Schema documentation for CDE_schema.xsd

october 17, 2018

Table of Contents

Schema(s)

Main schema `CDE_schema.xsd`

Element(s)

Element `data_element_set`
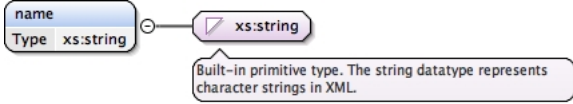
Diagram	
Model	id , name , description , (url | index_codes | modality | biological_sex | age_range | person | organization | (event*)) , elements
Instance	```<data_element_set> <id>{1,1}</id> <name>{1,1}</name> <description>{1,1}</description> <url>{1,1}</url> <index_codes>{1,1}</index_codes> <modality>{1,1}</modality> <biological_sex>{1,1}</biological_sex> <age_range>{1,1}</age_range> <person>{1,1}</person> <organization>{1,1}</organization> <event>{0,unbounded}</event> <elements>{1,1}</elements> </data_element_set>```

Element `data_element_set / id`

Annotations	Each set has a unique identifier of the format "RDES[0-9]+" (e.g., "RDES42") and a textual name.
Diagram	

Type	restriction of xs:string	
Facets	pattern	RDES\d+

Element `name`

Diagram	
	name Type xs:string ⊖— xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Elements data_element_set, element, organization, person, value_set/value

Element `description`

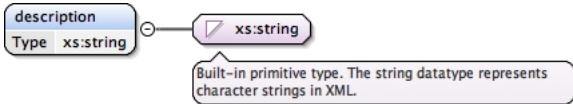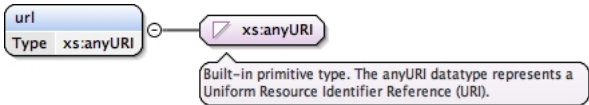| Diagram	
	description Type xs:string ⊖— xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element data_element_set
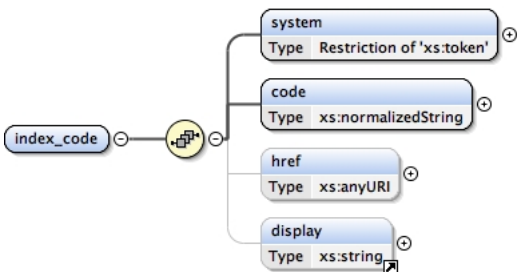
Element `data_element_set / url`

Diagram	
	url Type xs:anyURI ⊖— xs:anyURI Built-in primitive type. The anyURI datatype represents a Uniform Resource Identifier Reference (URI).
Type	xs:anyURI

Element `index_codes`

Diagram	
	index_codes ⊖— ⊕ ⊖— 1..∞ index_code ⊕
Used by	Elements data_element_set, element, value_set/value
Model	index_code+

Instance	`<index_codes>`
	  `<index_code>`{1,unbounded}`</index_code>`
	`</index_codes>`

Element `index_code`

Diagram	
	index_code ⊖— ⊕ ⊖— system Type Restriction of 'xs:token' ⊕ code Type xs:normalizedString ⊕ href Type xs:anyURI ⊕ display Type xs:string ⊕
Used by	Element index_codes
Model	system , code , href{0,1} , display{0,1}

Instance	`<index_code>`
	  `<system>`{1,1}`</system>`
	  `<code>`{1,1}`</code>`
	  `<href>`{0,1}`</href>`
	  `<display>`{0,1}`</display>`

```
</index_code>
```

Element `index_code / system`

Diagram	
	system Type Restriction of 'xs:token' — restricts: xs:token
Type	restriction of xs:token
Facets	enumeration RADLEX
	enumeration SNOMEDCT
	enumeration LOINC

Element `index_code / code`

Diagram	
	code Type xs:normalizedString — xs:normalizedString Built-in derived type. The normalizedString datatype represents white space normalized strings. The base type of...
Type	xs:normalizedString

Element `index_code / href`

Diagram	
	href Type xs:anyURI — xs:anyURI Built-in primitive type. The anyURI datatype represents a Uniform Resource Identifier Reference (URI).
Type	xs:anyURI

Element `display`

Diagram	
	display Type xs:string — xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element index_code

