> <code>mailxml_dd:MailingDate</code> <code>mailxml_dd:ContainerType</code> <code>mailxml_dd:IMcb</code> <code>mailxml_dd:IMtb</code> <code>mailxml_dd:CSAID</code> <code>mailxml_dd:ConsigneeAppTID</code> <code>mailxml_dd:FSNonFSAvailability...</code> <code>mailxml_dd:FullServiceIndicator</code> <code>mailxml_dd:VerificationError</code> Associations: <ul style="list-style-type: none"> <code>mailxml_dd:VerificationError</code> has a multiplicity of <code>0..∞</code>. <code>mailxml_dd:PreparerCRID</code> has a multiplicity of <code>0..∞</code>. <code>mailxml_dd:OwnerCRID</code> has a multiplicity of <code>0..∞</code>. <code>mailxml_dd:PieceInfo</code> has a multiplicity of <code>0..∞</code>. <code>mailxml_dd:PieceRangeInfo</code> has a multiplicity of <code>0..∞</code>.
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

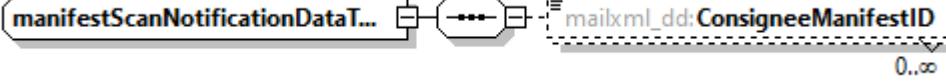
complexType **foreignAddressType**

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

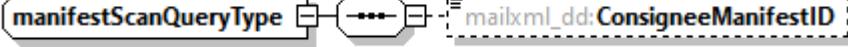
complexType **manifestScanEventDetailType**

diagram	<pre> sequenceDiagram manifestScanEventDetailType --> mailxml_dd:USPSEventExtractFi... manifestScanEventDetailType --> mailxml_dd:IMpb manifestScanEventDetailType --> mailxml_dd:ElectronicFileNumber manifestScanEventDetailType --> mailxml_dd:MailerID manifestScanEventDetailType --> mailxml_dd:MailerName manifestScanEventDetailType --> mailxml_dd:DestinationZipCode manifestScanEventDetailType --> mailxml_dd:DestinationZipPlusF... manifestScanEventDetailType --> mailxml_dd:ScanningFacilityZip manifestScanEventDetailType --> mailxml_dd:ScanningFacilityNa... manifestScanEventDetailType --> mailxml_dd:EventCode manifestScanEventDetailType --> mailxml_dd:EventName manifestScanEventDetailType --> mailxml_dd:EventDate manifestScanEventDetailType --> mailxml_dd:EventTime manifestScanEventDetailType --> mailxml_dd:MailerOwnerID manifestScanEventDetailType --> mailxml_dd:CustomerReferenc... manifestScanEventDetailType --> mailxml_dd:DestinationCountry... manifestScanEventDetailType --> mailxml_dd:RecipientName manifestScanEventDetailType --> mailxml_dd:OriginalLabel manifestScanEventDetailType --> mailxml_dd:UnitofMeasureCode manifestScanEventDetailType --> mailxml_dd:Weight manifestScanEventDetailType --> mailxml_dd:GuaranteedDeliver... manifestScanEventDetailType --> mailxml_dd:GuaranteedDeliver... manifestScanEventDetailType --> mailxml_dd:LogisticsManager... </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

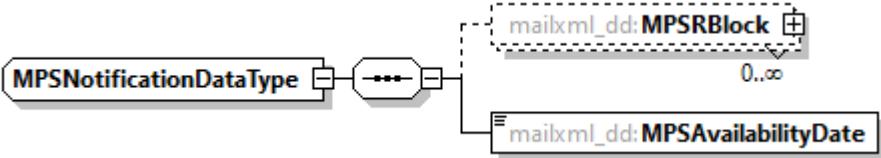
complexType **manifestScanNotificationDataType**

