Analysising the essential classes/ interfaces.

Introduction:

The purpose of this piece is to try and take a detailed look at the classes that are involved in the Simple Object access protocol and how Apache axis uses these classes to carry out this task.

Every class and interface directly related to SOAP will be looked at, by giving a general presentation of its Methods, constructors and other data I hope to gain a greater understanding as to how axis implements SOAP so that I can in some way improve on this and mimic certain aspects in order to achieve the goal of creating a "mimimal soap library". This general presentation is then followed by an overview of the classes which will be my own interpretation of what I think the class achieves.

The Classes looked at are as follows:

SOAP11Constants
SOAPElement
SOAPElementFactory
SOAPEnvelope
SOAPEnvelope
SOAPException
SOAPFactory
SOAPFactoryImpl
SOAPFault
SOAPFault
SOAPFaultBuilder
SOAPFaultElement
SOAPFaultElement
SOAPFaultException
SOAPHandler
SOAPHeader
SOAPHeader
SOAPHeaderElement
SOAPHeaderElement
SOAPMessage
SOAPMessageContext
SOAPMonitorConstants
SOAPMonitorHandler
SOAPMonitorService
SOAPPart
SOAPPart
SOAPService
SOAPUtils

Class SOAP11Constants

public class SOAP11Constants extends java.lang.Object implements SOAPConstants

Fields inherited from interface org.apache.axis.soap.SOAPConstants
SOAP11_CONSTANTS, SOAP12_CONSTANTS

Constructor Summary
SOAP11Constants()

Method Summary

QName getBodyQName()
Obtain the QName for the Body element

java.lang.String getEncodingURI()
Obtain the encoding namespace for this version of SOAP

java.lang.String getEnvelopeURI()
Obtain the enevelope namespace for this version of SOAP

QName getFaultQName()
Obtain the QName for the Fault element

QName getHeaderQName()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait

Constructor Detail
SOAP11Constants

Overview of class SOAP11Constants:

This class appears to have several functions firstly the Qname for the body element is obtained. The encoding namespace for this version of SOAP is also obtained, as is the QName for the fault element and the envelope namespace for this version.

A part from this the HeaderQname is also obtained. The QName is known as the Class to represent a qualified name: " The name of an internal XSLT object, specifically a named template (), a mode (, an attribute set , a key , a locale , a variable or a parameter is specified as a QName.

Namespaces:
XML namespaces provide a simple method for qualifying element and attribute names used in Extensible Markup Language documents by associating them with namespaces identified by URI references.

Class SOAP12Constant

public class SOAP12Constants extends java.lang.Object implements SOAPConstants

Fields inherited from interface org.apache.axis.soap.SOAPConstants
SOAP11_CONSTANTS, SOAP12_CONSTANTS

Constructor Summary
SOAP12Constants( )

Method Summary

QName getBodyQName( )
Obtain the QName for the Body element

java.lang.String getEncodingURI( )
Obtain the encoding namespace for this version of SOAP

java.lang.String getEnvelopeURI( )
Obtain the envelope namespace for the version SOAP

QName getFaultQName( )
Obtain the Qname for the Fault element

QName getHeaderQName( )
Obtain the QName for the Header element

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait

Constructor Detail

SOAP12Constants

public SOAP12Constants( )

Over view of Class SOAP12Constants:

This class appears to be a slightly more advanced version of the previous class.

Interface SOAPBody

public interface SOAPBody extends SOAPElement

An object that represents the contents of the SOAP body element in a SOAP message. A SOAP body element consists of XML data that affects the way the application-specific content is processed.

A SOAPBody object contains SOAPBodyElement objects, which have the content for the SOAP body. A SOAPFault object, which carries status and/or error information, is an example of a SOAPBodyElement object.

Method Summary

SOAPBodyElement addBodyElement(Name name)
Creates a new SOAPBodyElement object with the specified name and adds it to this SOAPBody object.

SOAPFault addFault( )
Creates a new SOAPFault object and adds it to this SOAPBody object.

SOAPFault getFault( )
Returns the SOAPFault object in this SOAPBody object.

boolean hasFault( )
Indicates whether a SOAPFault object exists in this SOAPBody object

Methods inherited from interface javax.xml.soap.SOAPElement

addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node

detachNode, getParentElement, getValue, recycleNode, setParentElement

Overview of Interface SOAPBody:

After reading the intro and the explanations for the various methods I believe the interface to be self explanatory.

-

Class SOAPBody

public class SOAPBody extends MessageElement implements SOAPBody
Holder for body elements.

Inner classes inherited from class org.apache.axis.message.MessageElement
MessageElement.QNameAttr

Fields inherited from class org.apache.axis.message.MessageElement
_isDirty, _isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, log, message, name, namespaces, namespaceURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName

Constructor Summary
SOAPBody(java.lang.String namespace, java.lang.String localPart, java.lang.String qName, org.xml.sax.Attributes attributes, DeserializationContext context, SOAPConstants soapConsts)

Method Summary

SOAPBodyElement addBodyElement(Name name)
Creates a new SOAPBodyElement object with the specified name and adds it to this SOAPBody object.

SOAPFault addFault( )
Creates a new SOAPFault object and adds it to this SOAPBody object.

void detachNode( )
Removes this node object from the tree.

protected MessageElement findElement(java.util.Vector vec, java.lang.String namespace, java.lang.String localPart)

SOAPFault getFault( )
Retrns the SOAPFault object in this SOAPBody object.

boolean hasFault( )
Indicates whether a SOAPFault object exists in this SOAPBody object.

protected void outputImpl(serializationContext context)
Subclasses can override.

void setParentElement(SOAPElement parent)
Sets the parent of this Node object to the given SOAPElement object.

Methods inherited from class org.apache.axis.message.MessageElement
addAttribute, addAttribute, addAttribute, addChild, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue, getAttributeValue, getChildElements, getChildElements, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, output, publishContents, publishToHandler, recycleNode, removeAttribute, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName,

setNamespaceURI, setNSMappings, setObjectValue, setPrefix, setRecorder, setType, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait

Methods inherited from interface javax.xml.soap.SOAPElement

addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node

getParentElement, getValue, recycleNode

Constructor Detail

SOAPBody

```
public SOAPBody( java.lang.String namespace,
                 java.lang.String localPart,
                 java.lang.String qName,
                 org.xml.sax.Attributes attributes,
                 DeserializationContext context,
                 SOAPConstants soapConsts)
```

Overview of Class SOAPBody:

This interface is slightly more complicated then most as it seems to inherit alot of functionality from elsewhere. It's main function is to prepare the SOAPBody it creates and adds new body elements agin with a specific name, also adds a SOAPFault object to the body. A lot of methods are inherited from class MessageElement which can be used on the Element being added to the SOAP body.

Interface SOAPBodyElement

public interface SOAPBodyElement extends SOAPElement

A SOAPBodyElement object represents the contents in a SOAPBody object. The SOAPFault interface is a SOAPBodyElement object that has been defined.

A new SOAPBodyElement object can be created and added to a SOAPBody object with the SOAPBody method addBodyElement. In the following line of code, sb is a SOAPBody object, and myName is a Name object.

SOAPBodyElement sbe = sb.addBodyElement(myName) ;

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode, setParentElement

Overview of Interface BodyElement:
Self explanatory:

Class SOAPBodyElement

public class SOAPBodyElement extends MessageElement implements SOAPBodyElement

Inner classes inherited from class org.apache.axis.message.MessageElement
MessageElement.QNameAttr

Fields inherited from class org.apache.axis.message.MessageElement
_isDirty, _isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, log, message, name, namespaces, namespaceURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName

Constructor Summary
SOAPBodyElement( )

SOAPBodyElement(org.w3c.dom.Element elem)

SOAPBodyElement(java.io.InputStream input)

SOAPBodyElement(Name name)

SOAPBodyElement(java.lang.String namespace, java.lang.String localPart, java.lang.String prefix, org.xml.sax.Attributes attributes, Deserialization context)

Method Summary
void detachNode( )
Removes this Node object from the tree.

void setParentElement(SOAPElement parent)
Sets the parent of this Node object to the given SOAPElement object.

