

Mail.XML Version 26.2

System Messages Specification

Monday, September 11, 2023

Working Group Chair

Shawn Baldwin, BCC Software

Technical Director

Shariq Mirza, DTAC Associate, Assurety Consulting & Solutions

Editor

Shariq Mirza, DTAC Associate, Assurety Consulting & Solutions

Copyright (c) 2023 – 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

© 2023 Delivery Technology Advocacy Council. All Rights Reserved.

Copyright 2023 – 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 26.2?

With this release, the Mail.XML Messaging Protocol moves to Version 26.2. This release supports structure changes required by mailing industry and Postal Service.

Changes supported by Mail.XML 26.2 include:

- CR 2615 - Updating characteristicIncentiveType and adding FG for First-Class Growth Incentive Credit Redemption
- CR 2616 - Updating characteristicIncentiveType and adding MG for USPS Marketing Mail Growth Incentive Credit Redemption
- CR 2617 - Updating containerLevelType, and adding AU for Protected Mixed ADC and AV for Protected Mixed NDC

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/mailxml26.2/mailxml"
targetNamespace="http://delivery-tech.org/Specs/mailxml26.2/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/mailxml26.2A/mailxml" then at least one of the XSDs is at same version such as filename ='Mail.XML_26.2A.xsd' <- Errata A
- If the wrapper version is labeled as xmlns:mailxml="http://deliverytech.org/Specs/mailxml26.2B/mailxml" then at least one of the XSDs is at same version such as filename ='Mail.XML_26.2B.xsd' <- Errata B

- If the wrapper version is labeled as xmlns:mailxml="http://deliverytech.org/Specs/mailxml26.2/mailxml" then at least one of the XSDs is at same version such as filename ='Mail.XML_26.2.xsd' <- Major Version

Mail.XML 26.2 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™ 26.2 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 26.2. Updates in this Specification are NOT backwardly compatible with previous versions. Other documents in the specification set include:

- Mail.XML™ 26.2: Transportation Messaging Specification documents all transportation messages
- Mail.XML™ 26.2: Mailing Messaging Specification documents all mailing messages
- Mail.XML™ 26.2: Informed Visibility Specification documents all informed visibility messages
- Mail.XML™ 26.2: Data Distribution Messaging Specification documents all data distribution messages
- Mail.XML™ 26.2: Identification Messaging Specification documents all identification messages
- Mail.XML™ 26.2: Supply Chain Messaging Specification documents all supply chain messages
- Mail.XML™ 26.2: System Messaging Specification documents all systems and fault messages
- Mail.XML™ 26.2: Simple Types Specification documents all simple types used across Mail.XML messages

- Mail.XML™ 26.2: Common Definitions Specification documents all shared elements and complex types.
- Mail.XML™ 26.2: 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 26.2?	3
About Mail.XML Schema Modularization	4
Mail.XML Module Versioning Rules	4
Mail.XML 26.2 XSD Modules	5
The Mail.XML™ 26.2 Messaging Documentation Set	5
Schema mailxml_dd_26.2.xsd	8

Schema mailxml_dd_26.2.xsd

schema location: [C:\Users\NabilRahman\Desktop\XML SCHEMA UPDATE\26.2\Mail.XML
26.2\MailXML_26.2\XSDs\mailxml_dd_26.2.xsd](C:\Users\NabilRahman\Desktop\XML SCHEMA UPDATE\26.2\Mail.XML\26.2\MailXML_26.2\XSDs\mailxml_dd_26.2.xsd)
attributeFormDefault: **qualified**
elementFormDefault: **qualified**
targetNamespace: http://delivery-tech.org/Specs/mailxml26.2/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](#)
[NonComplianceDataWithPostageOwedReport](#)

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](#)
[verificationWarningType](#)
[eType](#)

[QueryResponse](#)

[ScanSTCCount](#)

[ScanSTCReconciliationDelivery](#)

[ScanSTCReconciliationNotification](#)

[ScanSTCReconciliationQueryRequest](#)

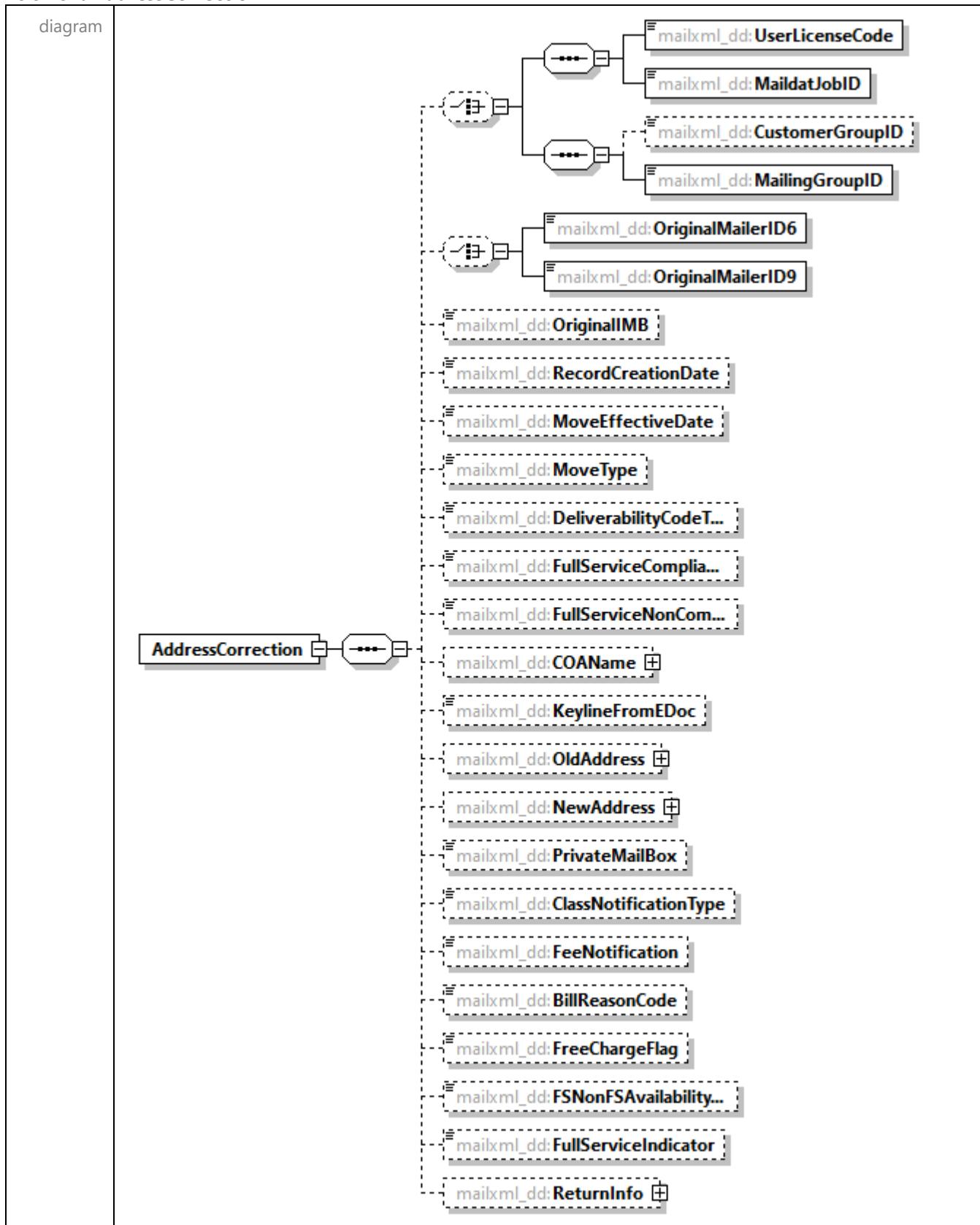
[ScanSTCReconciliationQueryResponse](#)

