SDMX Standards: Section 3A PaRT VI

SDMX-ML:

Schema and Documentation

Registry Namespace

Version 3.0

September 2021

© SDMX 2021

http://www.sdmx.org/

Contents

[1 Introduction 1](#_Toc82186349)

[2 Schema Documentation 1](#_Toc82186350)

[2.1 Registry Namespace 1](#_Toc82186351)

[2.1.1 Summary 1](#_Toc82186352)

[2.1.2 Complex Types 1](#_Toc82186353)

[2.1.3 Simple Types 33](#_Toc82186354)

# Introduction

The registry schemas have been updated to reflect the various changes in the standard and to introduce new constructs for managing subscriptions. In addtion, the constructs that had existed for querying structural metadata have been removed as this is now handled by the query messages.

As was done with the query, the registry message set now contains distinct messages for each operation. The messages will eventually allow the registry to function in more standard web service mannter.

# Schema Documentation

## Registry Namespace

**http://www.sdmx.org/resources/sdmxml/schemas/v3\_0/registry**

### Summary

Referenced Namespaces:

| **Namespace** | **Prefix** |
| --- | --- |
| http://www.sdmx.org/resources/sdmxml/schemas/v3\_0/common | common |
| http://www.sdmx.org/resources/sdmxml/schemas/v3\_0/structure | structure |
| http://www.w3.org/2001/XMLSchema | xs |

Contents:

39 Complex Types  
5 Simple Types

### Complex Types

**RegistrationType:** Registration provides the information needed for data and reference metadata set registration. A data source must be supplied here if not already provided in the provision agreement. The data set or metadata set must be associated with a provision agreement, a metadata flow, or a dataflow definition. If possible, the provision agreement should be specified. Only in cases where this is not possible should the dataflow or metadata flow be used.

Attributes:

id?, validFrom?, validTo?, lastUpdated?, indexTimeSeries?, indexDataSet?, indexAttributes?, indexReportingPeriod?

Content:

ProvisionAgreement, Datasource

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| id | IDType | The id attribute holds a registry assigned identification for the registration. This must be provided in a response message (unless an error occurred while submitting a new registration), and should be included when updating or deleting a registration. |
| validFrom | xs:dateTime | The validFrom attribute provides an inclusive start date for providing supplemental validity information about the registration, so that data visibility from the registry can be controlled by the registrant. |
| validTo | xs:dateTime | The validFrom attribute provides an inclusive end date for providing supplemental validity information about the registration, so that data visibility from the registry can be controlled by the registrant. |
| lastUpdated | xs:dateTime | The lastUpdated attribute provides a timestamp for the last time the data source was updated. |
| indexTimeSeries (default: false) | xs:boolean | The indexTimeSeries, if true, indicates that the registry must index all time series for the registered data. The default value is false, and the attribute will always be assumed false when the provision agreement references a metadata flow. |
| indexDataSet (default: false) | xs:boolean | The indexDataSet, if true, indicates that the registry must index the range of actual (present) values for each dimension of the data set or identifier component of the metadata set (as indicated in the set's structure definition). The default value is false. |
| indexAttributes (default: false) | xs:boolean | The indexAttributes, if true, indicates that the registry must index the range of actual (present) values for each attribute of the data set or the presence of the metadata attributes of the metadata set (as indicated in the set's structure definition). The default value is false. |
| indexReportingPeriod (default: false) | xs:boolean | The indexReportingPeriod, if true, indicates that the registry must index the time period ranges for which data is present for the data source. The default value is false, and the attribute will always be assumed false when the provision agreement references a metadata flow. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| ProvisionAgreement | ProvisionAgreementRe ferenceType | ProvisionAgreement provides a reference to the provision agreement that the data is being registered against. |
| Datasource | DataSourceType | Datasource identifies the data source(s) where the registered data can be retrieved. |

**DataSourceType:** DataSourceType specifies the properties of a data or metadata source. Either a simple data source, a queryable data source, or both must be specified.

Content:

(SimpleDataSource | QueryableDataSource)[1..2]

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| SimpleDataSource | xs:anyURI | SimpleDatasource describes a data source that is an SDMX-ML data or metadata message. It requires only the URL of the data. |
| QueryableDataSource | QueryableDataSourceT ype | QueryableDatasource describes a data source that can be queried using the SDMX REST interfaces. |

**SimpleDataSourceType:** SimpleDataSourceType describes a simple data source. The URL of the data is contained in the content.

Derivation:

xs:anySimpleType (restriction)   
   xs:anyURI (extension)   
         SimpleDataSourceType

Attributes:

TYPE?