Methods inherited from class org.apache.axis.message.MessageElement
addAttribute, addAttribute, addAttribute, addChild, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue, getAttributeValue, getChildElements, getChildElements, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, output, outputImpl, publishContents, publishToHandler, recycleNode, removeAttribute, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName, setNamespaceURI, setNSMappings, setObjectValue, setPrefix, setRecorder, setType, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
getParentElement, getValue, recycleNode

Constructor Detail

SOAPBodyElement

public SOAPBodyElement(java.lang.String namespace,
java.lang.String localPart,
java.lang.String prefix,
org.xml.sax.Attributes attributes,
DeserializationContext context)

SOAPBodyElement
public SOAPBodyElement(Name name)

SOAPBodyElement
public SOAPBodyElement(org.w3c.dom.Element elem)

SOAPBodyElement
public SOAPBodyElement( )

SOAPBodyElement

public SOAPBodyElement(java.io.InputStream input)

Overview of Class SOAPBodyElement:

Every SOAPBody is comprised of several soap elements the task of this class is to call on a variety of inherited methods to create these elements.

Class SOAPConnection

public abstract class SOAPConnection extends java.lang.Object

A point-to-point connexion that a client can use for sending messages directly to a remote party(represented by a URL, for instance) .

A client can obtain a SOAPConnection object simply by calling the following static method.

SOAPConnection con = SOAPConnection.newInstance( ) ;

A SOAPConnection object can be used to send messages directly to a URL following the request/response paradigm. That is, messages are sent using the method call, which sends the message and then waits until it gets a reply.

Constructor Summary
SOAPConnection( )

Method Summary

abstract SOAPMessage	call(SOAPMessage request, java.lang.Object endpoint) Sends the given message to the specified endpoint and blocks until it has returned the response.
abstract void	close( ) Closes this SOAPConnection object.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait

Constructor Detail

SOAPConnection

public SOAPConnection( )

Overview of Class SOAPConnection:

This class again is fairly self explaining it's basic aim is to establish point-to-point connections it does this by sending messages and waiting for reply's.

Class SOAPConnectionFactory

public abstract class SOAPConnectionFactory extends java.lang.Object

Constructor Summary
SOAPConnectionFactory( )

Method Summary

abstract SOAPConnection	createConnection( ) Create a new SOAPConnection.
static SOAPConnectionFactory	newInstance( ) Creates an instance of the default SOAPConnectionFactory object

Methods inheritedfrom class java.lang.Object
clone, equals, finalize, getClass, hashcode, notfiy, notifyAll, toString

ConstructorDetail

SOAPConnectionFactory

public SOAPConnectionFactory( )

Overview of class SOAPConnectionFactory.

The SOAPConnectionFactory like other factory classes is responsible for creating these soapConnection objects as was outlined in the previous class.

Class SOAPConnectionFactoryImpl

public class SOAPConnectionFactoryImpl extends SOAPConnectionFactory SOAP Connection Factory implementation.

Constructor Summary
SOAPConnectionFactoryImpl( )

Method Summary

SOAPConnection	createConnection( ) Create a new SOAPConnection.

Methods inherited from class javax.xml.soap.SOAPConnectionFactory
newInstance

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait

Constructor Detail
SOAPConnectionFactoryImpl( )

Overview of Class SOAPConnectionFactoryImpl:
The exact differeneces between this class and SOAPConnectionFactory seem to be few, again it sprimary function would be in the preperation of SOAPConnection objects.

Class SOAPConnectionImpl

public class SOAPConnectionImpl extends SOAPConnection

SOAP connexion implmentation

Constructor Summary
SOAPConnectionImpl( )

Method Summary

SOAPMessage	call(SOAPMessage request, java.lang.object endpoint) Sends the given message to the specified endpoint and blocks until it has returned the response.
void	close( ) Closes this SOAPConnection object.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, nofify, notifyAll, toString, wait.

Constructor Detail

SOAPConnectionImpl

public SOAPConnectionImpl( )

Overview of Class:

This small class seems to actually call the SOAPConnection ito use, it's two methods start and stop the connexion being established.

Interface SOAPConstants

public interface SOAPConstants

Field Summary

static java.lang	URI_NS_SOAP_ENCODING The namespace identifier for the SOAP encoding
static java.lang.String	URI_NS_SOAP_ENVELOPE The namespace identifier for the SOAP envelope.
static java.lang.string	URI_SOAP_ACTOR_NEXT The URI identifying the first application processing a SOAP request as the intended actor for a SOAP header entry

Field Detail
URI_NS_SOAP_ENVELOPE

public static final java.lang.string URI_NS_SOAP_ENCODING
The namespace identifier for the SOAP encoding

URI_SOAP_ACTOR_NEXT

public static final java.lang.String URI_SOAP_ACTOR_NEXT

The URI identifying the first application processing a SOAP request as the intended actor for a SOAP header entry.

Overview of Class SOAPConstants:

This class apperas to define a set of constants most of which seem to be Strings, again the importance of namespace identification can be clearly seen.

Interface SOAPElement.

public interface SOAPElement extends Node

An object representing the contents in a SOAPBody object, the contents in a SOAPHeader object, the content that can follow the SOAPBody object in a SOAPEnvelope object, or what can follow the detail element in a SOAPFault object. It is the base class of all of the classes that represent the SOAP objects as defined in the SOAP specification.

Method Summary

SOAPElement addAttribute(Name name, java.lang.String value)

Adds an attribute with the specified name and value to this SOAPElement object

SOAPElement addChildElement(Name name)

Creates a new SOAPElement object initialised with the given Name object and adds the new element to this SOAPElement object.

SOAPElement addChildElement(SOAPElement element)
Add a SOAPElement as a child of this SOAPElement instance.

SOAPElement addChildElement(java.lang.string localName)
Creates a new SOAPElement object initialized with the given String object and adds the new element to this SOAPElement object.

SOAPElement addChildElement(java.lang.String localName, java.lang.String prefix)
Creates a new SOAPElement object initialised with the specified local name and prefix and adds the new element to this SOAPElement object.

SOAPElement addChildElement(java.lang.String localName, java.lang.String prefix, java.lang.string uri)

Creates a new SOAPElement object initialised with the specified local name, prefix, and URI and adds the new element to this SOAPElement object.

SOAPElement addNamespaceDeclaration(java.lang.String prefix, java.lang.String uri)
Adds a namespace declaration with the specified prefix and URI to this SOAPElement object.

SOAPElement addTextNode(java.lang.String text)
Creates a new Text object initialised with the given String and adds it to this SOAPElement object.

Java.util.Iterator getAllAttributes( )
Returns an iterator over all of the attribute names in the SOAPElement object.

java.lang.String getAttributeValue(Name name)
Returns the value of the attribute with the specified name.

Name getElementName( )
Returns the name of this SOAPElement object.

java.lang.String getEncodingStyle( )
Returns the encoding style for this SOAPElement object.

java.util.Iterator getNamespacePrefixes( )
Returns an iterator of namespace prefixes.

java.lang.String getNamespaceURI(java.lang.String prefix)
Returns the URI of thr namespace that has thr given prefix.

boolean removeArrtibute(Name name)
Removes the attribute with the specified name.

boolean removeNamespaceDeclaration(java.lang.String prefix)
Removes the namespace declaration corresponding to the given prefix.

void setEncodingStyle(java.lang.String encodingStyle)
Sets the encoding style for this SOAPElement object to one specified.

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode, setParentElement

Overview of interface SoapElement:

All of the main parts of any SOAP message are comprised of elements, this classes main function is to prepare these elements before use in the various parts of the soap message.

class SOAPElementFactory

Public class SOAPElementFactory extends java.lang.Object

SOAPElementFactory is a factory for XML fragments that will eventually end up in the SOAP part. These fragments can be inserted as children of the SOAPHeader or SOAPBody or SOAPEnvelope. Elements created using this factory do not have the properties of an element that lives inside a SOAP header document. These elements are copied into the XML document tree when they are inserted.

SOAPElement create(Name name)
Deprecated. use
javax.xml.soap.SOAPFactory.createElement(javax.xml.soap.Name) instead.