[StartTheClockBEMUBlock](#)

[StartTheClockDropShipOrOrigin](#)

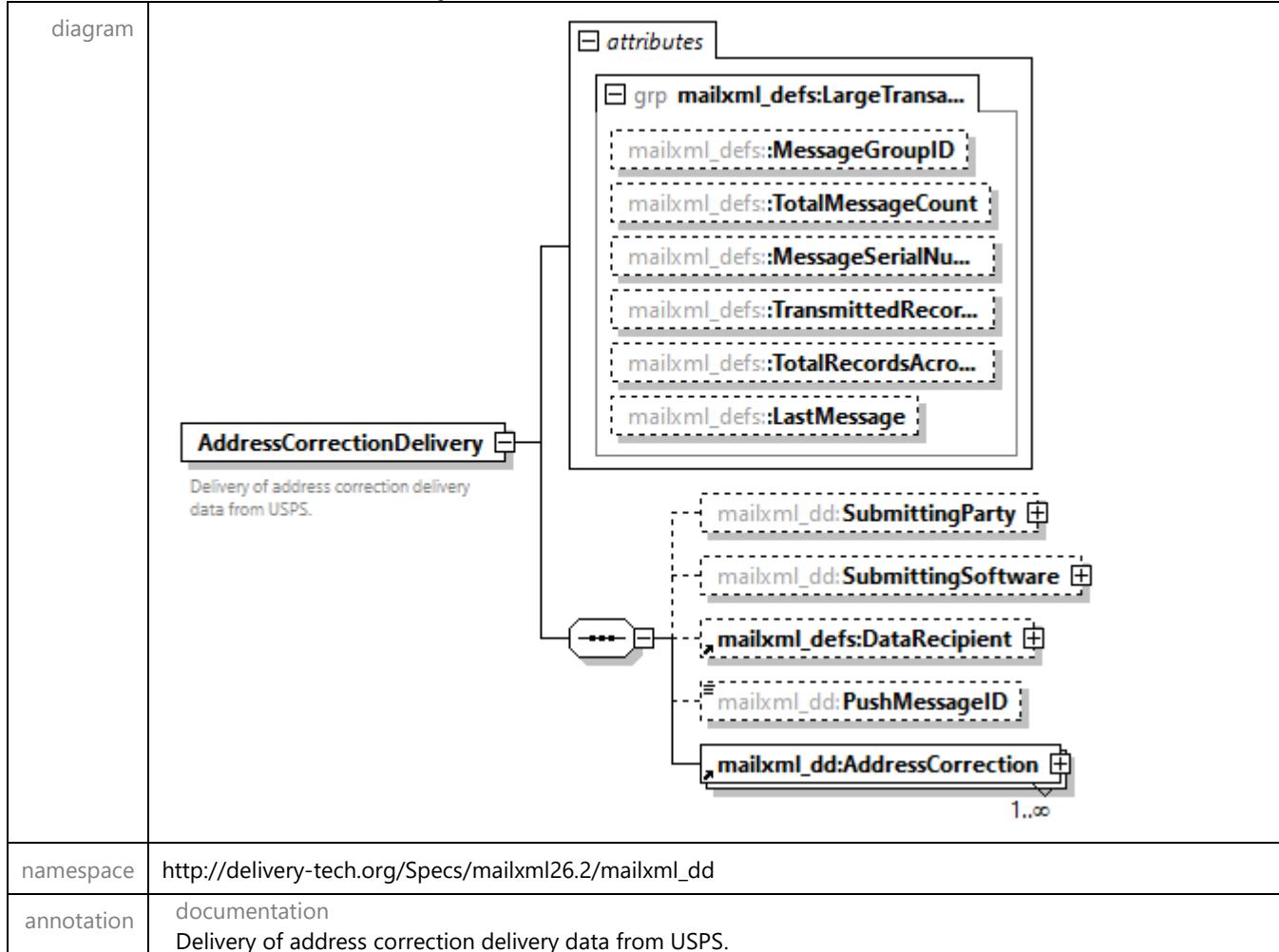
[StartTheClockPlantLoadBlock](#)

element **AddressCorrection**



namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
-----------	---

element **AddressCorrectionDelivery**



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 "..." --> SubmittingParty AddressCorrectionNotification "..." --> SubmittingSoftware AddressCorrectionNotification "..." --> CRID AddressCorrectionNotification "..." --> PushMessageID PushMessageID "*" --> UserLicenseCode PushMessageID "*" --> MaildatJobID PushMessageID "*" --> CustomerGroupID PushMessageID "*" --> MailingGroupID PushMessageID "*" --> FSNonFSAvailability... PushMessageID "*" --> NotificationDate PushMessageID "*" --> AvailableRecordCou... </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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... "..." --> SubmittingParty AddressCorrectionQueryReq... "..." --> SubmittingSoftware AddressCorrectionQueryReq... "..." --> SubmitterTrackingID AddressCorrectionQueryReq... "..." --> OwningParty PieceRequest "*" --> retrieveDataBy "1..∞" </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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 relationship line connects this class to a larger box labeled attributes. Inside the attributes box, there is a group header grp mailxml_defs:LargeTransa... followed by several attributes: mailxml_defs:MessageGroupID, mailxml_defs:TotalMessageCount, mailxml_defs:MessageSerialNu..., mailxml_defs:TransmittedRecor..., mailxml_defs:TotalRecordsAcro..., and mailxml_defs:LastMessage. Below these, there is a dashed-line box containing mailxml_dd:TrackingID and mailxml_dd:SubmitterTrackingID. Another dashed-line box contains mailxml_dd:QueryResults and mailxml_defs:QueryError. The QueryError box includes a note: "Error issued when the query data cannot be provided."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation Response to the query request for full service address correction data.</p>

element **ByForConflictDelivery**

diagram	<pre> classDiagram class ByForConflictDelivery { 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 ByForConflict ByForConflictDelivery "1..∞" --> SubmittingParty ByForConflictDelivery "1..∞" --> SubmittingSoftware ByForConflictDelivery "1..∞" --> DataRecipient ByForConflictDelivery "1..∞" --> PushMessageID ByForConflictDelivery "1..∞" --> ByForConflict </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation Delivery of by / for conflict information from USPS.</p>

