

# Schema documentation for service-definition-1.1.xsd

Publication date 7 august 2012

## Table of Contents

Namespace: ""	1
Schemas	1
Main schema service-definition-1.1.xsd	1
Elements	1
Element services	1
Element service	2
Element implements	4
Element in-parameters	4
Element auto-parameters	4
Element exclude	5
Element parameter	5
Element subtype	6
Element ParameterValidations	7
Element out-parameters	7
Element val-or	7
Element val-and	8
Element val-not	8
Element matches	8
Element number-range	8
Element number-integer	9
Element number-decimal	9
Element text-length	9
Element text-email	9
Element text-url	10
Element text-letters	10
Element text-digits	10
Element time-range	10
Element credit-card	10

## Namespace: ""


### Schemas

#### Main schema service-definition-1.1.xsd

Namespace	No namespace
Properties	attribute form default: unqualified
	element form default: qualified

### Elements

#### Element services

Namespace	No namespace
Diagram	
Properties	content: complex
Model	service+
Children	service
Instance	<pre>&lt;services&gt;   &lt;service allow- remote="false" authenticate="true" location="" method="" noun="" semaphore="none" semaphore- ignore="3600" semaphore-sleep="5" semaphore-timeout="120" transaction="use-or- begin" transaction-timeout="" type="inline" validate="true" verb=""&gt;[1,unbounded]&lt;/ service&gt;</pre>

</services>

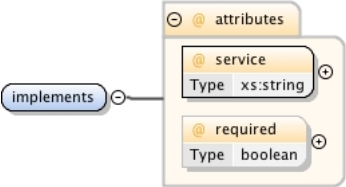
## Element service

Namespace	No namespace
Diagram	<p>The diagram illustrates the structure of the <code>service</code> element. It is a container for various attributes and children. The attributes are listed in a vertical stack, each with its type and default value. The children are listed in a horizontal stack, each with its type and cardinality. The <code>service</code> element is represented by a blue box with a yellow circle containing a plus sign. The attributes are represented by yellow boxes with a yellow circle containing a plus sign. The children are represented by blue boxes with a yellow circle containing a plus sign.</p> <ul style="list-style-type: none"> <li><b>Attributes:</b> <ul style="list-style-type: none"> <li><code>verb</code>: Type <code>xs:string</code></li> <li><code>noun</code>: Type <code>xs:string</code></li> <li><code>type</code>: Type <code>restriction of 'xs.token'</code>, Default <code>inline</code></li> <li><code>location</code>: Type <code>xs:string</code></li> <li><code>method</code>: Type <code>xs:string</code></li> <li><code>authenticate</code>: Type <code>restriction of 'xs.token'</code>, Default <code>true</code></li> <li><code>allow-remote</code>: Type <code>boolean</code>, Default <code>false</code></li> <li><code>validate</code>: Type <code>boolean</code>, Default <code>true</code></li> <li><code>transaction</code>: Type <code>restriction of 'xs.token'</code>, Default <code>use-or-begin</code></li> <li><code>transaction-timeout</code>: Type <code>xs:int</code></li> <li><code>semaphore</code>: Type <code>restriction of 'xs.token'</code>, Default <code>none</code></li> <li><code>semaphore-timeout</code>: Type <code>xs:int</code>, Default <code>120</code></li> <li><code>semaphore-sleep</code>: Type <code>xs:int</code>, Default <code>5</code></li> <li><code>semaphore-ignore</code>: Type <code>xs:int</code>, Default <code>3600</code></li> </ul> </li> <li><b>Children:</b> <ul style="list-style-type: none"> <li><code>description</code>: Type <code>xs:string</code></li> <li><code>implements</code>: 0..∞</li> <li><code>in-parameters</code>: 0..1</li> <li><code>out-parameters</code>: 0..1</li> <li><code>actions</code>: 0..1</li> </ul> </li> </ul>
Properties	content: complex
Used by	Element <code>services</code>
Model	<code>description{0,1}</code> , <code>implements*</code> , <code>in-parameters{0,1}</code> , <code>out-parameters{0,1}</code> , <code>actions{0,1}</code>
Children	<code>actions</code> , <code>description</code> , <code>implements</code> , <code>in-parameters</code> , <code>out-parameters</code>