Content:

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| TYPE (fixed: SIMPLE) | xs:string | TYPE is a fixed attribute that is used to ensure only one simple data source may be provided, when it is referenced in a uniqueness constraint. |

**QueryableDataSourceType:** QueryableDataSourceType describes a queryable data source, and add a fixed attribute for ensuring only one queryable source can be provided.

Derivation:

QueryableDataSourceType (extension)   
   QueryableDataSourceType

Attributes:

isRESTDatasource, isWebServiceDatasource, TYPE?

Content:

DataURL, WSDLURL?, WADLURL?

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| isRESTDatasource | xs:boolean | The isRESTDatasource attribute indicates, if true, that the queryable data source is accessible via the REST protocol. |
| isWebServiceDatasource | xs:boolean | The isWebServiceDatasource attribute indicates, if true, that the queryable data source is accessible via Web Services protocols. |
| TYPE (fixed: QUERY) | xs:string | TYPE is a fixed attribute that is used to ensure only one queryable data source may be provided, when it is referenced in a uniqueness constraint. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| DataURL | xs:anyURI | DataURL contains the URL of the data source. |
| WSDLURL | xs:anyURI | WSDLURL provides the location of a WSDL instance on the internet which describes the queryable data source. |
| WADLURL | xs:anyURI | WADLURL provides the location of a WADL instance on the internet which describes the REST protocol of the queryable data source. |

**IdentifiableQueryType:** IdentifiableQueryType describes the structure of a query for an identifiable object.

Attributes:

id?

Content:

{Empty}

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| id (default: %) | IDQueryType | The id attribute is used to query for an object based on its identifier. This is either an explicit value, or completely wild cared with the "%" value. |

**VersionableQueryType:** VersionableQueryType describes the structure of a query for a versionable object.

Derivation:

IdentifiableQueryType (extension)   
   VersionableQueryType

Attributes:

id?, version?

Content:

{Empty}

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| id (default: %) | IDQueryType | The id attribute is used to query for an object based on its identifier. This is either an explicit value, or completely wild cared with the "%" value. |
| version (default: \*) | VersionQueryType | The version attribute is used to query for an object based on its version. This can be and explicit value, wild-carded ("%"), or late-bound ("\*"). A wild carded version will match any version of the object where as a late-bound version will only match the latest version. |

**MaintainableQueryType:** MaintainableQueryType describes the structure of a query for a maintainable object.

Derivation:

IdentifiableQueryType (extension)   
   VersionableQueryType (extension)   
         MaintainableQueryType

Attributes:

id?, version?, agencyID?

Content:

{Empty}

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| id (default: %) | IDQueryType | The id attribute is used to query for an object based on its identifier. This is either an explicit value, or completely wild cared with the "%" value. |
| version (default: \*) | VersionQueryType | The version attribute is used to query for an object based on its version. This can be and explicit value, wild-carded ("%"), or late-bound ("\*"). A wild carded version will match any version of the object where as a late-bound version will only match the latest version. |
| agencyID (default: %) | NestedIDQueryType | The agencyID attribute is used to query for an object based on its maintenance agency's identifier. This is either an explicit value, or completely wild cared with the "%" value. |

**StatusMessageType:** StatusMessageType carries the text of error messages and/or warnings in response to queries or requests.

Attributes:

status

Content:

MessageText\*

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| status | StatusType | The status attribute carries the status of the query or request. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| MessageText | StatusMessageType | MessageText contains the text of the error and/or warning message. It can occur multiple times to communicate message for multiple warnings or errors. |

**SubmitRegistrationsRequestType:** SubmitRegistrationsRequestType defines the payload of a request message used to submit addtions, updates, or deletions of data/metadata set registrations.

Content:

RegistrationRequest+

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| RegistrationRequest | RegistrationRequestT ype | RegistrationRequest provides the details of a requested registration and the action to take on it. A reference to a provision agreement that exists in the registry must be provide along with a simple and/or queryable data source. The id should only be provided when updating or deleting a registration. |

**RegistrationRequestType:** RegistrationRequestType describes the structure of a single registration request. It contains the details of a registation and an action field to indicate the action to be taken on the contained registration.

Attributes:

action

Content:

Registration

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| action | ActionType | The action attribute indicates whether this is an addition, a modification, or a deletion of a registration. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Registration | RegistrationType | Registration contains the details of the data/metadata set registration to be added, updated, or deleted. |

**SubmitRegistrationsResponseType:** SubmitRegistrationsResponseType describes the structure of a registration response. For each submitted registration in the request, a registration status is provided. The status elements should be provided in the same order as the submitted registrations, although each status will echo the request to ensure accurate processing of the responses.