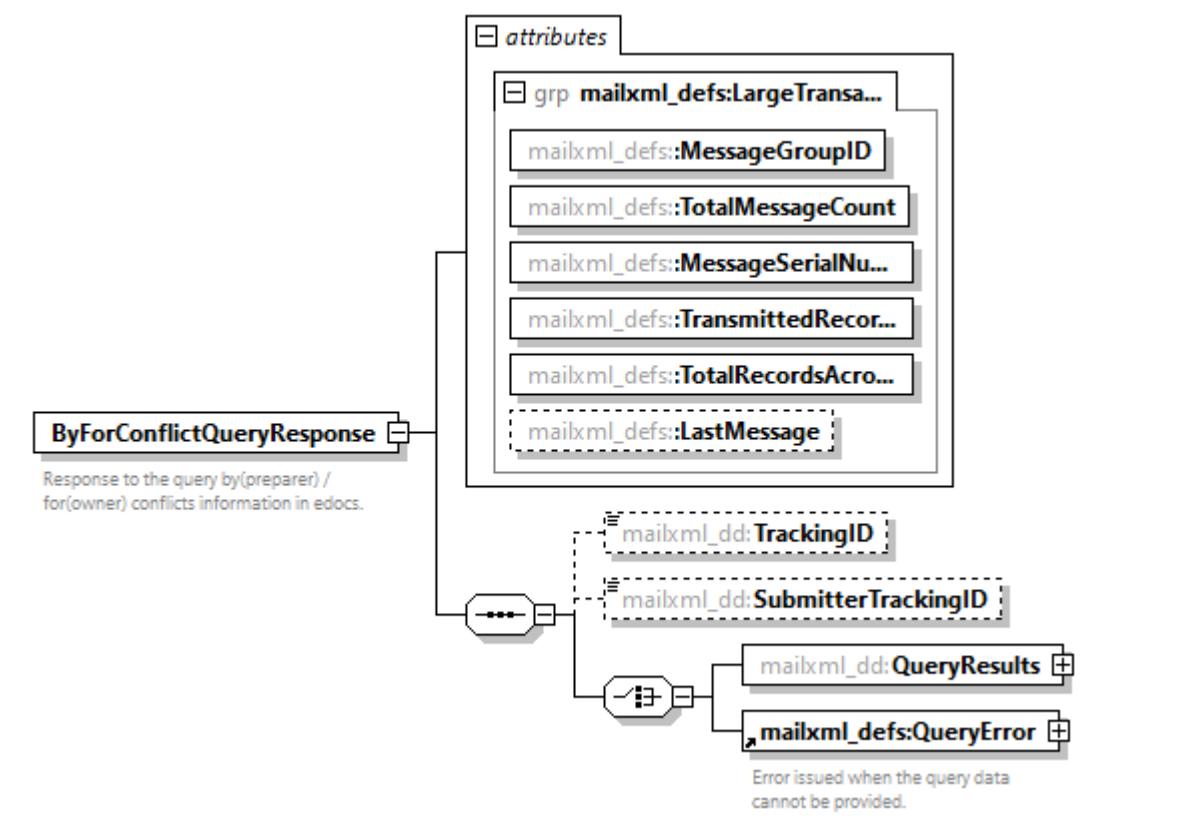
element **ByForConflictNotification**

diagram	<pre> graph LR A[ByForConflictNotification] --> B(()) B --> C1(mailxml_dd:SubmittingParty) B --> C2(mailxml_dd:SubmittingSoftware) B --> C3(mailxml_dd:PushMessageID) C1 --- D1(mailxml_dd:UserLicenseCode) C2 --- D2(mailxml_dd:MaildatJobID) C3 --- D3(mailxml_dd:CustomerGroupID) C3 --- D4(mailxml_dd:MailingGroupID) C3 --- D5(mailxml_dd:FSNonFSAvailability...) C3 --- D6(mailxml_dd:NotificationDate) C3 --- D7(mailxml_dd:AvailableRecordCou...) </pre> <p>Notification from USPS that a by / for conflict resolution is ready for pickup.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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> graph LR A[ByForConflictQueryRequest] --> B(()) B --> C1(mailxml_dd:SubmittingParty) B --> C2(mailxml_dd:SubmittingSoftware) B --> C3(mailxml_dd:SubmitterTrackingID) C1 --- D1(mailxml_dd:UserLicenseCode) C2 --- D2(mailxml_dd:MaildatJobID) C3 --- D3(mailxml_dd:CustomerGroupID) C3 --- D4(mailxml_dd:MailingGroupID) C3 --- D5(mailxml_dd:retrieveDataBy) </pre> <p>Query request for by(preparer) / for(owner) conflicts information in edocs.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation</p> <p>Query request for by(preparer) / for(owner) conflicts information in edocs.</p>