SOAPElement create(java.lang.String localName)
Deprecated. use javax.xml.soap.SOAPFactory.createElement5String localName) instead

SOAPElement create(java.lang.String localName, java.lang.String prefix, java.lang.Strng uri)
Deprecated. Use javax.xml.soap.SOAPFactory.createElement5String localName, String prefix, String uri) instead

static
SOAPElementFactory nexInstance( )
Deprecated.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait.

Overview of Class SOAPElementFactory.
This class creates XML fragments that become part of the SOAP part of a message.

interface SOAPEnvelope

public interface SOAPEnvelope extends SOAPElement

The container for the SOAPHeader and SOAPBody portions of a SOAPPart object. By default, a SOAPMessage object is created with a SOAPPart object that has a SOAPEnvelope object. The SOAPEnvelope object by default has an empty SOAPBody object and an empty SOAPHeader object. The SOAPBody object is required, and the SOAPHeader object, though optional, is used in the majority of cases. If the SOAPHeader object is not needed, it can be deleted, which is shown later.

A client can access the SOAPHeader and SOAPBody objects by calling the methods SOAPEnvelope.getHeader and SOAPEnvelope.getBody. The following lines of code use these two methods after starting with the SOAPMessage object message to get the SOAPPart object sp, which is then used to get the SOAPEnvelope object se.

```
SOAPPart sp = message.getSOAPPart( ) ;
SOAPEnvelope se = sp.getEnvelope( ) ;
SOAPHeader sh = se.getHeader( ) ;
SOAPBody sb = se.getBody( ) ;
```

It is possible to change the body or header of a SOAPEnvelope object by retrieving the current one, deleting it, and then adding a new body or header. The javax.xml.soap.Node method deleteNode deletes the XML element (node) on which it is called. For example, the following line of code deletes the SOAPBody object that is retrieved by the method getBody.

```
se.getBody( ) .detachNode( ) ;
```

To create a SOAPHeader object to replace the one that was removed, a client uses the method SOAPEnvelope.addHeader, which creates a new header and adds it to the SOAPEnvelope object. Similarly, the method addBody creates a new SOAPBody object and adds it to the SOAPEnvelope object. The following code fragment retrieves the current header, removes it, and adds a new one. Then it retrieves the current body, removes it, and adds a new one.

```
SOAPPart sp = message.getSOAPPart( ) ;
SOAPEnvelope se = sp.getEnvelope( ) ;
se.getHeader( ) .detachNode( ) ;
SOAPHeader sh = se.addHeader( ) ;
se.getBody( ) .detachNode( ) ;
SOAPBody sb = se.addBody( ) ;
```

It is an error to add a SOAPBody or SOAPHeader object if one already exists.

The SOAPEnvelope interface provides three methods for creating Name objects. One method creates Name objects with a local name, a namespace prefix, and a namesapce URI. The second method creates Name objects with a local name and a namespace prefix, and the third creates Name objects with just a local name. The following line of code, in which se is a SOAPEnvelope object, creates a new Name object with all three.

```
Name name = se.createName( "  GetLastTradePrice"  , "  WOMBAT"  ,
```

" http://www.wombat.org/trader") ;

Method Summary

SOAPBody addBody( )
Creates a SOAPBobdy object and sets it as the SOAPBody object for this SOAPEnvelope object.

SOAPHeader addHeader( )
Creates a SOAPHeader object and sets it as the SOAPHeader object for this SOAPEnvelope object.

Name createName(java.lang.String localName)
Creates a nex Name object initialised with the given local name.

Name createName(java.lang.String localName, java.lang.String prefix, java.lang.String uri)
Creates a new Name object initalised with the given local name, namespace prefix, and namespace URI.

SOAPBody getBody( )
Returns the SOAPBody object assoiciated with this SOAPEnvelope object.

SOAPHeader getHeader( )

Returns the SOAPHeader object for this SOAPEnvelope object.

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, add ChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle.

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode, setParentElement

Overview of Interface SOAPEnvelope
This is a rather complicated interface the introduction should provide enough information to understand the overall implications of theis class. The envelope is basically the container for the SOAP Header and Body parts. The methods are there to add these parts and create the container.

Class SOAPEnvelope

public class SOAPEnvelope extends MessageElement implements SOAPEnvelope

Inner classes inherited from class org.apache.axis.message.MessageElement
MessageElement.QNameAttr

Field Summary
Protected
static org.apache.commons.logging.log log

java.lang.string messageType

java.util.vector trailers

Fields inherited from class org.apache.axis.message.MessageElement

isDirty, isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, message, name, namespaces, namespacesURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName.

Constructor Summary
SOAPEnvelope( )
SOAPEnvelope(boolean registerPrefixes, SOAPConstants soapConstants)
SOAPEnvelope(java.io.InputStream input)
SOAPEnvelope(SOAPConstants soapConstants)

Method Sumary
SOAPBody addBody( )
Creates a SOAPBody object and sets it as the SOAPBody object for this SOAPEnvelope object.

void addBodyElement(SOAPBodyElement element)

SOAPHeader addHeader( )
Creates a SOAPHeader object and sets it as the SOAPHeader object for this SOAPEnvelope object.

void addHeaderSOAPHeaderElement hdr)

void addTrailer(MessageElement element)

void clearBody( )

Name createName(java.lang.String localName)
Creates a new Name object iniialised with the given local name, namespace prefix, and namespace URI.

SOAPBody getBody( )
Returns the SOAPBody abject associated with this SOAPEnvelope object.

SOAPBodyElement getBodyByName(java.lang.String namespace, java.lang.String localpart)

java.util.vector getBodyElements( )

SOAPBodyElement getFirstBody( )

SOAPHeader getHeader( )
Returns the SOAPHeader object for this SOAPEnvelope object.

SOAPHeaderElement getHeaderByName(java.lang.String namespace, java.lang .String localPart)
Get a header by name(alwaysrespecting the currentmy in- scope actors list)

SOAPHeaderElement getHeaderByName(java.lang.String namspace, java.lang.String localPart, boolean accessAllHeaders)

Get a header by name, filtering for headers targeted st this engine depending on the accessAllHeaders parameter.

java.util.Vector getHeaders( )

java.util.Vector getHeadersByActor(java.util.ArrayList actors)
Get all the headers targeted at a list of actors.

java.util.Enumeration getHeaderByName(java.lang.String namespace, java.lang.String localPart)

java.util.Enumeration getHeadersByName(java.lang.String namespace, java.lang.String localPart, boolean accessAllHeaders)

Return an Enumeration of headers which match the given namespace and localärt.

java.lang.String getMessageType( )

java.util.vector getTrailers( )

void outputImpl(serializationContext context)
Should make SOAPSerialization exception,

void removeBody( )

void removeBodyElement(SOAPBodyElement element)

void removeHeader(SOAPHeaderElement hdr)

void removeHeaders( )

void removeTrailer(MessageElemnt element)

void setBody(SOAPBody body)

void setHeader(SOAPHeader hdr)

void setMessageType5java.lang.String messageType)

Methods inherited from class org.apache.axis.message.MessageElementaddAttribute,
addAttribute, addChild, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, detachNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue,

getAttributeValue, getChildElements, getChildElements, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, output, publishContents, publishToHandler, recycleNode, removeAttribute, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName, setNamespaceURI, setNSMappings, setObjectValue, setParentElement, setPrefix, setRecorder, setType, toString

Methods inherited from class java.lang.Object
clone,equals, finalize, getClass, hashCode, nofify, notifyAll, wait

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle.

Methods inherited from interface javax.xml.soap.Node

detachNode, getParentElement, getValue, recycleNode, setParentElement

Field Detail
log

protected static org.apache.commons.logging.Log.log

Trailers

public java.util.Vector trailers

MessageType
public java.lang.String messageType

Constructor Detail

SOAPEnvelope

public SOAPEnvelope( )

SOAPEnvelope
public SOAPEnvelope(SOAPConstants soapConstants)

SOAPEnvelope

public SOAPEnvelope(boolean registerPrefixes, SOAPConstants soapConstants)

SOAPEnvelope
public SOAPEnvelope(java.io.InputStraem input) throws org.xml.sax.SAXException

Class SOAPEnvelope Overview:

Following from the Interface the class actually creates the SOAP Envelope, once created it's simply a matter of compiling all the various elements together this is what the class achieves through it's various methods and inherited methods.

