# Schema documentation for config.xsd

june 19, 2011

# **Table of Contents**

Namespace: "	"	l
Schem	a(s)	1
	Main schema config.xsd	1
Elemen	nt(s)	1
	Element jmxpoller	1
	Element formatter	2
	Element cluster	2
	Element mbean	3
	Element attribute	4
	Element jmxserver	5
Attribu	ate(s)	7
	Attribute formatter / @className	7
	Attribute attribute / @name	7
	Attribute attribute / @outputname	-
	Attribute mbean / @domain	1
	Attribute mbean / @properties	8
	Attribute jmxserver / @host	8
	Attribute jmxserver / @jmxpass	8
	Attribute jmxserver / @jmxport	8
	Attribute jmxserver / @jmxuser	9
	Attribute jmxserver / @jvmDescription	9
	Attribute jmxserver / @pid	9
	Attribute jmxserver / @pidFile	9
	Attribute jmxserver / @pidCommand	1(
	Attribute cluster / @name	1(
	Attribute cluster / @description	1(

## Namespace: ""

## Schema(s)

#### Main schema config.xsd

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

## Element(s)

#### Element jmxpoller

Namespace	No namespace
Annotations	Root element of the configuration file. This configuration file is where you specify local and remote JMX servers to connect to across your enterprise and extract whatever MBean attributes you have declared to query. The result will then be written to STDOUT for SPLUNK indexing.
Diagram	formatter → O Cluster → O innxserver →
Properties	content: complex
Model	formatter{0,1}, cluster*, jmxserver*

```
Children
                             cluster, formatter, jmxserver
Instance
                             <jmxpoller>
                               className="">{0,1}</formatter>
<formatter className="">{0,undounded}</cluster>
<cluster description="" name="">{0,undounded}</cluster>
<jmxserver host="" jmxpass="" jmxport="" jmxuser="" jvmDescription="" pid="" pidCommand="" pidFile="">
                             jmxserver>
                             </jmxpoller>
                             <xs:element name="jmxpoller">
Source
                               <xs:annotation>
                                  <xs:documentation>Root element of the configuration file. This configuration file is
                              where you specify local and remote JMX servers to connect to across your enterprise and
                              extract whatever MBean attributes you have declared to query. The result will then be
                              written to STDOUT for SPLUNK indexing.</xs:documentation>
                               </xs:annotation>
                               <xs:complexType>
                                  <xs:sequence>
                                    <xs:element minOccurs="0" maxOccurs="1" ref="formatter"/>
                                    <xs:element minOccurs="0" maxOccurs="unbounded" ref="cluster"/>
<xs:element minOccurs="0" maxOccurs="unbounded" ref="jmxserver"/>
                                  </xs:sequence>
                               </xs:complexType>
                             </xs:element>
```

#### **Element** formatter

Namespace	No namespace	No namespace					
Annotations		Custom formatter declaration allows you to override the default STDOUT format					
Diagram	(formatter) ⊙——	@ className Type xs:string					
Properties	content:	complex					
Used by	Element	jmxpoller					
Attributes	QName	Type	Fixed	Default	Use		
	className	xs:string			required		
		Fully qualified Java class name of the formatter implementation, implements the com.dtdsoftware.splunk.formatter.Formatter interface					
Source	STDOUT format< //xs:annotati <xs:complexty <xs:annot="" <xs:annot<="" <xs:attribu="" td=""><td>n&gt; .tation&gt;Custom fo /xs:documentatio on&gt; pe&gt; .te name="classNa ation&gt; .umentation&gt;Fully .com.dtdsoftware .tation&gt; .ute&gt;</td><td>n&gt; me" type="xs:stri qualified Java c</td><td></td><td></td></xs:complexty>	n> .tation>Custom fo /xs:documentatio on> pe> .te name="classNa ation> .umentation>Fully .com.dtdsoftware .tation> .ute>	n> me" type="xs:stri qualified Java c				

#### Element cluster

Namespace	No namespace
Annotations	For JVMs with the same MBeans, you can group them under this element so you only have to declare the common beans to query once. You can still declare additional mbeans specfic to each jmxserver within the jmxserver elements.