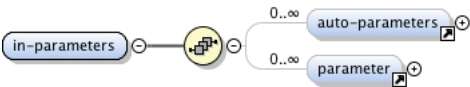
Instance	<pre> &lt;service allow- remote="false" authenticate="true" location="" method="" noun="" semaphore="none" semaphore- ignore="3600" semaphore-sleep="5" semaphore-timeout="120" transaction="use-or- begin" transaction-timeout="" type="inline" validate="true" verb=""&gt;   &lt;description&gt;{0,1}&lt;/description&gt;   &lt;implements required="" service=""&gt;{0,unbounded}&lt;/implements&gt;   &lt;in-parameters&gt;{0,1}&lt;/in-parameters&gt;   &lt;out-parameters&gt;{0,1}&lt;/out-parameters&gt;   &lt;actions&gt;{0,1}&lt;/actions&gt; &lt;/service&gt; </pre>				
Attributes	QName	Type	Fixed	Default	Use
	allow-remote	boolean		false	optional
	authenticate	restriction of xs:token		true	optional
	<p>If not set to false (is true by default) a user much be logged in to run this service. If the service is running in an ExecutionContext with a user logged in that will qualify. If not then either a "authUserAccount" parameter or the "authUsername" AND "authPassword" parameters must be specified and must contain valid values for a user of the system.</p> <p>If the "authUserAccount" parameter or the "authUsername" AND "authPassword" parameters are passed in they will be used for the service call even if a user is logged in to the ExecutionContext that the service is running in.</p> <p>An "authTenantId" parameter may also be specified to authenticate the user in a specific tenant instance. If specified will be used to run the service with that as the context tenant.</p> <p>If set to anonymous-all or anonymous-view not only with authentication not be required, but this service will run as if authorized (using the _NA_ UserAccount) for all actions or for view-only.</p>				
	location	xs:string			optional
	method	xs:string			optional
	noun	xs:string			optional
	<p>For entity-auto services this should be a valid entity name. In many other cases an entity name is the best way to describe what is being acted on, but this can really be anything.</p>				
	semaphore	restriction of xs:token		none	optional
	<p>Intended for use in long-running services (usually scheduled). This uses a record in the database to "lock" the service so that only one instance of it can run against a given database at any given time.</p>				
	semaphore-ignore	xs:int		3600	optional
	<p>Ignore existing semaphores after this time, in seconds.</p>				
	semaphore-sleep	xs:int		5	optional
	<p>When waiting how long to sleep between checking the semaphore, in seconds.</p>				
	semaphore-timeout	xs:int		120	optional
	<p>When waiting how long before timing out, in seconds.</p>				
	transaction	restriction of xs:token		use-or-begin	optional
	transaction-timeout	xs:int			optional
	<p>Defines the timeout for the transaction, in seconds. This value is only used if this service begins a transaction (either require-new, or use-or-begin and there is no other transaction already in place).</p>				
	type	restriction of xs:token		inline	optional
	validate	boolean		true	optional
	verb	xs:string			required
	<p>This can be any verb, and will often be one of: create, update, delete, or find. The full name of</p>				

	QName	Type	Fixed	Default	Use
		the service will be: "\${verb}\${noun}". The verb is required and the noun is optional so if there is no noun the service name will be just the verb.			

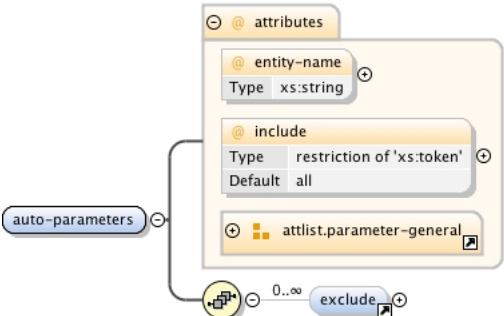
## Element implements

Namespace	No namespace				
Diagram					
Properties	content:	complex			
Used by	Element	service			
Attributes	QName	Type	Fixed	Default	Use
	required	boolean			optional
	If set to true or false all parameters inherited have that value for the required attribute.				
Attributes	service	xs:string			required

## Element in-parameters

Namespace	No namespace
Diagram	
Properties	content: complex
Used by	Element service
Model	auto-parameters* , parameter*
Children	auto-parameters, parameter
Instance	<pre>&lt;in-parameters&gt;   &lt;auto-parameters allow-html="none" entity- name=" " include="all" required="false"&gt;{0,unbounded}&lt;/auto-parameters&gt;   &lt;parameter allow-html="none" default- value=" " format=" " name=" " required="false" type="String"&gt;{0,unbounded}&lt;/parameter&gt; &lt;/in-parameters&gt;</pre>

## Element auto-parameters

Namespace	No namespace	
Diagram		
Properties	content:	complex
Used by	Elements	in-parameters, out-parameters
Model	exclude*	