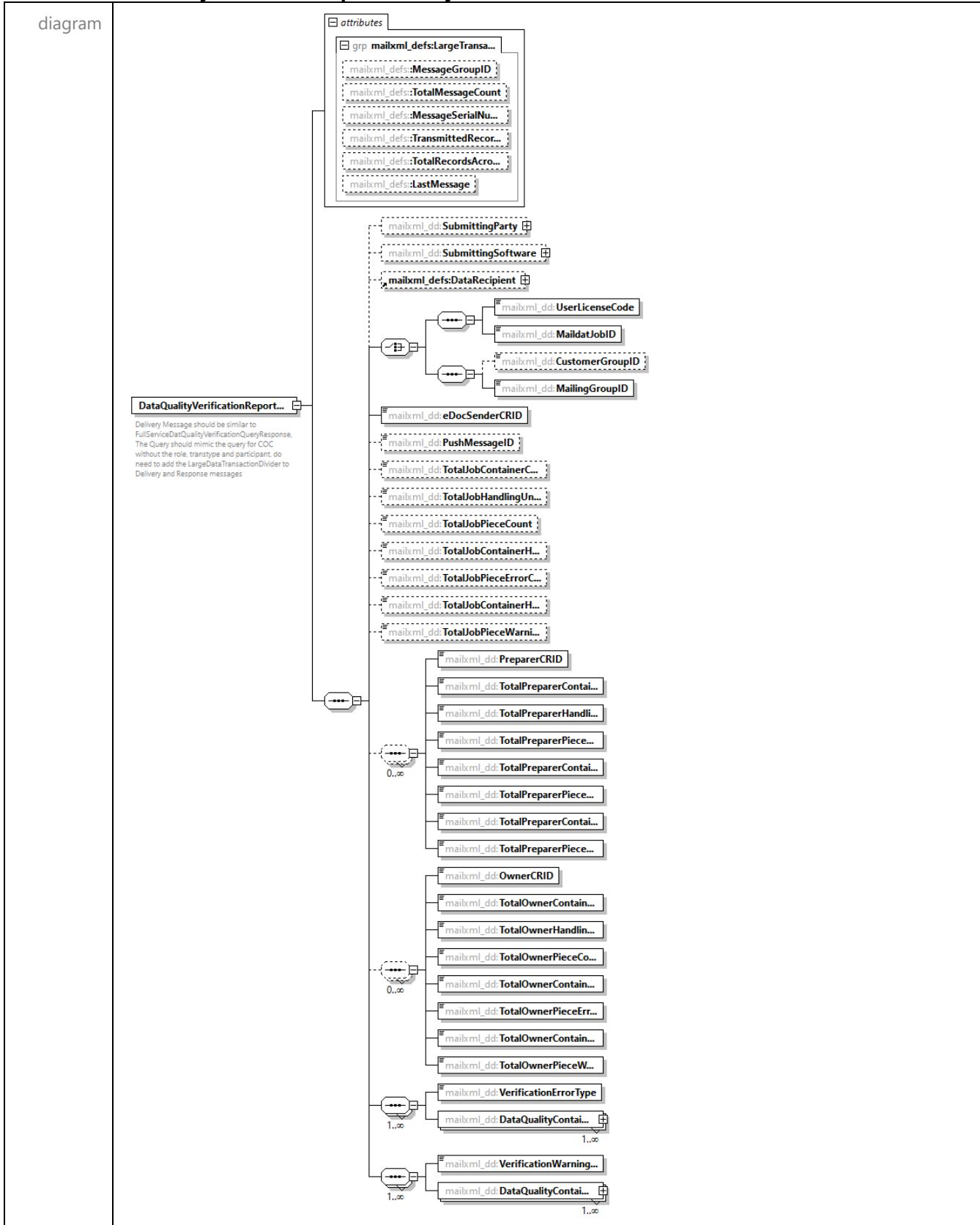
element **ByForConflictQueryResponse**

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" box is another line connecting to a dashed-line box labeled mailxml_dd:TrackingID. From TrackingID, a line leads to a dashed-line box labeled mailxml_dd:SubmitterTrackingID. From SubmitterTrackingID, two lines lead to separate boxes: mailxml_dd:QueryResults and mailxml_defs:QueryError. A note at the bottom right states: "Error issued when the query data cannot be provided."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation 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 connected by dashed lines:</p> <ul style="list-style-type: none">A central ContainerVisibilityEntry object.Associations from the central object to:<ul style="list-style-type: none">mailxml_dd:UserLicenseCodemailxml_dd:MaildataJobIDmailxml_dd:CustomerGroupIDmailxml_dd:MailingGroupIDmailxml_dd:ConsigneeApptIDmailxml_dd:LogicalIndicatormailxml_dd:CSAIDmailxml_dd:ContainerIDmailxml_dd:ParentContainerIDmailxml_dd:SiblingContainerIDmailxml_dd:ContainerTypemailxml_dd:ScanEvent (with a plus sign)mailxml_dd:IMcbmailxml_dd:IMtbmailxml_dd:IMpbmailxml_dd:FullServiceComplia...mailxml_dd:FullServiceNonCom...mailxml_dd:FSNonFSAvailability...mailxml_dd:FullServiceIndicator
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

element DataQualityVerificationReportDelivery



namespace	http://delivery-tech.org/Specs/mailxml26.2/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 { SubmittingParty SubmittingSoftware PushMessageID UserLicenseCode MaildatJobID CustomerGroupID MailingGroupID eDocSenderCRID VerificationErrorType VerificationWarning... FSNonFSAvailability... NotificationDate AvailableRecordCou... } </pre> <p>DataQualityVerificationReport...</p> <p>Notification from USPS that a full service data quality verification report is ready for pickup.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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>Annotations below the main class state: "Query request for full service data quality reports."</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	documentation Query request for full service data quality reports.

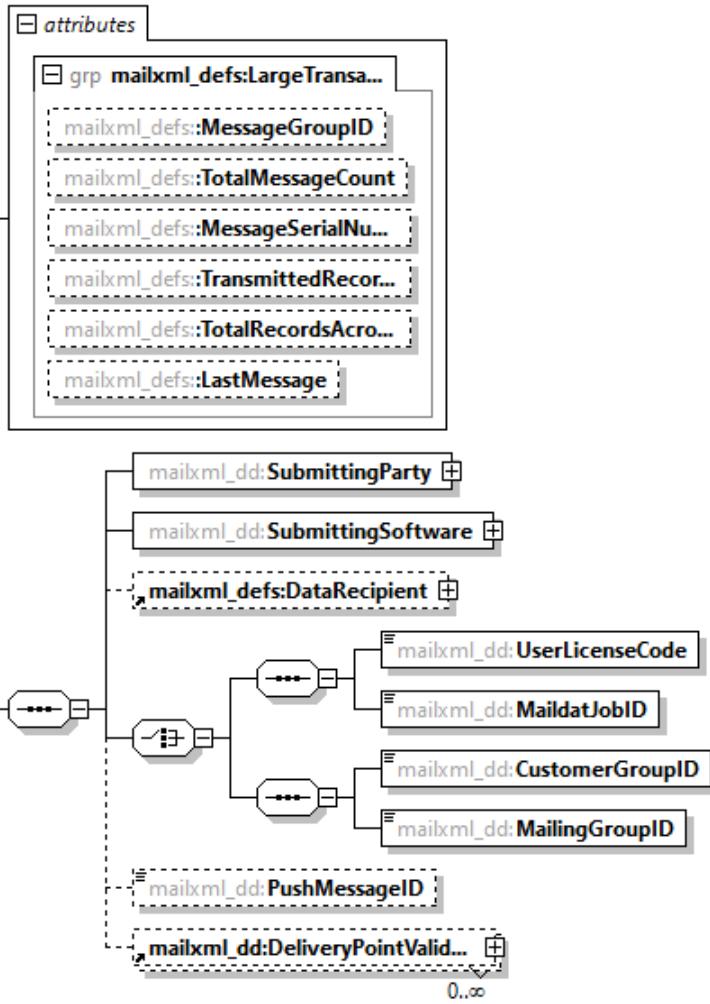
element **DataQualityVerificationReportQueryResponse**

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 note over mailxml_dd:QueryResults, mailxml_defs:QueryError: "Error issued when the query data cannot be provided." </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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; DV[DeliveryPointValidation] --> O1[OriginalMailerID6]; DV --> O2[OriginalMailerID9]; DV --> ULC[UserLicenseCode]; O1 --- MaildatJobID1[MaildatJobID]; O1 --- CGID1[CustomerGroupID]; O1 --- MGID1[MailingGroupID]; O2 --- MaildatJobID2[MaildatJobID]; O2 --- CGID2[CustomerGroupID]; O2 --- MGID2[MailingGroupID]; DV --> IDP[InvalidDeliveryPoint...]; DV --> PSI[PrimarySecondaryl...]; DV --> DPV[DPVDataRecipientC...]; DV --> FSC[FullServiceComplia...]; DV --> FSNC[FullServiceNonCom...]; DV --> FSI[FullServiceIndicator]; DV --> FNFSA[FSNonFSAvailability...]; DV --> RI[ReturnInfo]</pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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..1" SubmittingParty DeliveryPointValidationDelivery "1" -- "0..1" SubmittingSoftware DeliveryPointValidationDelivery "1" -- "0..1" DataRecipient DataRecipient "*" -- "0..1" UserLicenseCode DataRecipient "*" -- "0..1" MaildatJobID DataRecipient "*" -- "0..1" CustomerGroupID DataRecipient "*" -- "0..1" MailingGroupID DataRecipient "*" -- "0..1" PushMessageID DataRecipient "*" -- "0..1" DeliveryPointValid... </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation</p> <p>Delivery of Delivery Point Validation by USPS.</p>

element **DeliveryPointValidationNotification**

diagram	<pre> graph LR NP[DeliveryPointValidationNotific...] NP --- SP[SubmittingParty] NP --- SS[SubmittingSoftware] NP --- CRID[CRID] NP --- PMID[PushMessageID] NP --- UL[UserLicenseCode] NP --- MJID[MaildatJobID] NP --- CGID[CustomerGroupID] NP --- MGID[MailingGroupID] NP --- FNS[FSNonFSAvailability...] NP --- ND[NotificationDate] NP --- AR[AvailableRecordCou...] CRID --- JID[...] PMID --- UL PMID --- MJID CGID --- MGID </pre> <p>DeliveryPointValidationNotification Notification by USPS that Delivery Point Validation is ready for pickup.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation Notification by USPS that Delivery Point Validation is ready for pickup.</p>