Element `modality`

Diagram	
	modality — system Type Restriction of 'xs:token'; code Type Restriction of 'xs:token'; href Type Restriction of 'xs:token'
Used by	Elements data_element_set, element
Model	system , code , href

Instance	`<modality>`
	`  <system>{1,1}</system>`
	`  <code>{1,1}</code>`
	`  <href>{1,1}</href>`
	`</modality>`

Element `modality / system`

Diagram	
	system Type Restriction of 'xs:token' — restricts: xs:token
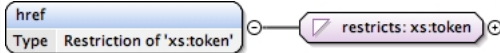
Type	restriction of xs:token	
Facets	enumeration	DICOM

Element `modality / code`

Diagram	
	code / Type Restriction of 'xs:token' — restricts: xs:token
Type	restriction of xs:token

Facets	enumeration	CR
	enumeration	CT
	enumeration	DX
	enumeration	IVUS
	enumeration	MG
	enumeration	MR
	enumeration	NM
	enumeration	PT
	enumeration	RF
	enumeration	RG
	enumeration	US
	enumeration	XA

Element `modality / href`

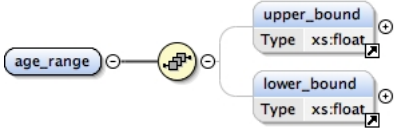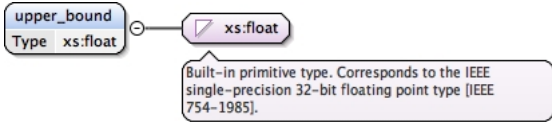| Diagram	
	href / Type Restriction of 'xs:token' — restricts: xs:token
Type	restriction of xs:token
Facets	enumeration	`http://dicom.nema.org/ medical/dicom/ current/output/html/ part03.html#sect_C.7.3.1.1.1`

Element `biological_sex`

Diagram	
	biological_sex / Type Restriction of 'xs:token' — restricts: xs:token
Type	restriction of xs:token
Facets	enumeration	M
	enumeration	F
Used by	Elements	data_element_set, element
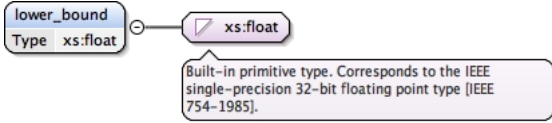
Element `age_range`

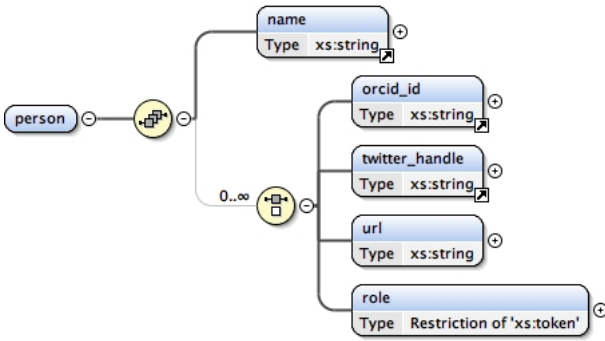| Diagram	
	age_range — upper_bound Type xs:float / lower_bound Type xs:float
Used by	Elements	data_element_set, element
Model	upper_bound{0,1} , lower_bound{0,1}

Instance	`<age_range>`
	`  <upper_bound>{0,1}</upper_bound>`
	`  <lower_bound>{0,1}</lower_bound>`
	`</age_range>`

Element `upper_bound`

Diagram	
	upper_bound / Type xs:float — xs:float — Built-in primitive type. Corresponds to the IEEE single-precision 32-bit floating point type [IEEE 754-1985].
Type	xs:float
Used by	Element age_range
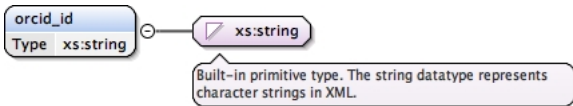
Element `lower_bound`

Diagram	
	lower_bound / Type xs:float — xs:float — Built-in primitive type. Corresponds to the IEEE single-precision 32-bit floating point type [IEEE 754-1985].
Type	xs:float
Used by	Element age_range

Element `person`

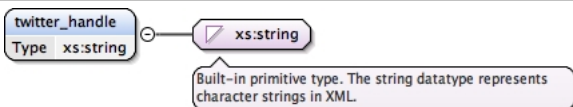| Diagram	
	person → name (Type xs:string); 0..∞ → (orcid_id, twitter_handle, url, role Type Restriction of 'xs:token')
Used by	Elements data_element_set, image Element Group authors
Model	name , (orcid_id | twitter_handle | url | role)