Children	exclude				
Instance	<pre>&lt;auto-parameters allow-html="none" entity-name="" include="all" required="false"&gt;   &lt;exclude field-name=""&gt;{0,unbounded}&lt;/exclude&gt; &lt;/auto-parameters&gt;</pre>				
Attributes	<b>QName</b>	<b>Type</b>	<b>Fixed</b>	<b>Default</b>	<b>Use</b>
	<b>allow-html</b>	restriction of xs:token		none	optional
		<p>Applies only to String fields. Only checked for incoming parameters (meant for validating input from users, other systems, etc). Defaults to "none" meaning no HTML is allowed (will result in an error message).</p> <p>If some HTML is desired then use "safe" which will follow the rules in the antisamy-esapi.xml file. This should be safe for both internal and public users.</p> <p>In rare cases when users are trusted or it is not a sensitive field the "any" option may be used to not check the HTML content at all.</p>			
	<b>entity-name</b>	xs:string			optional
	<b>include</b>	restriction of xs:token		all	optional
	<b>required</b>	restriction of xs:token		false	optional

## Element exclude

Namespace	No namespace				
Diagram					
Properties	content:	complex			
Used by	Element	auto-parameters			
Attributes	<b>QName</b>	<b>Type</b>	<b>Fixed</b>	<b>Default</b>	<b>Use</b>
	<b>field-name</b>	xs:string			required

## Element parameter

Namespace	No namespace				
Diagram					

Properties	content: complex				
Used by	Elements in-parameters, out-parameters, parameter				
Model	description{0,1} , subtype* , parameter* , ParameterValidations*				
Children	ParameterValidations, description, parameter, subtype				
Instance	<pre>&lt;parameter allow-html="none" default-value="" format="" name="" required="false" type="String"&gt;   &lt;description&gt;{0,1}&lt;/description&gt;   &lt;subtype key="" type=""&gt;{0,unbounded}&lt;/subtype&gt;   &lt;parameter allow-html="none" default-value="" format="" name="" required="false" type="String"&gt;{0,unbounded}&lt;/parameter&gt;   &lt;ParameterValidations&gt;{0,unbounded}&lt;/ParameterValidations&gt; &lt;/parameter&gt;</pre>				
Attributes	QName	Type	Fixed	Default	Use
	allow-html	restriction of xs:token		none	optional
		Applies only to String fields. Only checked for incoming parameters (meant for validating input from users, other systems, etc). Defaults to "none" meaning no HTML is allowed (will result in an error message).			
		If some HTML is desired then use "safe" which will follow the rules in the antisamy-esapi.xml file. This should be safe for both internal and public users.			
		In rare cases when users are trusted or it is not a sensitive field the "any" option may be used to not check the HTML content at all.			
	default-value	xs:string			optional
		The value specified will be used for the parameter if no value is passed in (only used if required=false).			
	format	xs:string			optional
		Used only when the parameter is passed in as a String but the type is something other than String to convert to that type.			
		For date/time uses standard Java format strings described here: <a href="http://download.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html">http://download.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html</a>			
	name	xs:string			required
	required	restriction of xs:token		false	optional
type	xs:string		String	optional	

## Element subtype

Namespace	No namespace
Annotations	<p>The valid type if the parameter.type or the parent subtype is a Map or any type of collection.</p> <p>If the parent type is a Map then the key should be a String and the value should be the type of the subtype where the key attribute matches the Map key, or for all other Map keys not specified the subtype with no key specified.</p>
Diagram	
Properties	content: complex
Used by	Elements parameter, subtype
Model	subtype*

Children	subtype				
Instance	<pre>&lt;subtype key="" type=""&gt;   &lt;subtype key="" type=""&gt;{0,unbounded}&lt;/subtype&gt; &lt;/subtype&gt;</pre>				
Attributes	QName	Type	Fixed	Default	Use
	key	xs:string			optional
	type	xs:string			required

## Element ParameterValidations


Namespace	No namespace	
Diagram		
Properties	abstract:	true
Used by	Elements	parameter, val-and, val-not, val-or

## Element out-parameters

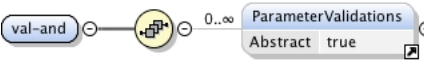
Namespace	No namespace	
Diagram		
Properties	content:	complex
Used by	Element	service
Model	auto-parameters*, parameter*	
Children	auto-parameters, parameter	
Instance	<pre>&lt;out-parameters&gt;   &lt;auto-parameters allow-html="none" entity- name="" include="all" required="false"&gt;{0,unbounded}&lt;/auto-parameters&gt;   &lt;parameter allow-html="none" default- value="" format="" name="" required="false" type="String"&gt;{0,unbounded}&lt;/parameter&gt; &lt;/out-parameters&gt;</pre>	