element **DeliveryPointValidationQueryRequest**

diagram	<pre> graph LR NP[DeliveryPointValidationQueryR...] NP --- SP[SubmittingParty] NP --- SS[SubmittingSoftware] NP --- STID[SubmitterTrackingID] NP --- RCRID[RequestorCRID] NP --- OMID6[OriginalMailerID6] NP --- OMID9[OriginalMailerID9] NP --- UL[UserLicenseCode] NP --- MJID[MaildatJobID] NP --- CGID[CustomerGroupID] NP --- MGID[MailingGroupID] NP --- DR[DateRange] NP --- RD[retrieveDataBy] STID --- OMID6 STID --- OMID9 </pre> <p>DeliveryPointValidationQueryRequest Query request for USPS for delivery point validation.</p>
---------	--

namespace	http://delivery-tech.org/Specs/mailxml26.2/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" -- "1" mailxml_dd : mailxml_defs "1" -- "1" mailxml_dd : mailxml_defs "1" -- "1" mailxml_dd : mailxml_dd "*" -- "1" mailxml_defs : mailxml_dd "*" -- "1" mailxml_defs : </pre> <p>The diagram illustrates the structure of the DeliveryPointValidationQueryResponse element. It consists of several components:</p> <ul style="list-style-type: none"> DeliveryPointValidationQueryR: A class representing the response, with a note: "Response to the Query request for USPS for delivery point validation." grp: A group containing the following attributes: <ul style="list-style-type: none"> mailxml_defs:LargeTransa... mailxml_defs:MessageGroupID mailxml_defs:TotalMessageCount mailxml_defs:MessageSerialNu... mailxml_defs:TransmittedRecor... mailxml_defs:TotalRecordsAcro... mailxml_defs:LastMessage mailxml_defs: A group containing attributes: <ul style="list-style-type: none"> MessageGroupID TotalMessageCount MessageSerialNu... TransmittedRecor... TotalRecordsAcro... LastMessage mailxml_dd: A group containing attributes: <ul style="list-style-type: none"> TrackingID SubmitterTrackingID QueryResults QueryError Associations: <ul style="list-style-type: none"> A single association between DeliveryPointValidationQueryR and grp. A single association between DeliveryPointValidationQueryR and mailxml_defs. A single association between DeliveryPointValidationQueryR and mailxml_dd. A bidirectional association between grp and mailxml_defs. A bidirectional association between mailxml_defs and mailxml_dd. A multiplicity of "*" on the mailxml_dd side of the association between DeliveryPointValidationQueryR and mailxml_dd, indicating many-to-one. A multiplicity of "*" on the mailxml_defs side of the association between DeliveryPointValidationQueryR and mailxml_dd, indicating many-to-one. Annotations: <ul style="list-style-type: none"> An annotation for mailxml_dd:QueryError: "Error issued when the query data cannot be provided."
namespace	http://delivery-tech.org/Specs/mailxml26.2/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 base class IMbMailpieceScanData, which has three associations:</p> <ul style="list-style-type: none">An association with a dashed box containing four attributes:<ul style="list-style-type: none">mailxml_dd:UserLicenseCodemailxml_dd:MaildataJobIDmailxml_dd:CustomerGroupIDmailxml_dd:MailingGroupIDAn association with a dashed box containing two attributes:<ul style="list-style-type: none">mailxml_dd:MailBundleCountmailxml_dd:MPSCountAn association with a dashed box containing four elements, each with a multiplicity of 1..∞:<ul style="list-style-type: none">mailxml_dd:IMcbAndIMtbPiec...mailxml_dd:IMcbPieceScanInfomailxml_dd:IMtbPieceScanInfomailxml_dd:IMbScanRec
namespace	http://delivery-tech.org/Specs/mailxml26.2/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/mailxml26.2/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/mailxml26.2/mailxml_dd
annotation	documentation Delivery from USPS of full service nixie detail.

element **NixieDetailNotification**

diagram	<pre> graph LR ND[NixieDetailNotification] --- NP(()) NP --- NP_Condition(()) NP_Condition --- NP_SubmittingParty[mailxml_dd: SubmittingParty] NP_Condition --- NP_SubmittingSoftware[mailxml_dd: SubmittingSoftware] NP_Condition --- NP_CRID[mailxml_dd: CRID] NP_Condition --- NP_PushMessageID[mailxml_dd: PushMessageID] NP_Condition --- NP_Uncoupling(()) NP_Uncoupling --- NP_UserLicenseCode[mailxml_dd: UserLicenseCode] NP_Uncoupling --- NP_MaildatJobID[mailxml_dd: MaildatJobID] NP_Uncoupling --- NP_CustomerGroupID[mailxml_dd: CustomerGroupID] NP_Uncoupling --- NP_MailingGroupID[mailxml_dd: MailingGroupID] NP_Uncoupling --- NP_FSNonFSAvailability[mailxml_dd: FSNonFSAvailability...] NP_Uncoupling --- NP_NotificationDate[mailxml_dd: NotificationDate] NP_Uncoupling --- NP_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/mailxml26.2/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_Condition(()) NP_Condition --- NP_SubmittingParty[mailxml_dd: SubmittingParty] NP_Condition --- NP_SubmittingSoftware[mailxml_dd: SubmittingSoftware] NP_Condition --- NP_SubmitterTrackingID[mailxml_dd: SubmitterTrackingID] NP_Condition --- NP_RequestorCRID[mailxml_dd: RequestorCRID] NP_Condition --- NP_Uncoupling(()) NP_Uncoupling --- NP_OriginalMailerID6[mailxml_dd: OriginalMailerID6] NP_Uncoupling --- NP_OriginalMailerID9[mailxml_dd: OriginalMailerID9] NP_Uncoupling --- NP_PieceRequest[mailxml_dd: PieceRequest] NP_PieceRequest --- NP_1_oo[1..oo] NP_PieceRequest --- NP_retrieveDataBy[mailxml_dd: retrieveDataBy] </pre> <p>NixieDetailQueryRequest</p> <p>Query request for full service nixie detail.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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, there is a dashed association line pointing to a box labeled mailxml_dd:TrackingID. From this point, another dashed line leads to a box labeled mailxml_dd:SubmitterTrackingID. From SubmitterTrackingID, a solid line connects to a box labeled mailxml_dd:QueryResults. A second solid line from SubmitterTrackingID connects to a box labeled mailxml_defs:QueryError, which includes a note: "Error issued when the query data cannot be provided." To the left of the main class, a box labeled attributes contains a list of elements under grp mailxml_defs:LargeTransa...: MessageGroupID, TotalMessageCount, MessageSerialNu..., TransmittedRecor..., TotalRecordsAcro..., and LastMessage.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	documentation Response to the Query request for full service nixie detail.

element **NonComplianceDataWithPostageOwedQueryRequest**

diagram	<p>Query request for full service Non Compliance Data reports.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	documentation Query request for full service Non Compliance Data reports.