Content:

RegistrationStatus+

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| RegistrationStatus | RegistrationStatusTy pe | RegistrationStatus provided the status details for the submitted registration. It echoes the original submission and provides status information about the request. |

**RegistrationStatusType:** RegistrationStatusType describes the structure of a registration status.

Content:

Registration, StatusMessage

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Registration | RegistrationType | Registration, at the very least echoes the submitted registration. It the request was to create a new registration and it was successful, an id must be supplied. |
| StatusMessage | StatusMessageType | StatusMessage provides that status for the registration request, and if necessary, any error or warning information. |

**QueryRegistrationRequestType:** QueryRegistrationRequestType describes the structure of a registration query request. The type of query (data, reference metadata, or both) must be specified. It is possible to query for registrations for a particular provision agreement, data provider, or structure usage, or to query for all registrations in the registry. In addition the search can be refined by providing constraints in the form of explicit time periods, constraint regions, and key sets. When constraint regions and key sets are provided they will be effectively processed by first matching all data for the included keys and regions (all available data if there are none) and then removing any data matching the excluded keys or regions.

Attributes:

returnConstraints?

Content:

QueryType, (All | ProvisionAgreement | DataProvider | Dataflow | Metadataflow), ReferencePeriod?, (DataKeySet | CubeRegion | MetadataTargetRegion)\*

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| returnConstraints (default: false) | xs:boolean | The returnConstraints attribute determines whether information about the constraints on the data or metadata sets returned should also be sent the results. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| QueryType | QueryTypeType | QueryType defines the type of sets (data, metadata, or both) that are being queried for. |
| All | EmptyType | All indicates that all registrations meeting the other criteria of the query should be returned. |
| ProvisionAgreement | ProvisionAgreementRe ferenceType | ProvisionAgreement provides a reference to a provision agreement in the registry, for which all registered sets meeting the other criteria of this query should be returned. The reference is provided as a URN and/or a complete set of reference fields. |
| DataProvider | DataProviderReferenc eType | DataProvider provides a reference to a data provider in the registry, for which all registered sets meeting the other criteria of this query should be returned. The reference is provided as a URN and/or a complete set of reference fields. |
| Dataflow | DataflowReferenceTyp e | Dataflow provides a reference to a data flow in the registry, for which all registered sets meeting the other criteria of this query should be returned. The reference is provided as a URN and/or a complete set of reference fields. |
| Metadataflow | MetadataflowReferenc eType | Metadataflow provides a reference to a metadata flow in the registry, for which all registered sets meeting the other criteria of this query should be returned. The reference is provided as a URN and/or a complete set of reference fields |
| ReferencePeriod | ReferencePeriodType | ReferencePeriod provides an inclusive start and end date for the data or metadata being sought. |
| DataKeySet | DataKeySetType | DataKeySet is used to provide a set of included or excluded keys which serves to refine the data being sought. |
| CubeRegion | CubeRegionType | CubeRegion is used to provide sets of include or excluded values for dimensions when querying for data. |
| MetadataTargetRegion | MetadataTargetRegion Type | MetadataTargetRegion is used to provide sets of included or excluded values for identifier components when querying for metadata. |

**QueryRegistrationResponseType:** QueryRegistrationResponseType describes the structure of a registration query response. It provides a status for the request, and if successful, the resulting data and/or metadata results.

Content:

StatusMessage, QueryResult\*

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| StatusMessage | StatusMessageType | StatusMessage provides that status for the registration query request, and if necessary, any error or warning information. |
| QueryResult | QueryResultType | QueryResult contains a result for a successful registration query. It can occur multiple times, for each registration the meets the conditions specified in the query. |

**QueryResultType:** QueryResultType describes the structure of a query result for a single data source. Either a data result or metadata result is detailed, depending on the data source.

Attributes:

timeSeriesMatch

Content:

(DataResult | MetadataResult)

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| timeSeriesMatch | xs:boolean | The timeSeriesMatch attribute is true when the result is an exact match with the key found in the registry - that is, when the registered data source provides a matching key. It is set to false when the data source is registered with cube-region constraints, or in any other circumstance when it cannot be established that the sought-for keys have been exactly matched. This is always true when the resulting data source is the source of a metadata set. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| DataResult | ResultType |  |
| MetadataResult | ResultType |  |

**ResultType:** ResultType contains the details about a data or metadata source, through the complete registration information. In addition, a reference to the content constraints for the data source may be provided, if the query requested this information.

Content:

Registration, Constraint\*

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Registration | RegistrationType | Registration provides the details of a matching registration. In addition to the data source and provision agreement information, the id of the registration must also be provided. |
| Constraint | ConstraintReferenceT ype | Constraint provides a reference to a data or metadata constraint in the registry for the resulting data source (or possibly constraints base on the registration provision agreement, data provider, structure usage, or structure). The reference is provided for by a URN and/or a complete set of reference fields. |

**SubmitStructureRequestType:** SubmitStructureRequestType describes the structure of a structure submission. Structural components are provided either in-line or referenced via a SDMX-ML Structure message external to the registry. A default action and external reference resolution action are all provided for each of the contained components, but can be overridden on a per component basis.