## Element val-or


Namespace	No namespace
-----------	--------------

Diagram	 The diagram shows a 'val-or' element connected to a 'ParameterValidations' element. The 'ParameterValidations' element has attributes 'Abstract' and 'true', and a cardinality of '0..∞'.
Properties	content: complex
Model	ParameterValidations*
Children	ParameterValidations
Instance	<pre>&lt;val-or&gt;   &lt;ParameterValidations&gt;{0,unbounded}&lt;/ParameterValidations&gt; &lt;/val-or&gt;</pre>

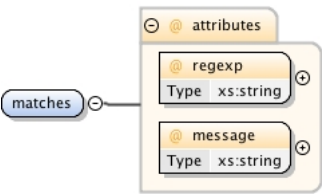
## Element val-and

Namespace	No namespace
Diagram	 The diagram shows a 'val-and' element connected to a 'ParameterValidations' element. The 'ParameterValidations' element has attributes 'Abstract' and 'true', and a cardinality of '0..∞'.
Properties	content: complex
Model	ParameterValidations*
Children	ParameterValidations
Instance	<pre>&lt;val-and&gt;   &lt;ParameterValidations&gt;{0,unbounded}&lt;/ParameterValidations&gt; &lt;/val-and&gt;</pre>

## Element val-not

Namespace	No namespace
Diagram	 The diagram shows a 'val-not' element connected to a 'ParameterValidations' element. The 'ParameterValidations' element has attributes 'Abstract' and 'true', and a cardinality of '0..∞'.
Properties	content: complex
Model	ParameterValidations*
Children	ParameterValidations
Instance	<pre>&lt;val-not&gt;   &lt;ParameterValidations&gt;{0,unbounded}&lt;/ParameterValidations&gt; &lt;/val-not&gt;</pre>

## Element matches

Namespace	No namespace				
Annotations	Validate the current parameter against the regular expression specified in the regexp attribute.				
Diagram	 The diagram shows a 'matches' element connected to a complex structure. This structure has an 'attributes' group containing a 'regexp' attribute (Type xs:string) and a 'message' attribute (Type xs:string).				
Properties	content: complex				
Attributes	<b>QName</b>	<b>Type</b>	<b>Fixed</b>	<b>Default</b>	<b>Use</b>
	message	xs:string			required
	regexp	xs:string			required

## Element number-range

Namespace	No namespace
-----------	--------------



Diagram					
Properties	content:	complex			
Attributes	<b>QName</b>	<b>Type</b>	<b>Fixed</b>	<b>Default</b>	<b>Use</b>
	<b>max</b>	xs:decimal			optional
	To pass number must be less than this value.				
	<b>max-include-equals</b>	boolean		false	optional
	Should the range include equal to the max number? Defaults to false (exclusive).				
	<b>min</b>	xs:decimal			optional
	To pass number must be greater than or equal to this value.				
	<b>min-include-equals</b>	boolean		true	optional
	Should the range include equal to the min number? Defaults to true.				

## Element number-integer

Namespace	No namespace
Diagram	

## Element number-decimal

Namespace	No namespace
Diagram	


## Element text-length

Namespace	No namespace				
Diagram					
Properties	content: complex				
Attributes	QName	Type	Fixed	Default	Use
	max	xs:nonNegativeInteger			optional
	min	xs:nonNegativeInteger			optional


## Element text-email

Namespace	No namespace
Diagram	

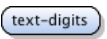
## Element text-url

Namespace	No namespace
Diagram	

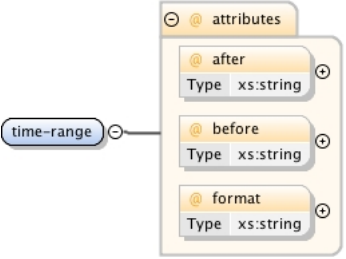
## Element text-letters

Namespace	No namespace
Diagram	

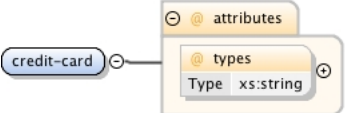
## Element text-digits

Namespace	No namespace
Diagram	

## Element time-range

Namespace	No namespace				
Diagram					
Properties	content:	complex			
Attributes	QName	Type	Fixed	Default	Use
	after	xs:string			optional
		Can be date/time string, or "now" to compare to the current time.			
	before	xs:string			optional
		Can be date/time string, or "now" to compare to the current time.			
	format	xs:string			optional
		If the value is a String instead of Date/Time/Timestamp, specify the format for conversion here.			

## Element credit-card

Namespace	No namespace				
Diagram					
Properties	content:	complex			
Attributes	QName	Type	Fixed	Default	Use
	types	xs:string			optional
		A comma-separate list of the types of credit card to allow. The available options include: visa, mastercard, amex, discover, enroute, jcb, solo, switch, dinersclub, visaelectron  If empty defaults to allow any type of card (ie doesn't check the card type, just checks the number using the Luhn MOD-10 checksum).			