Class SOAPException

public class SOAPException extends java.lang.Exception

An exception that signals that a SOAP exception has occurred. A SOAPException object may contain a String that gives the reason for the exception, an embedded Throwable object, or both. This class provides methods for retrieving reason messages and for retrieving the embedded Throwable object.

Typical reasons for throwing a SOAPException object are problems such as difficulty setting a header, not being able to send a message, and not being able to get a connection with the provider. Reasons for embedding a Throwable object include problems such as input/output errors or a parsing problem, such as an error in parsing a header.

Constructor Summary

SOAPException( )
Constructs a SOAPException object with a reason or embedded Throwable object.

SOAPException(java.lang.String reason)
Constructs a SOAPException object with the given String as the reason for the exception being thrown.

SOAPException(java.lang.String reason, java.lang.Throwable cause)
Constructs a SOAPException object initialized with the given Throwable object

Method Summary

java.lang.Throwable getCause( )
Returns the Throwable object embedded in this SOapException if there is one.

java.lang.String getMessage( )
Returns the detail message for this SOApException object.

java.lang.Throwable initCause(java.lang.Throwable cause)
Initalizes the cause field of this SOapException object with the given Throwable object.

Methods inherited from class java.lang.Trowable

fillInStackTrace, getLocalizedMessage, printStackTrace, printStackTrace, printStackTrace, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

SOAPException

public SOAPException( )
Constructs a SOAPException object with no reason or embedded Throwable object

SOAPException

public SOAPException(java.lang.String reason)
Constructs a SOaPException object with the given String as the reason for the exception being thrown
Parameters
reason - a description of what caused the exception.

SOAPException

public SOAPException(java.lang.String reason,java.lang.Throwable cause)

Constructs a SOAPException object with the given String as the reason for the exception being thrown and the given Throwable object as an embedded exception.

Parameters:
reason - a description of what caused the exception
cause - a Throwable object that is to be embedded in this SOAPException object

****
initCause

public java.lang.Throwable initCause(java.lang.Throwable cause)
Initializes the cause field of this SOAPException object with the given Throwable object.

This method can be called at most once. It is generally called from within the constructor or immediately after the constructor has returned a new SOAPException object. If this SOAPException object was created with the constructor SOAPException(java.lang.Throwable) or SOAPException(java.lang.String, java.lang.Throwable) , meaning that its cause field already has a value, this method cannot be called even once.

Parameters:
cause - the Throwable object that caused this SOAPException object to be thrown. The value of this parameter is saved for later retrieval by the getCause( ) method. A null value is permitted and indicates that the cause is nonexistent or unknown.
Returns:
a reference to this SOAPException instance
Throws:
java.lang.IllegalArgumentException - if cause is this Throwable object. (A Throwable object cannot be its own cause.)
java.lang.IllegalStateException - if this SOAPException object was created with SOAPException(java.lang.Throwable) or SOAPException(java.lang.String, java.lang.Throwable) , o this method has already been called on this SOAPException object

****

Overview of class SOAPException:
Exceptions are an essential element to every system and throughout the API we see several such classes to deal with these exceptions. This particular one creates exception objects with either a reason or a object that has caused the exception.

Class SOAPFactory

public abstract class SOAPFactory extends java.lang.Object

SOAPFactory is a factory for creating various objects that exist in the SOAP XML tree. SOAPFactory can be used to create XML fragments that will eventually end up in the SOAP part. These fragments can be inserted as children of the SOAPHeaderElement or SOAPBodyElement or SOAPEnvelope. SOAPFactory also has methods to create javax.xml.soap.Detail objects as well as java.xml.soap.Name objects.

Constructor Summary
SOAPFactory( )

Method Summary

abstract Detail	createDetail( ) Creates a new Detail object which serves as a container for DetailEntry objects.
abstract SOAPElement	createElement(Name name) Create a SOAPElement object initalised with the given Name object.
abstract SOAPElement	createElement(java.lang.String localName) Create a SOAPElement object initialised with the given local name.
abstract SOAPElement	createElement(java.lang.String localName, java.lang.String prefix, java.lang.String uri) Create a new Name object initialised with the given local name.
abstract Name	createName(java.lang.String localName) Creates a new Name object initialized with the given local name, namespace prefix, and namespace URI.
static SOAPFactory	newInstance( ) Creates a new instance of SOAPFactory.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait

SOAPFactory Overview:

Again quiet similar to the SOAPElementFactory class previously seen this class creates XML fragments that will be used in the soap part.

Class SOAPFactoryImpl

public class SOAPFactoryImpl extends SOAPFactory

SOAP Element Factroy implementation

Constructor Summary
SOAPFactoryImpl( )

Method Summary
Detail createDetail( )
Creates a new Detail object which serves as a container for DetailEntry objects.

SOAPElement createElement(Name name)
Create a SOAPElement object initailised with the given Name object.

SOAPElement createElement(java.lang.String localName)
Create a SOAPElement object initialised with the given local name.

SOAPElement
createElement(java.lang.String localName, java.lang.String prefix, java.lang.String uri)
Create a new SOAPElement object initialised with the given local name, prefix and uri.

Name createName(java.lang.String localName)
Creates a new Name object initialised with the given local name.

Name
createName(java.lang.String localName, java.lang.String prefix, java.lang.String uri)
Creates a new Name object initialised with the given local name, namespace prefix, and namespace URI.

Methods inherited from class javax.xml.soap.SOAPFactory
newInstance

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait.

Constructor Detail
SOAPFactoryImpl

public SOAPFactoryImpl( )

Overview of Class SoapFactoryImpl:
This class simply implements the previous SOAPFactory class

Interface SOAPFault

public interface SOAPFault extends SOAPBodyElement

An element in the SOAPBody object that contains error and/or status information. This information may relate to errors in the SOAPMessage object or to problems that are not related to the content in the message itself. Problems not related to the message itself are generally errors in processing, such as the inability to communicate with an upstream server.

The SOAPFault interface provides methods for retrieving the information contained in a SOAPFault object and for setting the fault code, the fault actor, and a string describing the fault. A fault code is one of the codes defined in the SOAP 1.1 specification that describe the fault. An actor is an intermediate recipient to whom a message was routed. The message path may include one or more actors, or, if no actors are specified, the message goes only to the default actor, which is the final intended recipient.

Method Summary

Detail addDetail( )
Creates a Detail object and sets it as the Detail object for this SOAPFault object.

Detail getDetail( )
Returns the detail element for this SOAPFault object.

java.lang.String getFaultActor( )
Gets the fault actor for this SOAPFault object.

java.lang.String getFaultCode( )
Gets the fault code for this SOAPFault object

java.lang.String getFaultString( )
Gets the fault string for this SOAPFault object.

void setFaultActor(java.lang.String faultActor)
Sets this SOAPFault object with the given fault actor.

void setFaultString(java.lang.String faultString)
Sets the fault string for this SOAPFault object to the given string.

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode, setParentElement

Overview of SOAPFault:

This class obtains and returns information on faults if they occur.

Class SOAPFault

public class SOAPFault extends SOAPBodyElement implements SOAPFault

Inner classes inherited from class org.apache.axis.message.MessageElement
MessageElement.QNameAttr

Field Summary
protected fault
AxisFault

Fields inherited from class org.apache.axis.message.MessageElement
_isDirty, _isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, log, message, name, namespaces, namespaceURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName

Constructor Summary
SOAPFault(AxisFault fault)

SOAPFault(java.lang.String namespace, java.lang.String localName, java.lang.Sring prefix, org.xml.sax.Attributes attrs, DeserializationContext context)

Method Summary
Detail addDetail( )
Creates a Detail object and sets it as the Detail object for this SOAPFaultException object.

Detail getDetail( )
Returns the detail element for this SOAPFaultException object

AxisFault getFault( )

java.lang.String getFaultActor( )
Gets the fault actor for this SOAPFaultException object

java.lang.String getFaultCode( )
Gets the fault code for this SOAPFaultException object

java.lang.string getFaultString( )
Gets the fault string for this SOAPFaultException object.

void outputImpl(SerializationContext context)
Subclasses can override

void setFault(AxisFault fault)

void setFaultActor(java.lang.String faultActor)
Sets this SOAPFaultException object with the given fault actor.

void setFaultCode(java.lang.String faultCode)

void setFaultSring(java.lang.String faultString)
Sets the fault string for this SOAPFaultException object to the given string.