	Ту	e attributes  e name  pe xs:string  description  pe xs:string  1  mbean  1  image: ima						
Properties	content:	content: complex						
Used by	Element	Element jmxpoller						
Model	mbean+, jmxserve	r+						
Children	jmxserver, mbean							
Instance	<mbean domain<="" th=""><th colspan="6"></th></mbean>							
Attributes	QName	Type	Fixed	Default	Use			
	description	xs:string			optional			
		Description of this cluster						
	name	xs:string			optional			
		Name for this	cluster		,			
		<pre>Name for this cluster  <pre> <xs:element name="cluster"></xs:element></pre></pre>						

#### Element mbean

Namespace	No namespace
Annotations	An MBean to query Standard JMX object name wildcard patterns * and ? are supported If no values are specified for the "domain" and "properties" attributes , the value will default to the * wildcard

Diagram	Typ	domain  e xs:string  properties  e xs.string  1  attribute					
Properties	content:	complex					
Used by	Elements	cluster, jmxserv	er				
Model	attribute+						
Children	attribute						
Instance		" properties=""> me="" outputname	e="">{1,unbounded}				
Attributes	QName	Type	Fixed	Default	Use		
	domain	xs:string			required		
		The MBean do	main				
	properties	xs:string			required		
		The MBean properties string in  "key=value,key2=value2" format					
Source	are supported the value will <td>n&gt; tation&gt;An MBean If no values are default to the on&gt; pe&gt; e&gt; nt maxOccurs="ur ce&gt; te name="domain" ation&gt; umentation&gt;The M tation&gt; ute&gt; te name="propert ation&gt; umentation&gt;The M tation&gt; ute&gt; tation&gt; umentation&gt;The M tation&gt; ute&gt;</td> <td>e specified for the * wildcard<td><pre>e "domain" and "pro cumentation&gt;  ribute"/&gt; ype="xs:string"&gt; ocumentation&gt;  d" type="xs:string"</pre></td><td>.ldcard patterns * and ? operties" attributes ,  '&gt; e,key2=value2" format<!--</td--></td></td>	n> tation>An MBean If no values are default to the on> pe> e> nt maxOccurs="ur ce> te name="domain" ation> umentation>The M tation> ute> te name="propert ation> umentation>The M tation> ute> tation> umentation>The M tation> ute>	e specified for the * wildcard <td><pre>e "domain" and "pro cumentation&gt;  ribute"/&gt; ype="xs:string"&gt; ocumentation&gt;  d" type="xs:string"</pre></td> <td>.ldcard patterns * and ? operties" attributes ,  '&gt; e,key2=value2" format<!--</td--></td>	<pre>e "domain" and "pro cumentation&gt;  ribute"/&gt; ype="xs:string"&gt; ocumentation&gt;  d" type="xs:string"</pre>	.ldcard patterns * and ? operties" attributes ,  '> e,key2=value2" format </td		

## **Element** attribute

Namespace	No namespace				
Annotations	An MBean attribute				
Diagram	attribute 🗇 Type	ributes  me  xs:string  tputname  xs:string			
Properties	content:	complex			
Used by	Element	mbean			
Attributes	QName	Type	Fixed	Default	Use
	name	xs:string			required
		The attribute name For attributes that			

	QName	Туре	Fixed	Default	Use
		level ie: composi attributes , then delimited notation specifying the at ie: foo:goo:myattribu	you can use a ":" n for tribute name.		
	outputname	xs:string			required
		The attribute key SPLUNK indexing	that is output to	STDOUT for	
Source	<pre><td>n&gt; tation&gt;An MBean attri on&gt; pe&gt; te name="name" use="r ation&gt; umentation&gt;The attrib tabular attributes , attribute name. ie: tation&gt; ute&gt; te name="outputname" ation&gt; umentation&gt;The attrib tation&gt; ute&gt; te name="outputname" ation&gt; umentation&gt;The attrib n&gt; tation&gt; ute&gt;</td><td>required" type="xs:  bute name For attri then you can use a foo:goo:myattribut  use="required" typ</td><td>string"&gt; butes that are mu. ":" delimited not e e="xs:string"&gt;</td><td>tation for</td></pre>	n> tation>An MBean attri on> pe> te name="name" use="r ation> umentation>The attrib tabular attributes , attribute name. ie: tation> ute> te name="outputname" ation> umentation>The attrib tation> ute> te name="outputname" ation> umentation>The attrib n> tation> ute>	required" type="xs:  bute name For attri then you can use a foo:goo:myattribut  use="required" typ	string"> butes that are mu. ":" delimited not e e="xs:string">	tation for