Attributes:

action?, externalDependencies?

Content:

(StructureLocation | Structures), SubmittedStructure\*

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| action (default: Append) | ActionType | The action attribute indicates the default action (append-add, replace-update, delete, or no action-informational) to be taken on all structural components in either the external SDMX-ML Structure message or the in-line components. The default action is Append. The Replace action is not applicable to final structures in the repository, and will produce an error condition, as these can be versioned but not modified. To submit a later version of a structural object, the object should include the incremented version number. |
| externalDependencies (default: false) | xs:boolean | The externalDependencies attribute indicates the default resolution of external dependencies. This should be set to true if the repository is expected to use external reference URLs in the structural components to retrieve any externally referenced objects that is used by a non-external object. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| StructureLocation | xs:anyURI | StructureLocation provides the location of a SDMX-ML Structure message, external to the repository that can be retrieved by the repository submission service. |
| Structures | StructuresType | Structures allows for the inline definition of structural components for submission. |
| SubmittedStructure | SubmittedStructureTy pe | SubmittedStructure contains a reference to one of the structural maintainable artefacts detailed in the external SDMX-ML Structure message or in-line and provides an override for the default action. This should only be used if the action to be performed on the referenced structural object is different than the default action. For example, one may want to append all structural components of a structure message, save one codelist. This codelist could be referenced in a submitted structure element and given an action of Informational. |

**SubmittedStructureType:** SubmittedStructureType generally references a submitted structural object. When used in a submit structure request, its purpose is to override the default action or external dependency resolution behavior. If neither of these indicators are set, then it will be ignored. In a submit structure response, it is used to reference a submitted object for the purpose of providing a status for the submission. In this case, the action attribute should be populated in order to echo the requested action.

Attributes:

action?, externalDependencies?

Content:

MaintainableObject

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| action | ActionType | The action attribute will indicate the action to be taken on the referenced structural object. This should be provided when this structure is used in a submit structure response. |
| externalDependencies | xs:boolean | The externalDependencies attribute should be set to true if the repository is expected to use external reference URLs in the structural components to retrieve objects on which the referenced object has dependencies. (Thus, if a data structure referenced here is being submitted to the repository, and the structure message has URLs which point to the locations of the codelists it uses, then this attribute should be set to true). This should not be provided when this structure is used in a submit structure response. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| MaintainableObject | MaintainableUrnRefer enceType |  |

**SubmitStructureResponseType:** SubmitStructureResponseType describes the structure of a response to a structure submission. For each submitted structure, a Result will be returned.

Content:

SubmissionResult+

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| SubmissionResult | SubmissionResultType | SubmissionResult provides a status for each submitted structural object. |

**SubmissionResultType:** SubmissionResultType provides the status of the structural object submission request. It will identify the object submitted, report back the action requested, and convey the status and any error messages which are relevant to the submission.

Content:

SubmittedStructure, StatusMessage

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| SubmittedStructure | SubmittedStructureTy pe | SubmittedStructure provides a reference to the submitted structural object and echoes back the action requested for it. |
| StatusMessage | StatusMessageType | StatusMessage provides that status for the submission of the structural object, and if necessary, any error or warning information. |

**SubmitSubscriptionsRequestType:** SubmitSubscriptionsRequestType defines the payload of a request message used to submit addtions, updates, or deletions of subscriptions. Subscriptions are submitted to the registry to subscribe to registration and change events for specific registry resources.

Content:

SubscriptionRequest+

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| SubscriptionRequest | SubscriptionRequestT ype |  |

**SubscriptionType:** SubscriptionType describes the details of a subscription to a registration or change event for registry resources. When it occurs as the content of a response message, the registry URN must be provide, unless the response is a failure notification for the creation of a new subscription.

Content:

Organisation, RegistryURN?, NotificationMailTo\*, NotificationHTTP\*, SubscriberAssignedID?, ValidityPeriod, EventSelector