Methods inherited from class org.apache.axis.message.SOAPBodyElement
detachNode, setParentElement

Methods inherited from class org.apache.axis.message.MessageElement
addAttribute, addAttribute, addAttribute, addChild, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue, getAttributeValue, getChildElements, getChildElements, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, output, publishContents, publishToHandler, recycleNode, removeAttribute, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName, setNamespaceURI, setNSMappings, setObjectValue, setPrefix, setRecorder, setType, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait.

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle.

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode, setParentElement

Field Detail.
fault

protected AxisFault fault

Constructor Detail

SOAPFault

public SOAPFault(java.lang.String namespace,
java.lang.String localName,
java.lang.String prefix,
org.xml.sax.Attributes attrs,
DeserializationContext context)

SOAPFault
public SOAPFault(AxisFault fault)

Overview Class SOAPFault:
This class creates the fault objects and adds all the relevant details about each fault to the specific fault object.

Class SOAPFaultBuilder

public class SOAPFaultBuilder extends SOAPHandler implements Callback
Build a Fault body element.

Field Summary

protected DeserializationContext	context
protected SOAPFault	element
protected java.lang.String	faultActor
protected	faultClassName
protected QName	faultCode
protected org.w3c.dom.Element[]	faultDetails
protected Java.lang.String	faultString

Fields inherited from class org.apache.axis.message.SOAPHandler
myElement

Constructor Summary
SOAPFaultBuilder(SOAPFault element, DeserializationContext context)

Method Summary
void
endElement(java.lang.String namespace, java.lang.String localName, DeserializationContext context)
Final call back where we can populate the exception with data.

void
onEndChild(java.lanag.String namespace, java.lang.String localName, DeserializationContext context)

SOAPHandler
onStartChild(java.lang.String namespace, java.lang.String name, java.lang.String prefix, org.xml.sax.Attributes attributes, DeserializationContext context)

void
setValue(java.lang.object value, java.lang.object hint)

Methods inherited from class org.apache.axis.message.SOAPHandler
makeNewElement, startElement

Methods inherited from class org.xml.sax.helpers.DefaultHandler
Characters, endDocument, endElement, endPrefixMapping, error, fatalError, ignorableWhitespace, notationDecl, processingInstruction, resolveEntity, setDocumentLocator, skippedEntity,

startDocument, startElement, startPrefixMapping, unparsedEntityDecl, warning

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait.

Field Detail
element

protected SOAPFault element

context
protected DeserializationContext context

faultClassName
protected java.lang.String faultClassName

faultCode
protected QName faultCode

faultString
protected java.lang.String faultString

fullActor
protected java.lang.String faultActor

faultDetails
protected org.w3c.dom.Element[] faultDetails

Constructor Detail

SOAPFaultBuilder

public SOAPFaultBuilder(SOAPFault element, Deserialization context)

Overview of Class SOAPFaultBuilder:

This class compiles information in the form of elements which is then added to specific Fault objects

Interface SOAPFaultElement

public interface SOAPFaultElement extends SOAPElement

A representation of the contents in a SOAPFault object. The Detail interface is a SOAPFaultElement object that has been defined.

Content is added to a SOAPFaultElement using the SOAPElement method addTextNode.

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode

Overview of Interface SOAPFaultElement:

See the next class:

Class SOAPFaultElement

public class SOAPFaultElement extends MessageElement implements SOAPFaultElement
SOAP Fault implementation

Fields inherited from class org.apache.axis.message.MessageElement
_isDirty, _isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, log, message, name, namespaces, namespaceURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName

Constructor Summary
SOAPFaultElement( )

Methods inherited from class org.apache.axis.message.MessageElement
addAttribute, addAttribute, addAttribute, addChild, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, detachNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue, getAttributeValue, getChildElements, getChildElements, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getQName, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, makeAttributesEditable, output, outputImpl, publishContents, publishToHandler, recycleNode, removeAttribute, removeChild, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName, setNamespaceURI, setNSMappings, setObjectValue, setParentElement, setPrefix, setQName, setRecorder, setType, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, gerClass, hashCode, notify, notifyAll, wait.

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
detachNode, getParentElement, getValue, recycleNode, setParentElement

Constructor Detail

SOAPFaultElement

public SOAPFaultElement()

overview of Class SOAPFaultElement:
This class and the previous interface are primarily concerned with the collection of information about any fault that occurs and the creation of Fault elements.

Class SOAPFaultException

public class SOAPFaultException extends java.lang.RuntimeException

The SOAPFaultException exception represents a SOAP fault.

The message part in the SOAP fault maps to the contents of faultdetail element accessible through the getDetail method on the SOAPFaultException. The method createDetail on the javax.xml.soap.SOAPFactory creates an instance of the javax.xml.soap.Detail.

The faultstring provides a human- readable description of the SOAP fault. The faultcode element provides an algorithmic mapping of the SOAP fault.

Constructor Summary
SOAPFaultException(QName faultcode, java.lang.String faultString, java.lang.String faultactor, Detail detail)
Constructor for SOAPFaultException

Method Summary

Detail	getDetail( )
Gets the detail element	
java.lang.String	getFaultActor( )
Gets the faultactor element	
QName	getFaultCode( )
java.lang.String	getFaultString( )
Gets the faultstring element	

Methods inherited from class java.lang.Throwable
fillInStackTrace, getLocalizedMessage, getMessage, printStackTrace, printStackTrace, toString.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait.

Constructor Detail

SOAPFaultException

public SOAPFaultException(QName faultcode,
java.lang.String faultstring,
java.lang.String faultactor,
Detail detail)
Constructor for SOAPFaultException

Parameters:
faultcode - QName for the SOAP faultcode
faultstring - faultstring element of SOAP fault
faultactor - faultactor element of SOAP fault
detail - faultdetail element of SOAP fault

Overview of Class SOAPFaultException:
This is another form of fault similar to the the previous fault classes.

Class SOAPHandler

public class SOAPHandler extends org.xml.sax.helpers.DefaultHandler

Field Summary
MessageElement myElement

Constructor Summary
SOAPHandler( )

Method Summary
void
endElement(java.lang.String namspace, java.lang.String localName, DeserializationContext context)

MessageElement
makeNewElement(java.lang.String namespace, java.lang.String localName, java.lang.String prefix, org.xml.sax.Attributes attributes, DeserializationContext context)

void
onEndChild(java.lang.Sring namespace, java.lang.String localName, deserializationContext context)

SOAPHandler
onStartChild(java.lang.String namespace, java.lang.String localName, java.lang.String prefix, org.xml.sax.Attributes attributes, DeserializationContext context)

void
startElement(java.lang.String namespace, java.lang.String localName, java.lang.String prefix, org.xml.sax.Attributes attributes, DeserializationContext context)

Methods inherited from class org.xml.sax.helpers.DeafultHandler
characters, endDocument, endElement, endPrefixMappng, error, fatalError, ignorableWhitespace, notationDecl, processingInstruction, resolveEntity, setDocumenLocator, skippedEntity, startDocument, startElement, startPrefixMapping, unparsedEntityDecl, warning

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, haschCode, notify, notifyAll, toString, wait.

FieldDetail
myElement
public MessageElement myElement

Constructor Detail

SOAPHandler

public SOAPHandler( )

Interface SOAPHeader

public interface SOAPHeader extends SOAPElement

A representation of the SOAP header element. A SOAP header element consists of XML data that affects the way the application- specific content is processed by the message provider. For example, transaction semantics, authentication information, and so on, can be specified as the content of a SOAPHeader object.