Instance	`<person>`
	  `<name>{1,1}</name>`
	  `<orcid_id>{1,1}</orcid_id>`
	  `<twitter_handle>{1,1}</twitter_handle>`
	  `<url>{1,1}</url>`
	  `<role>{1,1}</role>`
	`</person>`
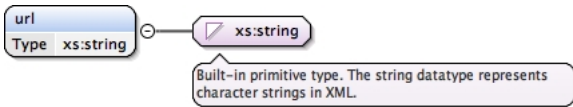
Element `orcid_id`

Diagram	
	orcid_id / Type xs:string — xs:string — Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element person
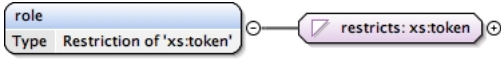
Element `twitter_handle`

Diagram	
	twitter_handle / Type xs:string — xs:string — Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element person
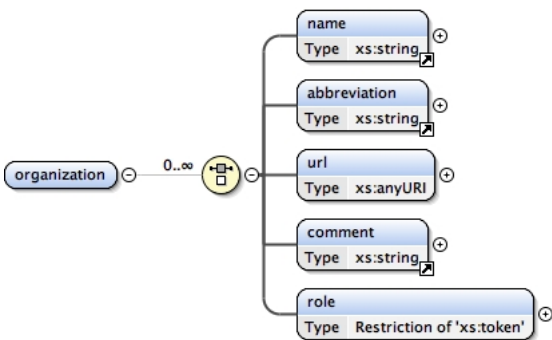
Element `person / url`

Diagram	
	url Type xs:string ⊖ — 📝 xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
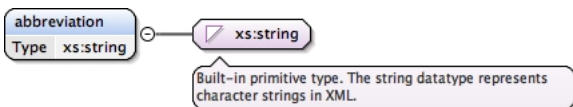
Element `person / role`

Diagram	
	role Type Restriction of 'xs:token' ⊖ — 📝 restricts: xs:token ⊕
Type	restriction of xs:token
Facets	enumeration `author`
	enumeration `editor`
	enumeration `translator`
	enumeration `reviewer`
	enumeration `contributor`
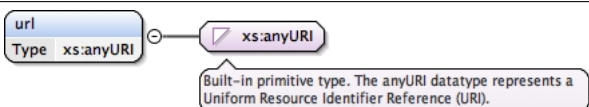
Element `organization`

Diagram	
	organization ⊖ — 0..∞ — ⊞ ⊖ name Type xs:string ⊕ / abbreviation Type xs:string ⊕ / url Type xs:anyURI ⊕ / comment Type xs:string ⊕ / role Type Restriction of 'xs:token' ⊕
Used by	Elements: data_element_set, image
	Element Group: authors
Model	name | abbreviation | url | comment | role

Instance	`<organization>`
	  `<name>{1,1}</name>`
	  `<abbreviation>{1,1}</abbreviation>`
	  `<url>{1,1}</url>`
	  `<comment>{1,1}</comment>`
	  `<role>{1,1}</role>`
	`</organization>`

Element `abbreviation`
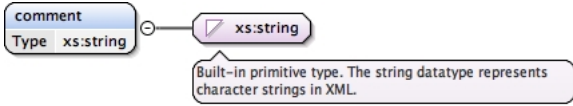
Diagram	
	abbreviation Type xs:string ⊖ — 📝 xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element: organization

Element `organization / url`

Diagram	
	url Type xs:anyURI ⊖ — 📝 xs:anyURI Built-in primitive type. The anyURI datatype represents a Uniform Resource Identifier Reference (URI).
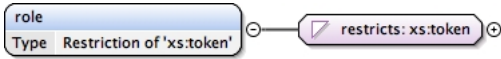
Type	xs:anyURI
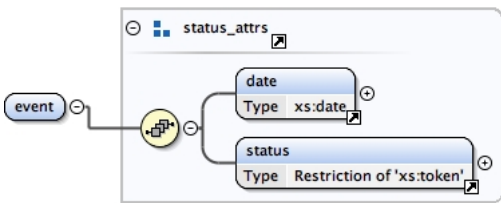
Element `comment`