#### Element jmxserver

Namespace	No namespace
Annotations	A local or remote JMX Server to connect to
Diagram	o attributes  o host Type xs:string  imxport Type xs:string  imxuser Type xs:string  imvalues  o imxport Type xs:string  o imxuser Type xs:string  o pid Type xs:string  o pidFile Type xs:string  o pidCommand Type xs:string
Properties	content: complex
Used by	Elements cluster, jmxpoller
Model	mbean*
Children	mbean
Instance	<pre><jmxserver host="" jmxpass="" jmxport="" jmxuser="" jvmdescription="" p<="" pid="" pidcommand="" td=""></jmxserver></pre>

QName	Type	Fixed	Default	Use
host	xs:string			optional
	IP Address, Hostr	ame or DNS Alias.	1	1
jmxpass	xs:string			optional
	JMX Password		I	
jmxport	xs:integer			optional
<u> </u>	JMX Port			•
jmxuser	xs:string			optional
3	JMX Username			-F
jymDescription	xs:string			optional
JVIIIDescription		this TIM		ориона
	A description of	LIIS JVM		
pid —	xs:integer  Process ID for at running JVM	taching directly t	a locally	optional
pidCommand	xs:string			optional
	_	at outputs to STDO y to a locally run		for
pidFile	xs:string			optional
	directly to a loc	he Process ID for ally running JVM.T first line of the	ne only file conte	nts should
<pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre></pre> <pre><pre><pre><pre><pre><pre><pre>&lt;</pre></pre></pre></pre></pre></pre></pre>	dation>A local or remons  despect to minOccurs="0" maxible  despect to mane="host" type= despect to mane="host" type= despect to mane="jmxpass" type despect to mane="jmxpass" type despect to mane="jmxpass" type despect to mane="jmxpass" type despect to mane="jmxport" type despect to maxible despec	Doccurs="unbounded"  "xs:string">  s, Hostname or DNS  Doce="xs:string">  Docd  "xs:documentation>  Doce="xs:string">  Doccurs="unbounded"  Doce="xs:string">  Doccurs="unbounded"  Doce="xs:string">  Doccurs="unbounded"  Doce="xs:string">  Doccurs="unbounded"  Doce="xs:string">  Doccurs="unbounded"  Doce="xs:string">  Doccurs="unbounded"  Docurs="unbounded"  DNS  Docurs="unbounded"  Docur	<pre>ref="mbean"/&gt; Alias.</pre>	

```
</xs:annotation>
  </xs:attribute>
  </xs:complexType>
  </xs:element>
```

# Attribute(s)

#### Attribute formatter / @className

Namespace	No namespace
Annotations	Fully qualified Java class name of the formatter implementation, implements the com.dtdsoftware.splunk.formatter.Formatter interface
Туре	xs:string
Properties	use: required
Used by	Element formatter
Source	<pre><xs:attribute name="className" type="xs:string" use="required"></xs:attribute></pre>

#### Attribute attribute / @name

Namespace	No namespace
Annotations	The attribute name  For attributes that are multi level ie: composite and tabular attributes , then you can use a ":" delimited notation for specifying the attribute name. ie: foo:goo:myattribute
Туре	xs:string
Properties	use: required
Used by	Element attribute
Source	<pre><xs:attribute name="name" type="xs:string" use="required">     <xs:annotation>     <xs:documentation>The attribute name For attributes that are multi level ie: composite and tabular attributes , then you can use a ":" delimited notation for specifying the attribute name. ie: foo:goo:myattribute<!--/xs:documentation-->     </xs:documentation></xs:annotation> </xs:attribute></pre>

#### Attribute attribute / @outputname

Namespace	No namespace
Annotations	The attribute key that is output to STDOUT for SPLUNK indexing
Туре	xs:string
Properties	use: required
Used by	Element attribute
Source	<pre><xs:attribute name="outputname" type="xs:string" use="required"></xs:attribute></pre>