A SOAPEnvelope object contains an empty SOAPHeader object by default. If the SOAPHeader object, which is optional, is not needed, it can be retrieved and deleted with the following line of code. The variable se is a SOAPEnvelope object.

```
se.getHeader( ) .detachNode( ) ;
```

A SOAPHeader object is created with the SOAPEnvelope method addHeader. This method, which creates a new header and adds it to the envelope, may be called only after the existing header has been removed.

```
se.getHeader( ) .detachNode( ) ;
SOAPHeader sh = se.addHeader( ) ;
```

A SOAPHeader object can have only SOAPHeaderElement objects as its immediate children. The method addHeaderElement creates a new HeaderElement object and adds it to the SOAPHeader object. In the following line of code, the argument to the method addHeaderElement is a Name object that is the name for the new HeaderElement object.

```
SOAPHeaderElement shElement = sh.addHeaderElement( name) ;
```

Method Summary

SOAPHeaderElement addHeaderElement(Name name)

Creates a new SOAPHeaderElement object initialized with the specific name and adds it to this SOAPHeader object.

java.util.Iterator examineHeaderElements(java.lang.String actor)
Returns a list of all the SOAPHeaderElement objects in this SOApHeader object that have the specified actor.

java.util.Iterator
extractHeaderElements(java.lang.String actor)
Returns a list of all the SOAPHeaderElement objects in this SOAPHeader object that have the specified actor and detaches them from this SOAPHeader
object.

Methods inherited from interface javax.xml.soap.SOAPElement

addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node

detachNode, getParentElement, getValue, recycleNode, setParentElement

Class SOAPHeader

public class SOAPHeader extends MessageElement implements SOAPHeader

Inner classes inherited from class org.apache.axis.message.MessageElement
MessageElement.QNameAttr

Fields inherited from class org.apache.axis.message.MessageElement
_isDirty, _isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, log, message, name, namespaces, namespaceURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName.

Constructor Summary
SOAPHeader(java.lang.String namespace, java.lang.String localPart, java.lang.String prefix, org.xml.sax.Attributes attributes, DeserializationContext context, SOAPConstants soapConsts)

Method Summary
void
addChild(MessageElement el)
Note that this method will log a error and no- op if there is avalue(set using setObjectValue) in the MessageElement.

SOAPHeaderElement
addHeaderElement5Name name)
Creates a new SOAPHeaderElement object initialized with the specified name and adds it to this SOAPHeader object.

void
detachNode( )
Removes this Node object from the tree.

java.util.Iterator
examineHeaderElements(java.lang.String actor)
Returns a list of all the SOAPHeaderElement objects in this SOAPHeader object that have the specified actor.

java.util.Iterator
extractHeaderElements(java.lang.String actor)

Returns a list of all the SOAPHeaderElement objects in this SOAPHeader object that have the specified actor and detaches them from this SOAPHeader object.

java.util.Iterator
getChildElements5Name name)
Returns an iterator over all the child elements with the specified name.

protected void
outputImpl(SerializationContext context)
Subclassses can override

void removeChild(MessageElement child)
Remove a child element

void
setParentElement(SOAPElement parenr)
Sets the parent of this Node object to the given SOAPElement object

Methods inherited from class org.apache.axis.message.MessageElement
addAttribute, addAttribute, addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue, getAttributeValue, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getQName, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, makeAttributesEditable, output, publishContents, publishToHandler, recycleNode, removeAttribute, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName, setNamespaceURI, setNSMappings, setObjectValue, setPrefix, setQName, setRecorder, setType, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait

Methods inherited from interface javax.xml.soap.SOAPElement
addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
getParentElement, getValue, recycleNode

Constructor Detail

SOAPHeader

public SOAPHeader(java.lang.String namespace,
java.lang.String localPart,
java.lang.String prefix,
org.xml.sax.Attributes attributes,
DeserializationContext context,
SOAPConstants soapConsts)

Overview of class SOAPHeader:

Creates the specified soapHeader object which comprises for several elements and uses linked lists to store the hedaer correctly.

Interface SOAPHeaderElement

public interface SOAPHeaderElement extends SOAPElement

An object representing the contents in the SOAP header part of the SOAP envelope. The immediate children of a SOAPHeader object can be represented only as SOAPHeaderElement objects.

A SOAPHeaderElement object can have other SOAPElement objects as its children.

Method Summary

java.lang.String getActor( )
Returns the uri of the actor associated with this SOAPHeaderElement object

boolean getMustUnderstand( )
Returns whether the mustUnderstand attribute for this SOAPHeaderElement object is turned on.

void setActor(java.lang.String actorURI)
Sets the actor associated with this SOApHeaderElement object to the specified actor.

void setMustUnderstand(boolean mustUnderstand)
Sets the mustUnderstand attribute for this SOApHeaderElement object to br on or off.

Methods inherited from interface javax.xml.soap.SOAPElement

addAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node

detachNode, getParentElement, getValue, recycleNode, setParentElement.

Overview of Interface SOAPHeaderElement:

See intro:

Class SOAPHeaderElement

public class SOAPHeaderElement extends MessageElement implements SOAPHeaderElement

A simple header element abstraction. Extends MessageElement with header- specific stuff like mustUnderstand, actor, and a 'processed' flag.

Inner classes inherited from class org.apache.axis.message.MessageElement
MessageElement.QNameAttr

Field Summary

protected java.lang.String	actor
protected boolean	mustUnderstand
protected boolean	processed

Fields inherited from class org.apache.axis.message.MessageElement
_isDirty, _isRoot, attributes, context, elementRep, encodingStyle, endEventIndex, href, id, log, message, name, namespaces, namespaceURI, parent, prefix, qNameAttrs, recorder, startContentsIndex, startEventIndex, textRep, typeQName

Constructor Summary
SOAPHeaderElement(org.w3c.dom.Element elem)
SOAPHeaderElement(Name name)
SOAPHeaderElement(java.lang.String namespace, java.lang.String. localPart)
SOAPHeaderElement(java.lang.String namespace, java.lang.String localPart, java.lang.Object value)
SOAPHeaderElement(java.lang.String namespace, java.lang.String localPart, java.lang.String prefix, org.xml.sax.Attributes attributes, DeserializationContext context)

Method Summary

void	detachNode( )
	Removes this Node object from the tree
java.lang.String	getActor( )
	Returns the uri of the actor associated with this SOAPHeaderElement object
boolean	getMustUnderstand( )
	Returns whether the mustUnderstand attribute for this SOAPHeaderElement object is turned on
boolean	isProcessed( )
void	setActor(java.lang.String a)
	Sets the actor associated with this SOAPHeaderElement object to the specified actor.
void	setMustUnderstand(boolean b)
	Sets the mustUnderstand attribute for this SOAPHeaderElement object to be on or off.

void setParentElement(SOAPElement parent)
Sets the parent of this Node object to the given SOAPElement object.

void
setProcessed(boolean value)

Methods inherited from class org.apache.axis.message.MessageElement
addAttribute, addAttribute, addAttribute, addChild, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addMapping, addNamespaceDeclaration, addTextNode, getAllAttributes, getAsDocument, getAsDOM, getAttributes, getAttributeValue, getAttributeValue, getChildElements, getChildElements, getChildren, getCompleteAttributes, getElementName, getEncodingStyle, getEnvelope, getFixupDeserializer, getHref, getID, getName, getNamespacePrefixes, getNamespaceURI, getNamespaceURI, getObjectValue, getParentElement, getPrefix, getPrefix, getQName, getRealElement, getRecorder, getType, getValue, getValueAsType, isDirty, isRoot, makeAttributesEditable, output, outputImpl, publishContents, publishToHandler, recycleNode, removeAttribute, removeChild, removeNamespaceDeclaration, setAttribute, setContentsIndex, setDirty, setEncodingStyle, setEndIndex, setEnvelope, setFixupDeserializer, setName, setNamespaceURI, setNSMappings, setObjectValue, setPrefix, setQName, setRecorder, setType, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Methods inherited from interface javax.xml.soap.SOAPElement
ddAttribute, addChildElement, addChildElement, addChildElement, addChildElement, addChildElement, addNamespaceDeclaration, addTextNode, getAllAttributes, getAttributeValue, getChildElements, getChildElements, getElementName, getEncodingStyle, getNamespacePrefixes, getNamespaceURI, removeAttribute, removeNamespaceDeclaration, setEncodingStyle

Methods inherited from interface javax.xml.soap.Node
getParentElement, getValue, recycleNode

Field Detail
processed

protected boolean processed

actor
protected java.lang.String actor

mustUnderstand
protected boolean mustUnderstand

Constructor Detail

SOAPHeaderElement
public SOAPHeaderElement(java.lanag.String namespace, java.lang.String localPart)

SOAPHeaderElement
public SOAPHeaderElement(Name name)

SOAPHeaderElement

public SOAPHeaderElement(java.lang.String namespace,
java.lang.String localPart,
java.lang.Object value)

SOAPHeaderElement
public SOAPHeaderElement(org.w3c.dom.Element elem)

SOAPHeaderElement

public SOAPHeaderElement(java.lang.String namespace,
java.lang.String localPart,
java.lang.String prefix,
org.xml.sax.Attributes attributes,
DeserializationContext context)

Overview of class SOAPHeaderMessage:
Creates instances of Elements used in the soap header

Class SOAPMessage

public abstract class SOAPMessage extends java.lang.Object

The root class for all SOAP messages. As transmitted on the " wire" , a SOAP message is an XML document or a MIME message whose first body part is an XML/SOAP document.