Diagram	
Type	xs:string
Used by	Element organization
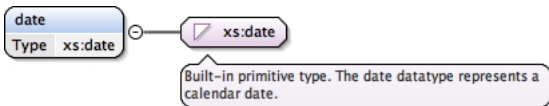
Element `organization / role`

Diagram	
Type	restriction of xs:token
Facets	enumeration contributor
	enumeration sponsor
	enumeration translator
	enumeration reviewer
	enumeration author
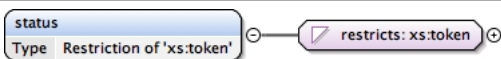
Element `event`

Diagram	
Used by	Element Group history
Model	date , status

Instance	`<event>`
	  `<date>{1,1}</date>`
	  `<status>{1,1}</status>`
	`</event>`

Element `date`
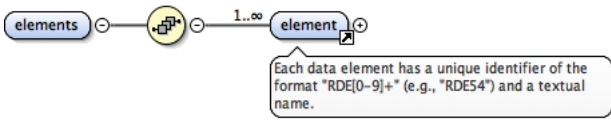
Diagram	
Type	xs:date
Used by	Element Group status_attrs

Element `status`
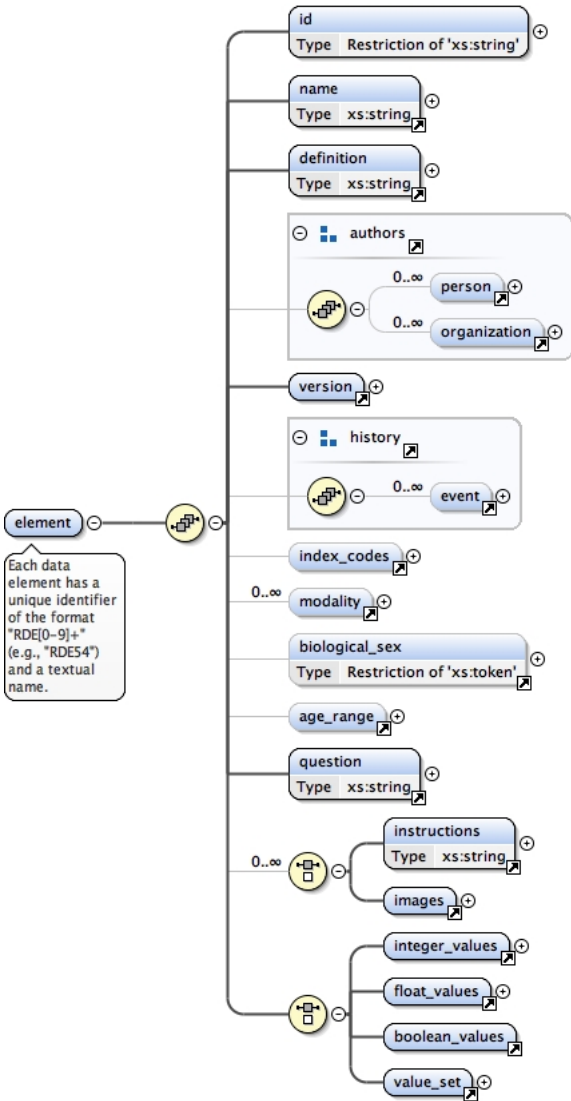
Diagram	
Type	restriction of xs:token
Facets	enumeration proposed
	enumeration approved
	enumeration published

	enumeration	deprecated
Used by	Element Group	status_attrs

Element `elements`

Diagram	
Used by	Element data_element_set
Model	element+

Instance	`<elements>`
	  `<element>`{1,unbounded}`</element>`
	`</elements>`

Element `element`

Annotations	Each data element has a unique identifier of the format "RDE[0-9]+" (e.g., "RDE54") and a textual name.
Diagram	
Used by	Element elements
Model	id , name , definition , person* , organization* , version , event* , index_codes{0,1} , modality* , biological_sex{0,1} , age_range{0,1} , question , (instructions | images) , (integer_values | float_values | boolean_values | value_set)