#### Attribute mbean / @domain

Namespace	No namespace
Annotations	The MBean domain

Type	xs:string
Properties	use: required
Used by	Element mbean
Source	<pre><xs:attribute name="domain" type="xs:string" use="required">     <xs:annotation>         <xs:documentation>The MBean domain</xs:documentation>         </xs:annotation> </xs:attribute></pre>

#### Attribute mbean / @properties

Namespace	No namespace
Annotations	The MBean properties string in "key=value,key2=value2" format
Туре	xs:string
Properties	use: required
Used by	Element mbean
Source	<pre><xs:attribute name="properties" type="xs:string" use="required">     <xs:annotation>         <xs:documentation>The MBean properties string in "key=value,key2=value2" format<!-- xs:documentation-->         </xs:documentation></xs:annotation>         </xs:attribute></pre>

## Attribute jmxserver / @host

Namespace	No namespace
Annotations	IP Address, Hostname or DNS Alias.
Туре	xs:string
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="host" type="xs:string">     <xs:annotation>         <xs:documentation>IP Address, Hostname or DNS Alias.</xs:documentation>         </xs:annotation>         </xs:attribute></pre>

#### Attribute jmxserver / @jmxpass

Namespace	No namespace
Annotations	JMX Password
Туре	xs:string
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="jmxpass" type="xs:string">     <xs:annotation></xs:annotation></xs:attribute></pre>

## Attribute jmxserver / @jmxport

Namespace	No namespace
Annotations	JMX Port
Туре	xs:integer
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="jmxport" type="xs:integer"></xs:attribute></pre>

#### Attribute jmxserver / @jmxuser

Namespace	No namespace
Annotations	JMX Username
Туре	xs:string
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="jmxuser" type="xs:string">     <xs:annotation></xs:annotation></xs:attribute></pre>

#### Attribute jmxserver / @jvmDescription

Namespace	No namespace
Annotations	A description of this JVM
Туре	xs:string
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="jvmDescription" type="xs:string"></xs:attribute></pre>

## Attribute jmxserver / @pid

Namespace	No namespace
Annotations	Process ID for attaching directly to a locally running JVM
Туре	xs:integer
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="pid" type="xs:integer">     <xs:annotation>         <xs:documentation>Process ID for attaching directly to a locally running JVM<!-- xs:documentation-->         </xs:documentation></xs:annotation>         </xs:attribute></pre>

#### Attribute jmxserver / @pidFile

Namespace	No namespace
Annotations	File containing the Process ID for attaching directly to a locally running JVM. The only file contents should be the PID on the first line of the file.
Туре	xs:string
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="pidFile" type="xs:string">     <xs:annotation>         <xs:documentation>File containing the Process ID for attaching directly to a locally running JVM.The only file contents should be the PID on the first line of the file.<!-- xs:documentation-->         </xs:documentation></xs:annotation>         </xs:attribute></pre>

## Attribute jmxserver / @pidCommand

Namespace	No namespace
Annotations	Command/Script that outputs to STDOUT the Process ID for attaching directly to a locally running JVM
Туре	xs:string
Properties	content: simple
Used by	Element jmxserver
Source	<pre><xs:attribute name="pidCommand" type="xs:string">     <xs:annotation>     <xs:documentation>Command/Script that outputs to STDOUT the Process ID for attaching directly to a locally running JVM</xs:documentation>     </xs:annotation> </xs:attribute></pre>

#### Attribute cluster / @name

Namespace	No namespace
Annotations	Name for this cluster
Туре	xs:string
Properties	content: simple
Used by	Element cluster
Source	<pre><xs:attribute name="name" type="xs:string">     <xs:annotation>           <xs:documentation>Name for this cluster</xs:documentation>           </xs:annotation>           </xs:attribute></pre>

### Attribute cluster / @description

Namespace	No namespace
Annotations	Description of this cluster
Туре	xs:string
Properties	content: simple
Used by	Element cluster
Source	<pre><xs:attribute name="description" type="xs:string">     <xs:annotation>           <xs:documentation>Description of this cluster</xs:documentation>           </xs:annotation>           </xs:attribute></pre>