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Organisation | OrganisationReferenc eType | Organisation provides a reference to the organisation that owns this subscription. The reference is provided via a URN and/or a complete set of reference fields. |
| RegistryURN | xs:anyURI | RegistryURN is used to identify the subscription in the case of deletion or modification. This should be provided in all response messages, unless the response it a notification of the failure to create a newly submitted subscription - in which case there will be no registry assigned URN. |
| NotificationMailTo | NotificationURLType | NotificationMailTo holds an e-mail address (the "mailto:" protocol). Multiple email address can be notified for a single subscription. |
| NotificationHTTP | NotificationURLType | NotificationHTTP holds an http address to which notifications can be addressed as POSTs. Multiple http address may be notified for a single subscription event. |
| SubscriberAssignedID | IDType | SubscriberAssignedID allows the subscriber to specify an identification which will be returned as part of the notification for the subscribed events. This should be used if multiple new requests are made, so that the responses can be accurately correlated to the requests. |
| ValidityPeriod | ValidityPeriodType | Validity period sets a start and end date for the subscription. |
| EventSelector | EventSelectorType | EventSelector indicates an event or events for the subscription. |

**SubscriptionRequestType:** SubscriptionRequestType describes the structure of a single subscription request. It contains subscription details and an action field to indicate the action to be taken on the contained subscription. Note that if the action is update or delete, then the registry supplied URN for the subscription must be included.

Attributes:

action

Content:

Subscription

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| action | ActionType | The action attribute indicates whether this is an addition, a modification, or a deletion of a subscription. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Subscription | SubscriptionType | Subscription contains the details of the subscription to be added, updated, or deleted. |

**SubmitSubscriptionsResponseType:** SubmitSubscriptionsResponseType describes the structure of the response to a new subscription submission. A status is provided for each subscription in the request.

Content:

SubscriptionStatus+

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| SubscriptionStatus | SubscriptionStatusTy pe | SubscriptionStatus contains information which describes the success or failure of a subscription request, providing any error messages in the event of failure. The statuses should occur in the same order as the requests when responding to a message with multiple subscription requests. If a subscriber-assigned identification for the subscription is provided, it will be returned to allow for accurate matching of the responses to the requests. A registry assigned URN will be returned for any successfully created, updated, or deleted subscription. |

**SubscriptionStatusType:** SubscriptionStatusType describes the structure a status for a single subscription request.

Content:

SubscriptionURN?, SubscriberAssignedID?, StatusMessage

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| SubscriptionURN | xs:anyURI | SubscriptionURN contains the registry generated URN for the subscription, and will be returned for any successfully created, updated, or deleted subscription. |
| SubscriberAssignedID | IDType | SubscriberAssignedID is the id assigned by the subscriber to the subscription. If it existed in the subscription request, it will be returned. |
| StatusMessage | StatusMessageType | StatusMessage provides that status for the subscription request, and if necessary, any error or warning information. |

**QuerySubscriptionRequestType:** QuerySubscriptionRequestType describes the structure of a query for subscriptions. Subscriptions for a given organisation may be retrieved.

Content:

Organisation

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Organisation | OrganisationReferenc eType | Organisation provides a reference to the data consumer for which the subscription details should be returned. |

**QuerySubscriptionResponseType:** QuerySubscriptionResponseType describes the structure of a subscription query response. A status will describe the success or failure of the request (and provide error or warning messages if necessary). If the query was successful, details of all of the organisation's subscriptions will be provided.

Content:

StatusMessage, Subscription\*

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| StatusMessage | StatusMessageType | StatusMessage provides that status for the query subscription request, and if necessary, any error or warning information. |
| Subscription | SubscriptionType | Subscription contains the details of a subscription for the organisation. This may occur multiple times for each subscription. |

**NotifyRegistryEventType:** NotifyRegistryEventType describes the structure a registry notification, in response to a subscription to a registry event. At a minimum, the event time, a reference to the change object, a reference to the underlying subscription triggering the notification, and the action that took place on the object are sent. In addition, the full details of the object may be provided at the discretion of the registry. In the event that the details are not sent, it will be possible to query for the details of the changed object using the reference provided.

Content:

EventTime, (ObjectURN | RegistrationID), SubscriptionURN, EventAction, (StructuralEvent | RegistrationEvent)?

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| EventTime | xs:dateTime | EventTime specifies the time of the triggering event. |
| ObjectURN | xs:anyURI | ObjectURN provides the URN of the object on which the event occurred, unless the event is related to a registration, in which case the RegistrationID element should be used. |
| RegistrationID | IDType | RegistrationID provides the id of the registration that underwent an event. |
| SubscriptionURN | xs:anyURI | SubscriptionURN provides the registry/repository URN of the subscription that is the cause of this notification being sent. |
| EventAction | ActionType | EventAction indicates the nature of the event - whether the action was an addition, a modification, or a deletion. |
| StructuralEvent | StructuralEventType | StructuralEvent is used to provide the details of the structural object that has changed. |
| RegistrationEvent | RegistrationEventTyp e | RegistrationEvent is used to provide the details or the registration object that has changed. |

**NotificationURLType:** NotificationURLType describes the structure of an http or email address. The content holds the addresses while an attribute indicates whether or not is expects the contents in a SOAP message.