Instance	```xml
	<element>
	  <id>{1,1}</id>
	  <name>{1,1}</name>
	  <definition>{1,1}</definition>
	  <person>{0,unbounded}</person>
	  <organization>{0,unbounded}</organization>
	  <version>{1,1}</version>
	  <event>{0,unbounded}</event>
	  <index_codes>{0,1}</index_codes>
	  <modality>{0,unbounded}</modality>
	  <biological_sex>{0,1}</biological_sex>
	  <age_range>{0,1}</age_range>
	  <question>{1,1}</question>
	  <instructions>{1,1}</instructions>
	  <images>{1,1}</images>
	  <integer_values>{1,1}</integer_values>
	  <float_values>{1,1}</float_values>
	  <boolean_values>{1,1}</boolean_values>
	  <value_set>{1,1}</value_set>
	</element>
	``

Element `element / id`

Diagram	
Type	restriction of xs:string
Facets	pattern RDE\d+

Element `definition`

Diagram	
Type	xs:string
Used by	Elements element, value_set/value

Element `version`

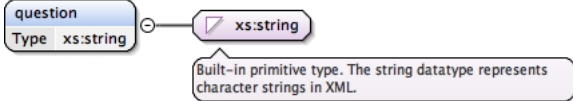| Diagram	
Used by	Element element
Model	version_number , date , status

Instance	```xml
	<version>
	  <version_number>{1,1}</version_number>
	  <date>{1,1}</date>
	  <status>{1,1}</status>
	</version>
	``

Element `version_number`
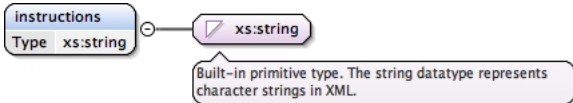
Diagram	

Type	xs:positiveInteger	
Used by	Element	version

Element `question`

Diagram	
Type	xs:string
Used by	Element element
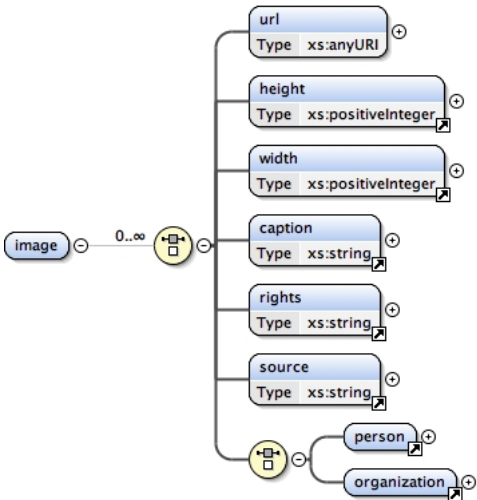
Element `instructions`

Diagram	
Type	xs:string
Used by	Element element

Element `images`

Diagram	
Used by	Elements element, value_set/value
Model	image+

Instance	`<images>`
	  `<image>{1,unbounded}</image>`
	`</images>`

Element `image`
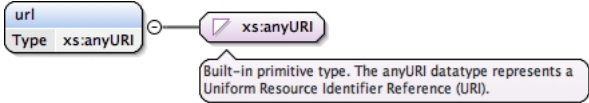
Diagram	
Used by	Element images
Model	url | height | width | caption | rights | source | person | organization

Instance	`<image>`
	  `<url>{1,1}</url>`
	  `<height>{1,1}</height>`
	  `<width>{1,1}</width>`
	  `<caption>{1,1}</caption>`
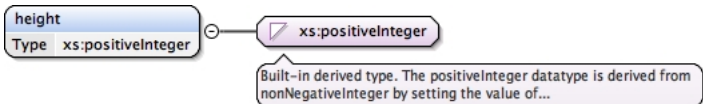
```
    <rights>{1,1}</rights>
    <source>{1,1}</source>
    <person>{1,1}</person>
    <organization>{1,1}</organization>
</image>
```

Element `image / url`

Diagram	
	url Type xs:anyURI ⊟— xs:anyURI Built-in primitive type. The anyURI datatype represents a Uniform Resource Identifier Reference (URI).
Type	xs:anyURI