element **NonComplianceDataWithPostageOwedReportDelivery**

diagram	<pre> classDiagram class NonComplianceDataWithPostageOwedReportDelivery { 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 NonComplianceData class FSNonFSAvailability class FullServiceIndicator NonComplianceDataWithPostageOwedReportDelivery "1..∞" --> NonComplianceData NonComplianceDataWithPostageOwedReportDelivery --> SubmittingParty NonComplianceDataWithPostageOwedReportDelivery --> SubmittingSoftware NonComplianceDataWithPostageOwedReportDelivery --> DataRecipient NonComplianceDataWithPostageOwedReportDelivery --> PushMessageID NonComplianceDataWithPostageOwedReportDelivery --> FSNonFSAvailability NonComplianceDataWithPostageOwedReportDelivery --> FullServiceIndicator </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation 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". This node has two outgoing connections: one to the left and one to the right. The left connection leads to a dashed-line box containing three nodes: "mailxml_dd:SubmittingParty", "mailxml_dd:SubmittingSoftware", and "mailxml_dd:PushMessageID". The right connection leads to another dashed-line box containing several nodes: "UserLicenseCode", "MaildatJobID", "CustomerGroupID", "MailingGroupID", "eDocSenderCRID", "FSNonFSAvailability...", "NotificationDate", and "AvailableRecordCou...". Each of these right-side nodes is preceded by a small connector node with three dots, indicating they are part of a repeating group.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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:LargeTransa...] LargeTransactions --- MGID[mailxml_defs:MessageGroupID] LargeTransactions --- TMC[mailxml_defs:TotalMessageCount] LargeTransactions --- MSN[mailxml_defs:MessageSerialNu...] LargeTransactions --- TRR[mailxml_defs:TransmittedRecor...] LargeTransactions --- TRA[mailxml_defs:TotalRecordsAcro...] LargeTransactions --- LM[mailxml_defs:LastMessage] LM -.-> TrackingID[mailxml_dd:TrackingID] LM -.-> STrackingID[mailxml_dd:SubmitterTrackingID] STrackingID --- QR[mailxml_dd:QueryResults] STrackingID --- QE[mailxml_defs:QueryError] </pre> <p>Response to the Request for Full Service Non Compliance Data With Postage Owed Report Query Request</p> <p>NonComplianceDataWithPostageOwedReportQueryResponse</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:TrackingID</p> <p>mailxml_dd:SubmitterTrackingID</p> <p>mailxml_dd:QueryResults</p> <p>mailxml_defs:QueryError</p> <p>Error issued when the query data cannot be provided.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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/mailxml26.2/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>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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>The diagram illustrates the structure of the ScanSTCReconciliationNotification element. It begins with a main element 'ScanSTCReconciliationNotifica...' which contains several child elements represented by dashed boxes. These include 'SubmittingParty', 'SubmittingSoftware', 'PushMessageID', 'UserLicenseCode', 'MaildatJobID', 'CustomerGroupID', 'MailingGroupID', 'FSNonFSAvailability...', 'NonFullServiceAvail...', 'AvailableRecordCou...', 'CountType', and 'NotificationDate'. Each of these child elements is preceded by a dashed box labeled 'mailxml_dd:' followed by the element name.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/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>The diagram illustrates the structure of the ScanSTCReconciliationQueryRequest element. It begins with a main element 'ScanSTCReconciliationQueryR...' which contains several child elements represented by dashed boxes. These include 'SubmittingParty', 'SubmittingSoftware', 'SubmitterTrackingID', 'UserLicenseCode', 'MaildatJobID', 'CustomerGroupID', 'MailingGroupID', 'ConsigneeApptID', and 'retrieveDataBy'. Each of these child elements is preceded by a dashed box labeled 'mailxml_dd:' followed by the element name.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
annotation	<p>documentation</p> <p>Query request for for full service container visisbility information.</p>

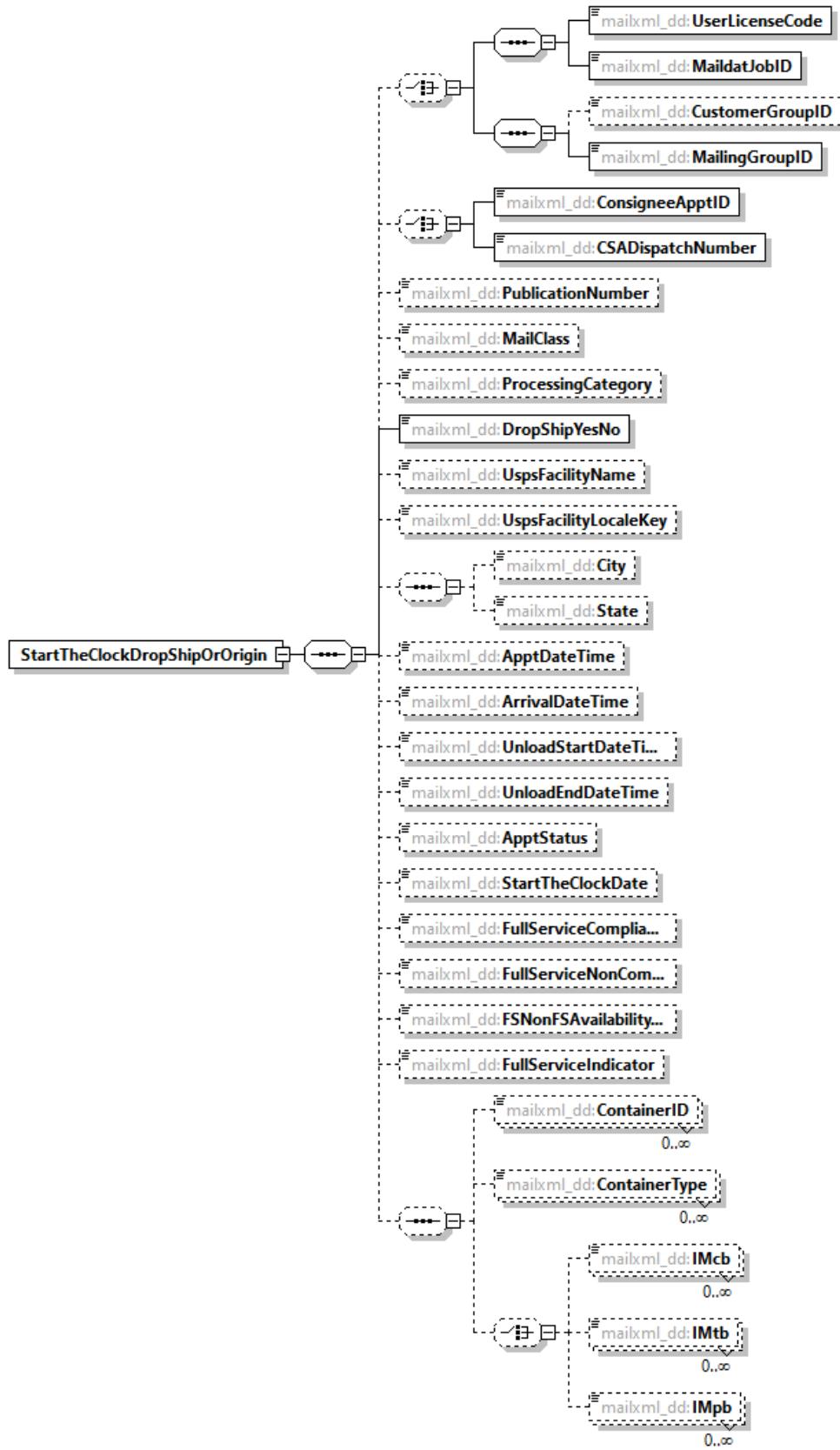
element ScanSTCReconciliationQueryResponse

element **StartTheClockBEMUBlock**

diagram	<p>A UML sequence diagram fragment showing a message exchange. On the left is a participant labeled "StartTheClockBEMUBlock". A horizontal lifeline extends to the right, ending in a hollow rectangle representing a message. This message is directed at another participant labeled "mailxml_dd:StartTheClockBEMU". A return arrow originates from the "StartTheClockBEMU" participant and points back to the "StartTheClockBEMUBlock" participant. Below the lifelines, the multiplicity "1..∞" is indicated.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