Derivation:

xs:anySimpleType (restriction)   
   xs:anyURI (extension)   
         NotificationURLType

Attributes:

isSOAP?

Content:

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| isSOAP (default: false) | xs:boolean | The isSOAP attribute, if true, indicates the provided URL expects the notification to be sent in a SOAP message. |

**ValidityPeriodType:** ValidityPeriodType specifies inclusive start and end-dates for the subscription period.

Content:

StartDate, EndDate

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| StartDate | xs:date | StartDate is an inclusive start date for the subscription. |
| EndDate | xs:date | EndDate is an inclusive end date for the subscription. |

**EventSelectorType:** EventSelectorType describes the details of the events for a subscription. It allows subscribers to specify registry and repository events for which they wish to receive notifications.

Content:

(StructuralRepositoryEvents | DataRegistrationEvents | MetadataRegistrationEvents)[1..3]

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| StructuralRepository Events | StructuralRepository EventsType | StructuralRepositoryEvents details structural events for the subscription. |
| DataRegistrationEven ts | DataRegistrationEven tsType | DataRegistrationEvents details the data registration events for the subscription. |
| MetadataRegistration Events | MetadataRegistration EventsType | MetadataRegistrationEvents details the metadata registration events for the subscription. |

**StructuralRepositoryEventsType:** StructuralRepositoryEventsType details the structural events for the subscription. At least one maintenance agency must be specified, although it may be given a wildcard value (meaning the subscription is for the structural events of all agencies). This can also be a list of agencies to allow the subscription to subscribe the events of more than one agency. It should be noted that when doing so, all of the subsequent objects are assumed to apply to every agency in the list. The subscription is then refined by detailing the structural objects maintained by the agency for which the subscription should apply. It is possible to explicitly select all object events, all objects of given types, or to individually list out specific objects. Note that for any object, it is also possible to provide an explicit URN to reference a distinct object. In this case, the reference to maintenance agency described above is ignored. Although it is not required, if specific objects are being referenced by explicit URNs, it is good practice to list the agencies.

Attributes:

TYPE?

Content:

AgencyID+, (AllEvents | (AgencyScheme | DataConsmerScheme | DataProviderScheme | OrganisationUnitScheme | Dataflow | Metadataflow | CategoryScheme | Categorisation | Codelist | HierarchicalCodelist | ConceptScheme | MetadataStructureDefinition | KeyFamily | StructureSet | ReportingTaxonomy | Process | AttachmentConstraint | ContentConstraint | ProvisionAgreement | TransformationScheme | NameAliasScheme | NamePersonalisationScheme | RulesetScheme | UserDefinedOperatorScheme)+)

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| TYPE (fixed: STRUCTURE) | xs:string | TYPE is a fixed attribute that is used to ensure only of each event selector may be provided, when it is referenced in a uniqueness constraint. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| AgencyID | NestedIDQueryType | AgencyID specifies a maintenance agency for the object or objects indicated in the other fields. This can be either a specific ID, or a single wildcard character ("%"). A wild card character can be used to select all agencies, allowing a subscription to all events for particular object types. This can occur multiple times to list a of group of maintenance agencies, creating event subscriptions for all of the subsequent objects for each agency in the group. Note that if multiple agencies are supplied, then the wildcard character should not be used for any of them. |
| AllEvents | EmptyType | AllEvents creates a subscription to structural events for all structural objects maintained by the agencies referenced. |
| AgencyScheme | VersionableObjectEve ntType | AgencyScheme is used to subscribe to changes of agency schemes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| DataConsmerScheme | VersionableObjectEve ntType | DataConsmerScheme is used to subscribe to changes of data consumer schemes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| DataProviderScheme | VersionableObjectEve ntType | DataProviderScheme is used to subscribe to changes of data provider schemes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| OrganisationUnitSche me | VersionableObjectEve ntType | OrganisationUnitScheme is used to subscribe to changes of organisation unit schemes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| Dataflow | VersionableObjectEve ntType | Dataflow is used to subscribe to changes of data flows. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| Metadataflow | VersionableObjectEve ntType | Metadataflow is used to subscribe to changes of metadata flows. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| CategoryScheme | VersionableObjectEve ntType | CategoryScheme is used to subscribe to changes of category schemes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| Categorisation | IdentifiableObjectEv entType | Categorisation is used to subscribe to changes of categorizations. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id field can be selected. |
| Codelist | VersionableObjectEve ntType | Codelist is used to subscribe to changes of code lists. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| HierarchicalCodelist | VersionableObjectEve ntType | HierarchicalCodelist is used to subscribe to changes of hierarchical code lists. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| ConceptScheme | VersionableObjectEve ntType | ConceptScheme is used to subscribe to changes of concept schemes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| MetadataStructureDef inition | VersionableObjectEve ntType | MetadataStructureDefinition is used to subscribe to changes of metadata structure definitions. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| KeyFamily | VersionableObjectEve ntType | KeyFamily is used to subscribe to changes of key families. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| StructureSet | VersionableObjectEve ntType | StructureSet is used to subscribe to changes of structure sets. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| ReportingTaxonomy | VersionableObjectEve ntType | ReportingTaxonomy is used to subscribe to changes of reporting taxonomies. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| Process | VersionableObjectEve ntType | Process is used to subscribe to changes of processes. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| AttachmentConstraint | VersionableObjectEve ntType | AttachmentConstraint is used to subscribe to changes of attachment constraints. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| ContentConstraint | VersionableObjectEve ntType | ContentConstraint is used to subscribe to changes of content constraints. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| ProvisionAgreement | VersionableObjectEve ntType | ProvisionAgreement is used to subscribe to changes of a provision agreement. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| TransformationScheme | VersionableObjectEve ntType | TransformationScheme is used to subscribe to changes of a transformation scheme. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| NameAliasScheme | VersionableObjectEve ntType | NameAliasScheme is used to subscribe to changes of a name alias scheme. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| NamePersonalisationS cheme | VersionableObjectEve ntType | NamePersonalisationScheme is used to subscribe to changes of a name peronsalisation scheme. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| RulesetScheme | VersionableObjectEve ntType | RulesetScheme is used to subscribe to changes of a ruleset scheme. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |
| UserDefinedOperatorS cheme | VersionableObjectEve ntType | UserDefinedOperatorScheme is used to subscribe to changes of a user defined operator scheme. The maintenance agencies of the object are those identified in the AgencyID collection, effectively making separate version of this query for each agency specified. The agency is ignored if the content of this is a URN, which references an explicit object. Otherwise, either all objects of this type or specific object according to the id and version fields can be selected. |