Element `height`
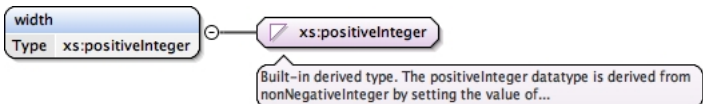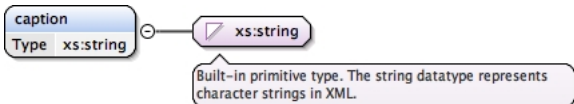
Diagram	
	height Type xs:positiveInteger ⊟— xs:positiveInteger Built-in derived type. The positiveInteger datatype is derived from nonNegativeInteger by setting the value of...
Type	xs:positiveInteger
Used by	Element image

Element `width`
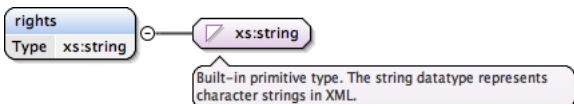
Diagram	
	width Type xs:positiveInteger ⊟— xs:positiveInteger Built-in derived type. The positiveInteger datatype is derived from nonNegativeInteger by setting the value of...
Type	xs:positiveInteger
Used by	Element image
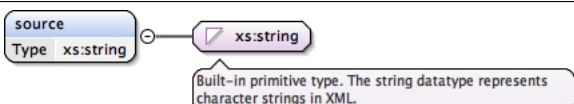
Element `caption`

Diagram	
	caption Type xs:string ⊟— xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element image

Element `rights`

Diagram	
	rights Type xs:string ⊟— xs:string Built-in primitive type. The string datatype represents character strings in XML.
Type	xs:string
Used by	Element image

Element `source`

Diagram	
	source Type xs:string ⊟— xs:string Built-in primitive type. The string datatype represents character strings in XML.

Type	xs:string	
Used by	Element	image

Element `integer_values`

Diagram	
Used by	Element element
Model	min{0,1} , max{0,1} , step{0,1} , unit{0,1}

Instance	```xml
	<integer_values>
	  <min>{0,1}</min>
	  <max>{0,1}</max>
	  <step>{0,1}</step>
	  <unit>{0,1}</unit>
	</integer_values>
	``

Element `integer_values / min`

Diagram	
Type	xs:integer

Element `integer_values / max`

Diagram	
Type	xs:integer

Element `step`

Diagram	
Type	xs:integer
Used by	Elements float_values, integer_values

Element `unit`

Diagram	
Type	xs:Name

Used by	Elements	float_values, integer_values

Element `float_values`

Diagram	
Used by	Element element
Model	min{0,1} , max{0,1} , step , unit

Instance	`<float_values>`
	`  <min>{0,1}</min>`
	`  <max>{0,1}</max>`
	`  <step>{1,1}</step>`
	`  <unit>{1,1}</unit>`
	`</float_values>`

Element `float_values / min`

Diagram	
Type	xs:float

Element `float_values / max`

Diagram	
Type	xs:float

Element `boolean_values`

Diagram	
Used by	Element element

Element `value_set`

Diagram	
Used by	Element element
Model	min_cardinality{0,1} , max_cardinality{0,1} , value+

Instance	`<value_set>`
	`  <min_cardinality>{0,1}</min_cardinality>`
	`  <max_cardinality>{0,1}</max_cardinality>`

```
    <value>{1,unbounded}</value>
</value_set>
```

Element `min_cardinality`

Diagram	
Type	xs:positiveInteger
Used by	Element value_set

Element `max_cardinality`

Diagram	
Type	xs:positiveInteger
Used by	Element value_set

Element `value_set / value`

Diagram	
Model	value , name , definition{0,1} , images{0,1} , index_codes{0,1}

Instance	```<value>```
	``` <value>{1,1}</value>```
	``` <name>{1,1}</name>```
	``` <definition>{0,1}</definition>```
	``` <images>{0,1}</images>```
	``` <index_codes>{0,1}</index_codes>```
	```</value>```

Element `value_set / value / value`
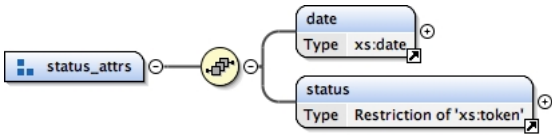
Diagram	
Type	xs:Name

Element Group(s)

Element Group `history`

Diagram	
Used by	Elements data_element_set, element
Model	event*

Element Group `status_attrs`

Diagram	
Used by	Elements event, version
Model	date , status

Element Group `authors`

Diagram	
Used by	Element element
Model	person* , organization*