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

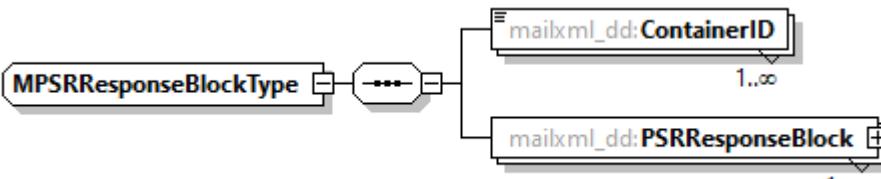
complexType **manifestScanQueryType**

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

complexType **MPSNotificationDataType**

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

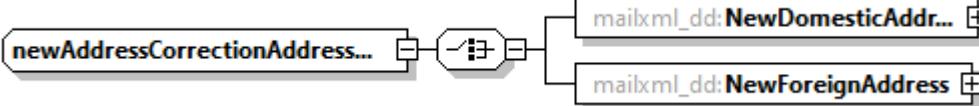
complexType **MPSRResponseBlockType**

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

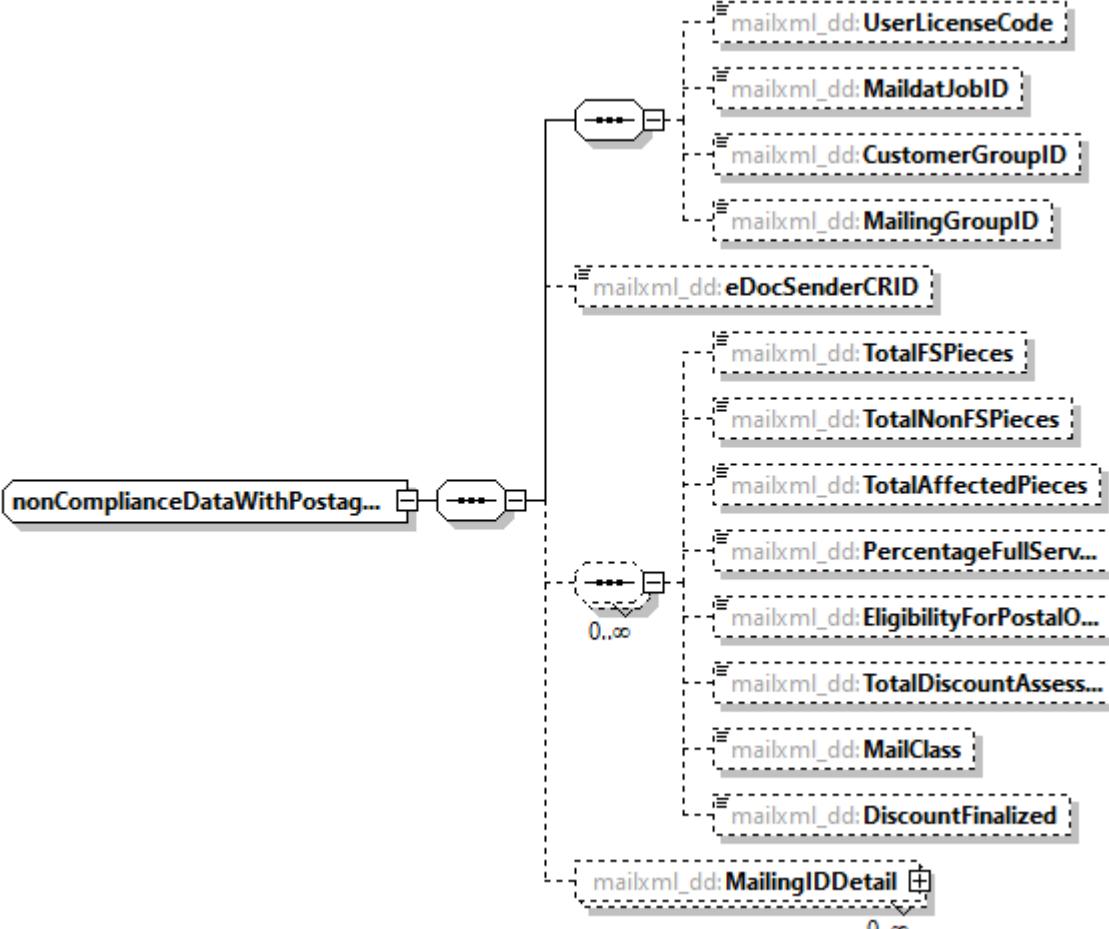
complexType **MPSVisScanQueryType**

diagram	<pre> classDiagram class MPSVisScanQueryType { <<mailxml_dd:ScanQueryType>> mailxml_dd:UserLicenseCode mailxml_dd:MaildatJobID mailxml_dd:CustomerGroupID mailxml_dd:MailingGroupID mailxml_dd:ResultOptions* mailxml_dd:RangeLowerDate mailxml_dd:RangeUpperDate mailxml_dd:MID6OnThePiece mailxml_dd:MID9OnThePiece mailxml_dd:FacilityLocaleKey mailxml_dd:EventType* mailxml_dd:BundleScanType* mailxml_dd:IMcb mailxml_dd:IMtb mailxml_dd:IMB mailxml_dd:BundleID mailxml_dd:FieldIncludedScan... } MPSVisScanQueryType < -- mailxml_dd:ScanQueryType </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