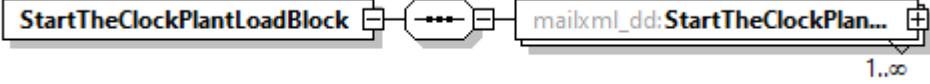
element **StartTheClockDropShipOrOrigin**

diagram



namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
-----------	---

element **StartTheClockPlantLoadBlock**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

complexType **addressCorrectionAddressType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

complexType **byForConflictType**

diagram	<p>The diagram illustrates the structure of the byForConflictType complex type. It starts with a base class conflictType (represented by a dashed rectangle). An inheritance arrow points from byForConflictType to conflictType. From conflictType, two arrows point to specific attributes: mailxml_dd:UserLicenseCode and mailxml_dd:MaildatJobID. Another arrow points to a group of three attributes: mailxml_dd:CustomerGroupID, mailxml_dd:MailingGroupID, and mailxml_dd:MailOwnerConflict (with multiplicity 0..∞). A final arrow points to mailxml_dd:MailPreparerConflict (with multiplicity 0..∞). Below these, there are three more attributes: mailxml_dd:FSNonFSAvailability..., mailxml_dd:FullServiceIndicator, and mailxml_dd:ReturnInfo.</p>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

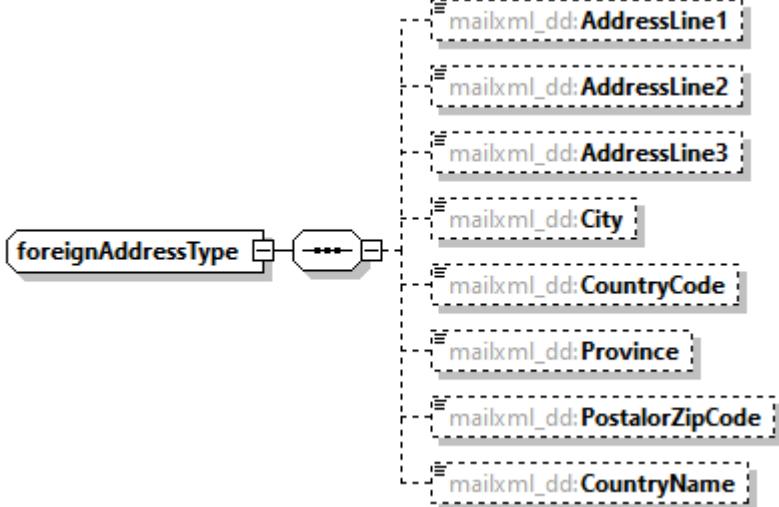
complexType **clockStartedType**

diagram	<p>The diagram illustrates the structure of the clockStartedType complex type. It starts with a base class startedType (represented by a dashed rectangle). An inheritance arrow points from clockStartedType to startedType. From startedType, three arrows point to specific attributes: 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/mailxml26.2/mailxml_dd

complexType **dqrContainerInfoType**

diagram	<pre> classDiagram dqrContainerInfoType < -- mailxml_dd 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>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

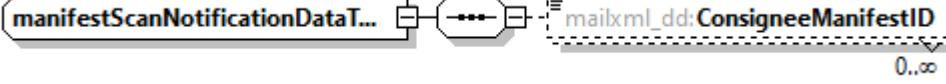
complexType **foreignAddressType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

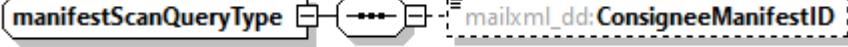
complexType **manifestScanEventDetailType**

diagram	<pre> graph TD A[manifestScanEventDetailType] --> B[mailxml_dd:USPSEventExtractFi...] B --> C[mailxml_dd:IMpb] C --> D[mailxml_dd:ElectronicFileNumber] D --> E[mailxml_dd:MailerID] E --> F[mailxml_dd:MailerName] F --> G[mailxml_dd:DestinationZipCode] G --> H[mailxml_dd:DestinationZipPlusF...] H --> I[mailxml_dd:ScanningFacilityZip] I --> J[mailxml_dd:ScanningFacilityNa...] J --> K[mailxml_dd:EventCode] K --> L[mailxml_dd:EventName] L --> M[mailxml_dd:EventDate] M --> N[mailxml_dd:EventTime] N --> O[mailxml_dd:MailerOwnerID] O --> P[mailxml_dd:CustomerReferenc...] P --> Q[mailxml_dd:DestinationCountry...] Q --> R[mailxml_dd:RecipientName] R --> S[mailxml_dd:OriginalLabel] S --> T[mailxml_dd:UnitofMeasureCode] T --> U[mailxml_dd:Weight] U --> V[mailxml_dd:GuaranteedDeliver...] V --> W[mailxml_dd:GuaranteedDeliver...] W --> X[mailxml_dd:LogisticsManager...] </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

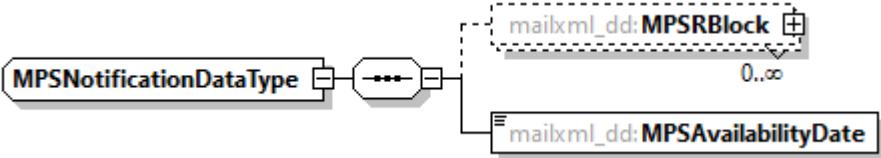
complexType **manifestScanNotificationDataType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

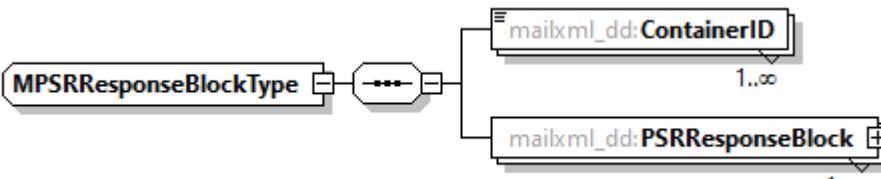
complexType **manifestScanQueryType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

complexType **MPSNotificationDataType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

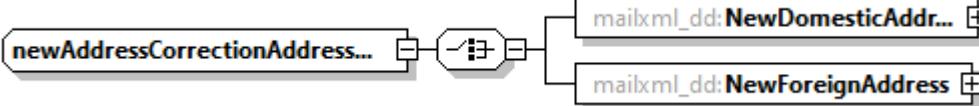
complexType **MPSRResponseBlockType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