A SOAPMessage object consists of a SOAP part and optionally one or more attachment parts. The SOAP part for a SOAPMessage object is a SOAPPart object, which contains information used for message routing and identification, and which can contain application- specific content. All data in the SOAP Part of a message must be in XML format.

A new SOAPMessage object contains the following by default:

A SOAPPart object
A SOAPEnvelope object
A SOAPBody object
A SOAPHeader object
The SOAP part of a message can be retrieved by calling the method SOAPMessage.getSOAPPart( ) . The SOAPEnvelope object is retrieved from the SOAPPart object, and the SOAPEnvelope object is used to retrieve the SOAPBody and SOAPHeader objects.

```
SOAPPart sp = message.getSOAPPart( ) ;
SOAPEnvelope se = sp.getEnvelope( ) ;
SOAPBody sb = se.getBody( ) ;
SOAPHeader sh = se.getHeader( ) ;
```

In addition to the mandatory SOAPPart object, a SOAPMessage object may contain zero or more AttachmentPart objects, each of which contains application- specific data. The SOAPMessage interface provides methods for creating AttachmentPart objects and also for adding them to a SOAPMessage object. A party that has received a SOAPMessage object can examine its contents by retrieving individual attachment parts.

Unlike the rest of a SOAP message, an attachment is not required to be in XML format and can therefore be anything from simple text to an image file. Consequently, any message content that is not in XML format must be in an AttachmentPart object.

A MessageFactory object creates new SOAPMessage objects. If the MessageFactory object was initialized with a messaging Profile, it produces SOAPMessage objects that conform to that Profile. For example, a SOAPMessage object created by a MessageFactory object initialized with the ebXML Profile will have the appropriate ebXML headers.

Constructor Summary

SOAPMessage( )

Method Summary

abstract void addAttachmentPart(AttachmentPart attachmentpart)
Adds the given AttachmentPart object to this SOAPMessage object.

abstract int countAttachments( )
Gets a count of the number of attachments in this message

abstract AttachmentPart createAttachmentPart( )
Creates a new empty AttachmentPart object

AttachmentPart createAttachmentPart(javax.activation.DataHandler datahandler)
Creates an AttachmentPart object and populates it using the given DataHandler object.

AttachmentPart createAttachmentPart(javax.activation.DataHandler datahandler)
Creates an AttachmentPart object and populates it using the given DataHandler object.

AttachmentPart creaeAttachmentPart(java.lang.object content,java.lang.String contentType)
Creates an AttachmentPart object and populates it with the specified data of the specified content type.

abstract java.util.Iterator getAttachments( )
Retrieves all the AttachmentPart objectsthat are part of this SOAPMessage object.

abstract java.util.Iterator getAttachments(MimeHeaders headers)
Retrieves all the AttachmentPart objects that have header entries that match the specified headers.

abstract java.lang.String getContentDescription( )
Retrieves a description of this SOAPMessage object's content.

abstract MimeHeaders getMimeHeaders( )
Returns all the transport- specific MIME headers for this SOAPMessage object in a transport- independent fashion.

abstract SOAPPart getSOAPPart( )
Gets the SOAP part of this SOAPMessage object.

abstract void removeAllAttachments( )
Remoes all AttachmentPart objects that have been added to this SOAPMessage object.

abstract void saveChanges( )
Updates this SOAPMesage object with all the changes that have been made to it.

abstract boolean saveRequired( )
Indicates whether this SOAPMessage object has had the method saveChanges called on it.

abstract void setContentDescription(java.lang.String description)
Sets the description of this SOAPMessage object's content with the given description.

abstract void writeTo(java.io.OutputStream out)
Writes this SOAPMessage object to the given output stream.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait.

Constructor Detail

SOAPMessage

public SOAPMessage( )

Overview of CLASS SOAPMessage:

see intro.

Interface SOAPMessageContext

public interface SOAPMessageContext extends MessageContext

The interface javax.xml.rpc.soap.SOAPMessageContext provides access to the SOAP message for either RPC request or response. The javax.xml.soap.SOAPMessage specifies the standard Java API for the representation of a SOAP 1.1 message with attachments.

Method Summary

SOAPMessage getMessage( )
Gets the SOAPMessage from this message context.

java.lang.String[] getRoles( )
Gets the SOAP actor roles associated with an execution of the HandlerChain and its contained Handler instances.

void setMessage(SOAPMessage message)
Sets the SOAPMessage for this message context

Methods inherited from interface javax.xml.rpc.handler.MessageContext

containsProperty, getProperty, getPropertyNames, removeProperty, setProperty

Overview of Interface SOAPMessageContext:

see intro.

Class SOAPMonitorConstants

public class SOAPMonitorConstants extends java.lang.Object

Field Summary

static java.lang.String	SOAP_MONITOR_ID Unique SOAP monitor id tag
static java.lang.String	SOAP_MONITOR_PORT Servlet initialization parameters names
static int	SOAP_MONITOR_REQUEST SOAP message types
static int	SOAP_MONTOR_REPONSE

Constructor Summary

SOAPMonitorConstants( )

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field detail

SOAP_MONITOR_REQUEST

public static final int SOAP_MONITOR_REQUEST
SOAP message types.

SOAP_MONITOR_RESPONSE

public static final int SOAP_MONITOR_RESPONSE

SOAP_MONITOR_PORT
public static final java.lang.String SOAP_MONITOR_PORT
Servlet initialization parameter names.

SOAP_MONITOR_ID
public static final java.lang.String SOAP_MONITOR_ID
Unique SOAP monitor id tag

Constructor Detail

SOAPMonitorConstants

public SOAPMonitorConstants( )

Overview of Class SOAPMonitorConstants:

This class monitors four constants two strings and two ints the methods in this class are primarily there to request and respond the constants.

Class SOAPMonitorHandler

public class SOAPMonitorHandler extends BasicHandler
This handler is used to route SOAP messages to the SOAP monitor service.

Fields inherited from class org.apache.axis.handlers.BasicHandler
log, name,options

Constructor Summary
SOAPMonitorHandler( )
Constructor

Method Summary
void invoke(MessageContext messageContext)
Process and SOAP message

Methods inherited from class org.apache.axis.handlers.BasicHandler
canHandleBlock, cleanup, generateWSDL, getDeploymentData, getName, getOption, getOptions, getUnderstoodHeaders, init, initHashtable, onFault, setName, setOption, setOptionDefault, setOptions.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail
SOAPMonitorHandler

public SOAPMonitorHandler( )
Constructor

Overview of Class SOAPMonitorHandler:

This is a very simple method which when invoked routes soap messages to the soap monitor service.

Class SOAPMonitorService

public class SOAPMonitorService
extends javax.servlet.http.HttpServlet

This is a SOAP Monitor Service class. During the HTTP server startup, the servlet init method is invoked. This allows the code to open a server socket that will be used to communicate with running applets. When an HTTP GET request is received, the servlet dynamically produces an HTML document to load the SOAP monitor applet and supply the port number being used by the server socket (so the applet will know how to connect back to the server) . Each time a socket conneciton is established, a new thread is created to handle communications from the applet. The publishMethod routine is invoked by the SOAP monitor handler when a SOAP message request or response is detected. The information about the SOAP message is then forwared to all current socket connections for display by the applet.