**IdentifiableObjectEventType:** IdentifiableObjectEventType describes the structure of a reference to an identifiable object's events. Either all instances of the object matching the inherited criteria, a specific instance, or specific instances of the object may be selected.

Content:

(All | URN | ID)

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| All | EmptyType | All subscribes to the events all instances of the containing object meeting the other criteria specified. |
| URN | xs:anyURI | URN subscribes to the events of the specific instance of the object type referenced by this URN. Note that when this field is used, the agency information inherited from the structural repository event container is ignored. |
| ID | IDQueryType | ID subscribes to the events of the specific instance of the object type where the value provided here matches the id of the object. The default value is the wildcard character("%"). |

**VersionableObjectEventType:** VersionableObjectEventType describes the structure of a reference to a versionable object's events. Either all instances of the object matching the inherited criteria, a specific instance, or specific instances of the object may be selected.

Content:

(All | URN | (ID, Version))

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| All | EmptyType | All subscribes to the events all instances of the containing object meeting the other criteria specified. |
| URN | xs:anyURI | URN subscribes to the events of the specific instance of the object type referenced by this URN. Note that when this field is used, the agency information inherited from the structural repository event container is ignored. |
| ID | IDQueryType | ID subscribes to the events of the specific instance of the object type where the value provided here matches the id of the object and the value provided in the version field matches the version of the object. The default value is the wildcard character("%"). |
| Version | VersionQueryType | Version subscribes to the events of the specific instance of the object type where the value provided in the id field matches the id of the object and the value here matches the version of the object. The default value is the wildcard character("%"). Note that in addition to being wild-carded, this can also be bound to the latest version of the object with the late-bound character("\*"). |

**DataRegistrationEventsType:** DataRegistrationEventsType details the data registration events for the subscription. It is possible to subscribe to all data registration events in the repository, or specific events for; single registrations, provision agreements, data providers, data flows, key families, and categories that categorize data flows or key families.

Attributes:

TYPE?

Content:

(AllEvents | (RegistrationID | ProvisionAgreement | DataProvider | DataflowReference | KeyFamilyReference | Category)+)

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| TYPE (fixed: DATA) | xs:string | TYPE is a fixed attribute that is used to ensure only of each event selector may be provided, when it is referenced in a uniqueness constraint. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| AllEvents | EmptyType | AllEvents subscribes to all data registration events in the repository. |
| RegistrationID | IDType | RegistrationID subscribes to all the data registration events for the unique registration referenced. |
| ProvisionAgreement | ProvisionAgreementRe ferenceType | ProvisionAgreementReference subscribes to all data registration events for the explicitly referenced provision agreement. |
| DataProvider | DataProviderReferenc eType | DataProviderReference subscribes to all data registration events for the explicitly referenced data provider. |
| DataflowReference | MaintainableEventTyp e | DataflowReference subscribes to all data registration events for the data flows referenced by this object. This may reference one or more data flows, as the specific references fields allow for a wild-carded value. |
| KeyFamilyReference | MaintainableEventTyp e | KeyFamilyReference subscribes to all data registration events for the key families referenced by this object. This may reference one or more key families, as the specific references fields allow for a wild-carded value. |
| Category | CategoryReferenceTyp e | Category subscribes to all data registration events for any data flows or key families that are categorized by the referenced category. |