complexType **newAddressCorrectionAddressType**

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

complexType **nonComplianceDataWithPostageOwedReportType**

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

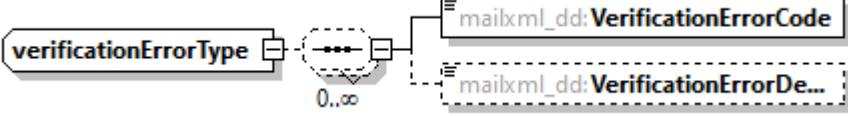
complexType **PSRResponseBlockType**

diagram	<p>The diagram illustrates the structure of the PSRResponseBlockType complex type. It starts with a central node labeled "PSRResponseBlockType". From this node, three dashed lines branch out to three separate nodes: "mailxml_dd:PRID", "mailxml_dd:IMBLS", and "mailxml_dd:IMBUS". From the "mailxml_dd:IMBLS" and "mailxml_dd:IMBUS" nodes, solid lines lead to a single node labeled "mailxml_dd:ServiceTypeCode". From this "ServiceTypeCode" node, two more solid lines lead to "mailxml_dd:MailClass" and "mailxml_dd:ServiceLevelIndicat...". Finally, a line from "mailxml_dd:ServiceLevelIndicat..." leads to "mailxml_dd:MailpieceScanCount".</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

complexType **unManifestedScanEventDetailType**

diagram	<pre> graph LR A[unManifestedScanEventDetailT...] --- B{...} B --- C[mailxml_dd: USPSEventExtractFi...] B --- D[mailxml_dd: IMpb] B --- E[mailxml_dd: ElectronicFileNumber] B --- F[mailxml_dd: MailerID] B --- G[mailxml_dd: MailerName] B --- H[mailxml_dd: DestinationZipCode] B --- I[mailxml_dd: DestinationZipPlusF...] B --- J[mailxml_dd: ScanningFacilityZip] B --- K[mailxml_dd: ScanningFacilityNa...] B --- L[mailxml_dd: EventCode] B --- M[mailxml_dd: EventName] B --- N[mailxml_dd: EventDate] B --- O[mailxml_dd: EventTime] B --- P[mailxml_dd: MailerOwnerID] B --- Q[mailxml_dd: CustomerReferenc...] B --- R[mailxml_dd: DestinationCountry...] B --- S[mailxml_dd: RecipientName] B --- T[mailxml_dd: OriginalLabel] B --- U[mailxml_dd: UnitofMeasureCode] B --- V[mailxml_dd: Weight] B --- W[mailxml_dd: GuaranteedDeliver...] B --- X[mailxml_dd: GuaranteedDeliver...] B --- Y[mailxml_dd: LogisticsManager...] </pre> <p>The diagram illustrates the structure of the unManifestedScanEventDetailType complex type. It begins with a root element, unManifestedScanEventDetailT..., which has a sequence relationship (indicated by a sequence symbol) with a central node. This central node is connected to various child elements, each preceded by the prefix mailxml_dd:. The child elements are: USPSEventExtractFi..., IMpb, ElectronicFileNumber, MailerID, MailerName, DestinationZipCode, DestinationZipPlusF..., ScanningFacilityZip, ScanningFacilityNa..., EventCode, EventName, EventDate, EventTime, MailerOwnerID, CustomerReferenc..., DestinationCountry..., RecipientName, OriginalLabel, UnitofMeasureCode, Weight, GuaranteedDeliver..., GuaranteedDeliver..., and LogisticsManager.... Some elements, such as MailerID, DestinationZipPlusF..., CustomerReferenc..., OriginalLabel, UnitofMeasureCode, GuaranteedDeliver..., and LogisticsManager..., are enclosed in dashed boxes, suggesting they are optional or have specific constraints.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd

complexType verificationErrorType

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

simpleType addressCorrectionMoveType

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

simpleType addressTypeType

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

simpleType containerScanStateType

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

simpleType deliverabilityCodeType

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

simpleType eDocTypeType

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

simpleType primarySecondaryIndicatorType

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

simpleType **verificationErrorTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

simpleType **verificationWarningTypeType**

namespace	http://delivery-tech.org/Specs/mailxml26.3/mailxml_dd
type	restriction of xs:string

XML Schema documentation generated by **XMLSpy** Schema Editor <http://www.altova.com/xmlspy>