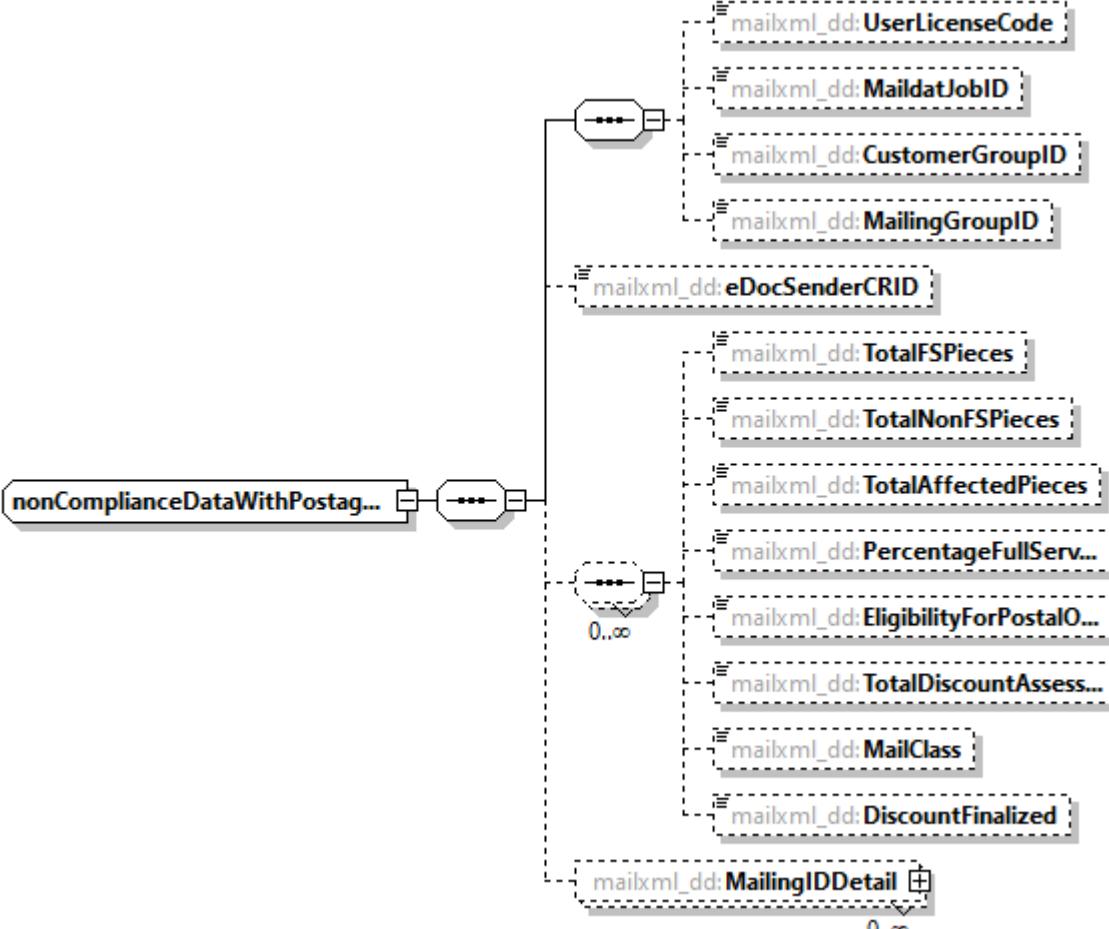
complexType **MPSVisScanQueryType**

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/mailxml26.2/mailxml_dd

complexType **newAddressCorrectionAddressType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

complexType **nonComplianceDataWithPostageOwedReportType**

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

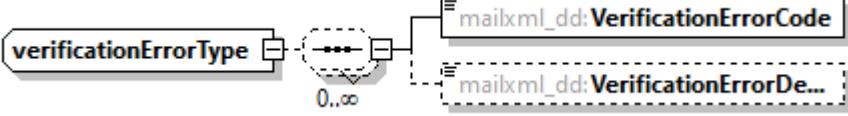
complexType **PSRResponseBlockType**

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/mailxml26.2/mailxml_dd

complexType **unManifestedScanEventDetailType**

diagram	<pre> classDiagram class unManifestedScanEventDetailT... class mailxml_dd.USPSEventExtractFi... class mailxml_dd.IMpb class mailxml_dd.ElectronicFileNumber class mailxml_dd.MailerID class mailxml_dd.MailerName class mailxml_dd.DestinationZipCode class mailxml_dd.DestinationZipPlusF... class mailxml_dd.ScanningFacilityZip class mailxml_dd.ScanningFacilityNa... class mailxml_dd.EventCode class mailxml_dd.EventName class mailxml_dd.EventDate class mailxml_dd.EventTime class mailxml_dd.MailerOwnerID class mailxml_dd.CustomerReferenc... class mailxml_dd.DestinationCountry... class mailxml_dd.RecipientName class mailxml_dd.OriginalLabel class mailxml_dd.UnitofMeasureCode class mailxml_dd.Weight class mailxml_dd.GuaranteedDeliver... class mailxml_dd.GuaranteedDeliver... class mailxml_dd.LogisticsManager... unManifestedScanEventDetailT... "1" --> mailxml_dd.USPSEventExtractFi... unManifestedScanEventDetailT... "1" --> mailxml_dd.IMpb unManifestedScanEventDetailT... "1" --> mailxml_dd.ElectronicFileNumber unManifestedScanEventDetailT... "1" --> mailxml_dd.MailerID unManifestedScanEventDetailT... "1" --> mailxml_dd.MailerName unManifestedScanEventDetailT... "1" --> mailxml_dd.DestinationZipCode unManifestedScanEventDetailT... "1" --> mailxml_dd.DestinationZipPlusF... unManifestedScanEventDetailT... "1" --> mailxml_dd.ScanningFacilityZip unManifestedScanEventDetailT... "1" --> mailxml_dd.ScanningFacilityNa... unManifestedScanEventDetailT... "1" --> mailxml_dd.EventCode unManifestedScanEventDetailT... "1" --> mailxml_dd.EventName unManifestedScanEventDetailT... "1" --> mailxml_dd.EventDate unManifestedScanEventDetailT... "1" --> mailxml_dd.EventTime unManifestedScanEventDetailT... "1" --> mailxml_dd.MailerOwnerID unManifestedScanEventDetailT... "1" --> mailxml_dd.CustomerReferenc... unManifestedScanEventDetailT... "1" --> mailxml_dd.DestinationCountry... unManifestedScanEventDetailT... "1" --> mailxml_dd.RecipientName unManifestedScanEventDetailT... "1" --> mailxml_dd.OriginalLabel unManifestedScanEventDetailT... "1" --> mailxml_dd.UnitofMeasureCode unManifestedScanEventDetailT... "1" --> mailxml_dd.Weight unManifestedScanEventDetailT... "1" --> mailxml_dd.GuaranteedDeliver... unManifestedScanEventDetailT... "1" --> mailxml_dd.GuaranteedDeliver... unManifestedScanEventDetailT... "1" --> mailxml_dd.LogisticsManager... </pre>
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

complexType verificationErrorType

diagram	
namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd

simpleType addressCorrectionMoveType

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

simpleType addressTypeType

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

simpleType containerScanStateType

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string
annotation	documentation Scan data for Container Scan States

simpleType deliverabilityCodeType

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

simpleType eDocTypeType

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

simpleType primarySecondaryIndicatorType

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

simpleType **verificationErrorType**

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

simpleType **verificationWarningType**

namespace	http://delivery-tech.org/Specs/mailxml26.2/mailxml_dd
type	restriction of xs:string

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