**MetadataRegistrationEventsType:** MetadataRegistrationEventsType details the metadata registration events for the subscription. It is possible to subscribe to all metadata registration events in the repository, or specific events for; single registrations, provision agreements, data providers, metadata flows, metadata structure definitions, and categories that categorize metadata flows or metadata structure definitions.

Attributes:

TYPE?

Content:

(AllEvents | (RegistrationID | ProvisionAgreement | DataProvider | MetadataflowReference | MetadataStructureDefinitionReference | Category)+)

Attribute Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| TYPE (fixed: METADATA) | xs:string | TYPE is a fixed attribute that is used to ensure only of each event selector may be provided, when it is referenced in a uniqueness constraint. |

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| AllEvents | EmptyType | AllEvents subscribes to all metadata registration events in the repository. |
| RegistrationID | IDType | RegistrationID subscribes to all the metadata registration events for the unique registration referenced. |
| ProvisionAgreement | ProvisionAgreementRe ferenceType | ProvisionAgreementReference subscribes to all metadata registration events for the explicitly referenced provision agreement. |
| DataProvider | DataProviderReferenc eType | DataProvider subscribes to all metadata registration events for the explicitly referenced data provider. |
| MetadataflowReferenc e | MaintainableEventTyp e | MetadataflowReference subscribes to all metadata registration events for the metadata flows referenced by this object. This may reference one or more metadata flows, as the specific references fields allow for a wild-carded value. |
| MetadataStructureDef initionReference | MaintainableEventTyp e | MetadataStructureDefinitionReference subscribes to all metadata registration events for the metadata structure definitions referenced by this object. This may reference one or more metadata structure definitions, as the specific references fields allow for a wild-carded value. |
| Category | CategoryReferenceTyp e | Category subscribes to all metadata registration events for any metadata flows or metadata structure definitions that are categorized by the referenced category. |

**MaintainableEventType:** MaintainableEventType provides a reference to a maintainable object's event with either the URN of the specific object, or a set of potentially wild-carded reference fields.

Content:

(URN | Ref)

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| URN | xs:anyURI | URN provides an explicit reference to a single object. |
| Ref | MaintainableQueryTyp e | Ref provides a reference to potentially many object through the use of possible wild-carded reference fields. |

**StructuralEventType:** StructuralEventType provides the details of a structure event, specifically the object that changed.

Content:

Structures

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Structures | StructuresType | Structures contains the details of the structural object that has triggered the event. Although this container allows for multiple structural object, it should only contain the one changed object. |

**RegistrationEventType:** This provides the details of a data or metadata registration event for the purposes of notification.

Content:

Registration

Element Documentation:

| **Name** | **Type** | **Documentation** |
| --- | --- | --- |
| Registration | RegistrationType | Registration provides the changed details of the data or metadata registration. |

### Simple Types

**IDQueryType:** IDQueryType is a simple type that allows for an identifier to be substituted with a wild card character ("%").

Union of:

IDType, WildCardValueType.

**NestedIDQueryType:** NestedIDQueryType is a simple type that allows for a nested identifier to be substituted with a wild card character ("%").

Union of:

NestedIDType, WildCardValueType.

**VersionQueryType:** VersionQueryType is a simple type that allows for a version number to be substituted with a wild card character ("%") or a late bound character ("\*").

Union of:

LegacyVersionNumberType, SemanticVersionNumberType, SemanticVersionReferenceType, WildcardType, WildCardValueType.

**StatusType:** StatusType provides an enumeration of values that detail the status of queries or requests.

Derived by restriction of xs:NMTOKEN .

Enumerations:

|  |  |
| --- | --- |
| **Value** | **Documentation** |
| Success | Query or request successful. |
| Warning | Query or request successful, but with warnings. |
| Failure | Query or request not successful. |

**QueryTypeType:** QueryType provides an enumeration of values which specify the objects in the result-set for a registry query.

Derived by restriction of xs:NMTOKEN .

Enumerations:

|  |  |
| --- | --- |
| **Value** | **Documentation** |
| DataSets | Only references data sets should be returned. |
| MetadataSets | Only references to metadata sets should be returned. |
| AllSets | References to both data sets and metadata sets should be returned. |