Constructor Summary
SOAPMonitorService( )
Constructor

Method Summary
void destroy( )
Servlet termination

void
doGet(javax.servlet.http.HttpServletRequest request, javax.servlet.http.HttpservletResponse response)
HTTP GET request

void init( )
Servlet initialiation

static void
publishMessage(java.lang.Long id, java.lang.Integer type,java.lang.String target, java.lang.String soap)
Publish a SOAP message to listeners

Methods inherited from class javax.servlet.http.Http.Servlet
doDelete, doHead, doOptions, doPost, doPut, doTrace, getLastModified, service, service

Methods inherited from class javax.servlet.GenericServlet
getInitParameter, getInitParameterNames, getServletConfig, getServletContext, getServletInfo, getServletName, init, log, log

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail
SOAPMonitorService

public SOAPMonitorService( )

Overview of class SOAPMonitorService:

see intro:

Class SOAPPart

public abstract class SOAPPart extends java.lang.Object

The container for the SOAP- specific portion of a SOAPMessage object. All messages are required to have a SOAP part, so when a SOAPMessage object is created, it will automatically have a SOAPPart object.

A SOAPPart object is a MIME part and has the MIME headers Content- Id, Content- Location, and Content- Type. Because the value of Content- Type must be " text/xml" , a SOAPPart object automatically has a MIME header of Content- Type with its value set to " text/xml" . The value must b " text/xml" because content in the SOAP part of a message must be in XML format. Content that is not of type " text/xml" must be in an AttachmentPart object rather than in the SOAPPart object.

When a message is sent, its SOAP part must have the MIME header Content- Type set to " text/xml" Or, from the other perspective, the SOAP part of any message that is received must have the MIME header Content- Type with a value of " text/xml" .

A client can access the SOAPPart object of a SOAPMessage object by calling the method SOAPMessage.getSOAPPart. The following line of code, in which message is a SOAPMessage object, retrieves the SOAP part of a message.

SOAPPart soapPart = message.getSOAPPart( ) ;

A SOAPPart object contains a SOAPEnvelope object, which in turn contains a SOAPBody object and a SOAPHeader object. The SOAPPart method getEnvelope can be used to retrieve the SOAPEnvelope object.

Constructor Summary
SOAPPart( )

Method Summary

abastract void
addMimeHeader5java.lang.String name, java.lang.String value)
Creates a MimeHeader object with the specified name and value and adds it to this SOAPPart object

abstract getAllMimeHeaders( )
java.util.Iterator Retrieves all the headers for this SOAPPart object as an iterator over the MimeHeader objects.

abstract source getContent( )
Returns the content of the SOAPEnvelope as a JAXP source object.

java.lang.String getContentId( )
Retrieves the value of the MIME header whoses name is " Content- id"

java.lang.String getContentLocation( )
Retrieves the value of the MIME header whose name is " Content- Location"

abstract getEnvelope( )
SOAPEnvelope Gets the SOAPEnvelope object associated with this SOAPPart object.

abstract getMatchingMimeHeaders(java.lang.String[] names)
java.util.Iterator Retrieves all MimeHeader objects that match a name in a given array.

abstract getMimeHrader(java.lang.String name)
java.lanag.String[] Gets all the values of the MimeHeader object in this SOAPPart object that is identified by the given String.

abstract getNonMatchingMimeHeaders(java.lang.String[] names)
java.util.Iterator Retrieves all MimeHeader objects whose name does not match a name in the given array.

abstract void removeAllMimeHeaders( )
Removes all the MimeHeader objects for this SOAPEnvelope object.

abstract void removeMimeHeader(java.lang.String header)
Removes all MIME headers that match the given name.

abstract void setContent(Source source)
Sets the content of the SOAPEnvelope object with data from the given Source object.

void setContentId(java.lang.String contentId(java.lang.String contentId)
Sets the value of the MIME header named " Content- id" to the given String.

void setContentLocation(java.lang.String contentLocation)
Sets the value of the MIME header " Content- Location" to the given String.

abstract void setMimeHeader(java.lang.String name, java.lang.String value)
Changes the first header entry that matches the given header name so that its value is the given value, adding a new header with the given name and value if no existing header is a match.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, nofityAll, toString, wait

Constructor Detail
SOAPPart

public SOAPPart( )

Overview of Class SOAPPart:

see intro.

Class SOAPService

public class SOAPService extends SimpleTargetedChain

A SOAPService is a Handler which encapsulates a SOAP invocation. It has an request chain, an response chain, and a pivot- point, and handles the SOAP semantics when invoke( ) d.

Field Summary

protected
static org.apache.commons.logging.Log log

Fields inherited from class org.apache.axis.SimpleTargetedChain
pivotHandler, requestHandler, responseHandler

Fields inherited from class org.apache.axis.SimpleChain
handlers, invoked

Fields inherited from class org.apache.axis.handlers.BasicHandler
name, options

Constructor Summary
SOAPService( )
Standard, no- arg constructor.

SOAPService(Handler serviceHandler)
Convenience constructor for wrapping SOAP semantics " service handlers" which actually do work

SOAPService(Handler regHandler, Handler pivHandler, Handler respHandler)
Constructor with real or null request, pivot, and response handlers.

Method Summary.

boolean	availableFromTransport(java.lang.String transportName)
void	disableTransport(java.lang.String transportName) Disable access to this service from a particular transport
void	enableTransport(java.lang.String transportName) Make this service available on a particular transport
void	generateWSDL(MessageContext msgContext) Generate WSDL
serviceDesc	getInitializedServiceDesc(MessageContext msgContext)
serviceDesc	getServiceDescription( )
style	getStyle( )
TypeMappingRegistry	getTypeMappingRegistry( )
boolean	needsHighFidelityRecording( )

void setEngine(AxisEngine engine)
Tell this service which engine it's deployed to.

void setHighFidelityRecording(boolean highFidelityRecording)

void setPropertyParent(java.util.Hashtable parent)

void setServiceDescription(serviceDesc serviceDescription)

void setStyle(Style style)

void setTypeMappingRegistry(TypeMappingRegistry map)

void start( )
Placeholder for " enable this service" method

void stop( )
Placeholder for " disable this service" method

Methods inherited from class org.apache.axis.SimpleTargetedChain
getPivotHandler, getRequestHandler, getResponseHandler, init

Methods inherited from class org.apache.axis.SimpleChain
addHandler, canHandleBlock, cleanup, contains, getDeploymentData, getHandlers, init, invoke, onFault.

Methods inherited from class org.apache.axis.handlers.BasicHandler
getName, getOption, getOptions, getUnderstoodHeaders, initHashtable, setName, setOption, setOptionDefault, setOptions

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.apache.axis.Chain
addHandler, contains, getHandlers

Methods inherited from interface org.apache.axis.Handler
canHandleBlock, cleanup, getDeploymentData, getName, getOption, getOptions, getUnderstoodHeaders, init, invoke, onFault, setName, setOption, setOptions

Field Detail
log
protected static org.apache.commons.logging.Log log

Constructor Detail

SOAPService
public SOAPService( )
Standard, no- arg constructor.

SOAPService

public SOAPService(Handler reqHandler,
Handler pivHandler,
Handler respHandler)
Constructor with real or null request, pivot, and response handlers. A special request handler is specified to inject SOAP semantics.

SOAPService

public SOAPService(Handler serviceHandler)
Convenience constructor for wrapping SOAP semantics around " service handlers" which actually do work.

Overview of Class SOAPService:

This is a rather complicated class but in essence it carries out A SOAPService which is a Handler which encapsulates a SOAP invocation. It has an request chain, an response chain, and a pivot- point, and handles the SOAP semantics when invoked.

Class SOAPUtils

public class SOAPUtils extends java.lang.Object

Field Summary

protected static org.apache commons .logging.Log	log

Constructor Summary

SOAPUtils( )

Method Summary

static java.lang.String	getNewContentIdValue( )

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait

Field Detail

log

protected static org.apache.commons.logging.Log log

Constructor Detail

SOAPUtls

public SOAPUtils( )

Overview of Class SOAPUtils: