

All Classes

Packages

[org.TELCERT.CRT.STT](#)

[org.TELCERT.CRT.STT.](#)

[schema](#)

[org.TELCERT.CRT.STT.utilities](#)

All Classes

[STTAppProfile](#)

[STTBaseSchema](#)

[STTConstants](#)

[STTConstraintsDocument](#)

[STTDefinitionSchema](#)

[STTMessages](#)

[STTReferencedSchema](#)

[STTResultantSchema](#)

[STTSchema](#)

[STTSchemaProcessor](#)

[STTSchemaSuperclass](#)

[STTTransformer](#)

[STTUtilities](#)

[STTXMLOutputter](#)

[SchemaTransformTool](#)

Packages

[org.](#)
[TELCERT.](#)
[CRT.STT](#)

[org.](#)
[TELCERT.](#)
[CRT.STT.](#)
[schema](#)

[org.](#)
[TELCERT.](#)
[CRT.STT.](#)
[utilities](#)

All Classes

[STTAppProfile](#)

[STTBaseSchema](#)

[STTConstants](#)

[STTConstraintsDocument](#)

[STTDefinitionSchema](#)

[STTMessages](#)

[STTReferencedSchema](#)

[STTResultantSchema](#)

[STTSchema](#)

[STTSchemaProcessor](#)

[STTSchemaSuperclass](#)

[STTTransformer](#)

[STTUtilities](#)

[STTXMLOutputter](#)

[SchemaTransformTool](#)

org.TELCERT.CRT.STT.schema

Class STTAppProfile

java.lang.Object

└ org.jdom.Document

└ [org.TELCERT.CRT.STT.schema.STTSchemaSuperclass](#)

└ org.TELCERT.CRT.STT.schema.STTAppProfile

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

public class **STTAppProfile**

extends [STTSchemaSuperclass](#)

This class is the type for the resultant schema. It extends the STTSchemaSuperclass class. It is able to apply processing related to condition elements. If condition elements are present within the instance, it will process them to create an associated condition elements document that holds only condition elements and elements that reference them through cnd attributes, In addition it also holds functionality related to the generation of a complete Xpath expression for a given modification element, and can verify if it contains any modification elements, and is related to a given base schema.

Author:

wluxton

See Also:

[Serialized Form](#)

Constructor Summary

[STTAppProfile](#)(org.jdom.Document baseDocument)

Method Summary

boolean	<p><u>containsConditionals</u>()</p> <p>This method parses this STTAppProfile instance for the presence of condition elements.</p>
org.jdom.Document	<p><u>generateConditionOnlyDocument</u>()</p> <p>This method creates a Document object that contains a number of condition elements that have been copied from the application profile.</p>
java.lang.String	<p><u>returnCompleteXPathExpressionFromModificationElement</u>(org.jdom.Element passedElement)</p> <p>This method produces an absolute XPath expression from values held by a modification subcomponent and its parents in either subElement or element attributes.</p>
boolean	<p><u>verifyAppProfileAndBaseSchemaRelationship</u>(org.jdom.Document baseSchema)</p> <p>This method assesses the relationship between a passed base schema and a passed application profile by looking at the value of the 'baseSchema' attribute in the root element of the application profile.</p>
boolean	<p><u>verifyModificationsElementExistence</u>()</p> <p>This method assesses whether this app profile contains a modifications element or not.</p>

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchemaSuperclass](#)

[addDefinitionDocumentIncludeElement](#),
[containsImportOrIncludeStatements](#),
[getElementWithAttributeValueInElement](#),
[getElementWithNameInElement](#), [getElementWithNameOrIDAttribute](#),
[getElementWithNameOrIDAttributeInElement](#),
[getElementWithReferencingAttribute](#), [getFilenameWithoutExtension](#),
[getImportAndIncludeStatements](#), [getIndexValueForFirstElement](#),
[getName](#), [getUpdatedFileName](#), [returnXPathElementFromElement](#),
[returnXPathElementFromThisSchema](#), [updateNamespace](#),
[updateVariablesForModification](#), [updateVersionString](#)

Methods inherited from class org.jdom.Document

```
addContent, addContent, addContent, addContent, clone,
cloneContent, detachRootElement, equals, getBaseURI, getContent,
getContent, getContent, getContentSize, getDescendants,
getDescendants, getDocType, getDocument, getParent, getProperty,
getRootElement, hashCode, hasRootElement, indexOf, removeContent,
removeContent, removeContent, removeContent, setBaseURI,
setContent, setContent, setContent, setContent, setDocType,
setProperty, setRootElement, toString
```

Methods inherited from class java.lang.Object

```
getClass, notify, notifyAll, wait, wait, wait
```

Constructor Detail

STTAppProfile

```
public STTAppProfile(org.jdom.Document baseDocument)
```

Method Detail

verifyModificationsElementExistance

```
public boolean verifyModificationsElementExistance()
```

This method assesses whether this app profile contains a modifications element or not.

Returns:

true if this application profile contains a modifications element false if not.

verifyAppProfileAndBaseSchemaRelationship

```
public boolean verifyAppProfileAndBaseSchemaRelationship(org.jdom.
Document baseSchema)
```

This method assesses the relationship between a passed base schema and a passed application profile by looking at the value of the 'baseSchema' attribute in the root element of the application profile. It then compares this to the file name of the base schmea URI. It returns true if there is a match, false if there is no match.

Parameters:

baseSchema - The schema to test for relationship

Returns:

true if there is a match between base schema file name and app. profile baseSchema attribute, false if there is no match

containsConditionals

```
public boolean containsConditionals()
```

This method parses this STTAppProfile instance for the presence of condition elements. It returns true if condition elements are found, false otherwise.

Returns:

true if condition elements are found, false otherwise.

generateConditionOnlyDocument

```
public org.jdom.Document generateConditionOnlyDocument()
```

This method creates a Document object that contains a number of condition elements that have been copied from the application profile. In addition to the condition elements, it also contains any element in the schema that references the condition elements through a cnd attribute.

Returns:

A JDOM Document object that contains a number of unformatted condition elements

returnCompleteXPathExpressionFromModificationElement

```
public java.lang.String  
returnCompleteXPathExpressionFromModificationElement(org.jdom.  
Element passedElement)
```

This method produces an absolute XPath expression from values held by a modification subcomponent and its parents in either subElement or element attributes.

Parameters:

`passedElement` - The element that the XPath expression and parent elements are drawn from.

Returns:

An absolute XPath expression

org.TELCERT.CRT.STT.schema

Class STTSchemaSuperclass

java.lang.Object

└ org.jdom.Document

└ org.TELCERT.CRT.STT.schema.STTSchemaSuperclass

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

Direct Known Subclasses:

[STTAppProfile](#), [STTResultantSchema](#), [STTSchema](#)

```
public abstract class STTSchemaSuperclass
```

```
extends org.jdom.Document
```

This abstract class provides functionality for STTSchemaSuperclass objects. It extends the org.Jdom Document class, and adds additional functionality to extract if the schema has any import or include statements, the ability to return if any are present, the ability to locate elements by element name or name and ID attributes within the schema or a given element. it can return elements for a given XPath expression, and also has processing related to returning the name of the schema and its updated name one the modifications have been applied.

Author:

wluxton

See Also:

[Serialized Form](#)

Constructor Summary

[STTSchemaSuperclass](#)(org.jdom.Document baseDocument)

Method Summary

void	addDefinitionDocumentIncludeElement ()
boolean	containsImportOrIncludeStatements ()
org.jdom.Element	getElementWithAttributeValueInElement (java.lang.String searchAttributeName, java.lang.String searchAttributeValue, org.jdom.Element searchElement)
org.jdom.Element	getElementWithNameInElement (java.lang.String elementName, org.jdom.Element searchElement)

This method inserts a new include element in to the modified schema as a definition document that holds new simpleType declarations from the Application Profile has been created.

This method evalautates the schema to determine if it contains either include or import elements.

This method returns an element object that has an attribute that matches the passed in searchAttributeName, and a value in the named attribute that matches the searchAttributeValue variable.

This method returns an element object with a name (not name attribute) that matches the passed in elementName variable if found in the passed element.

org. jdom. Element	getElementWithNameOrIDAttribute (java.lang.String elementName) This method returns an element object that has a 'name' or 'id' attribute that matches the passed in elementName variable if found in the root element of the holding document instance.
org. jdom. Element	getElementWithNameOrIDAttributeInElement (java.lang.String elementName, org.jdom.Element searchElement) This method returns an element object that has a 'name' or 'id' attribute that matches the passed in elementName variable if found in the passed element.
org. jdom. Element	getElementWithReferencingAttribute (java.lang.String elementName) This method returns an element object that has a 'type' or 'ref' attribute that matches the passed in elementName variable if found in the root element of the holding document instance.
java. lang. String	getFilenameWithoutExtension () This method returns a string holding the filename of the schema without the .xsd extension.
java. util. Vector	getImportAndIncludeStatements () This method returns a Vector containing any include or import elements from this schema.
java. lang. Integer	getIndexValueForFirstElement (org.jdom.Element passedParentElement) This method receives an element, and works out the index value of the first child element object it contains.
java. lang. String	getName ()
java. lang. String	getUpdatedFileName () This method returns a string holding an updated filename for use when the schema is to be written out after modifications have been applied.
org. jdom. Element	returnXPathElementFromElement (org.jdom.Element targetElement, org.jdom.Element modifyingElement, boolean generateAlertMessage) This method fetches an Element object from a Element object, using an XPath expression to identify the element.
org. jdom. Element	returnXPathElementFromThisSchema (org.jdom.Element modifyingElement, boolean generateAlertMessage) This method fetches an Element object from the holding STTSchemaSuperclass object, using an XPath expression to identify the element.
void	updateNamespace () This method updates the targetnamespace attribute to reflect that this has been modified (localised) during the application of the Application Profile data.
void	updateVariablesForModification () This method coordinates the updating of a number of attributes to reflect the changes introduced by producing a localised schema that has been derived from another schema.
void	updateVersionString () This method updates the version attribute with a data string to reflect that this has been modified (localised) during the application of the Application Profile data.

Methods inherited from class org.jdom.Document

addContent, addContent, addContent, addContent, clone, cloneContent, detachRootElement, equals, getBaseURI, getContent, getContent, getContent, getContentSize, getDescendants, getDescendants, getDocType, getDocument, getParent, getProperty, getRootElement, hashCode, hasRootElement, indexOf, removeContent, removeContent, removeContent, removeContent, setBaseURI, setContent, setContent, setContent, setContent, setDocType, setProperty, setRootElement, toString

Methods inherited from class java.lang.Object

getClass, notify, notifyAll, wait, wait, wait

Constructor Detail

STTSchemaSuperclass

```
public STTSchemaSuperclass(org.jdom.Document baseDocument)
```

Method Detail

getElementWithNameOrIDAttribute

```
public org.jdom.Element getElementWithNameOrIDAttribute(java.lang.String elementName)
```

This method returns an `element` object that has a 'name' or 'id' attribute that matches the passed in `elementName` variable if found in the root element of the holding document instance. It returns the located element, or `null` if an element was not found.

Parameters:

`elementName` - A string defining the name of the element to be located

Returns:

A named element that was located in the holding schema - `null` if not found

getElementWithReferencingAttribute

```
public org.jdom.Element getElementWithReferencingAttribute(java.lang.String elementName)
```

This method returns an `element` object that has a 'type' or 'ref' attribute that matches the passed in `elementName` variable if found in the root element of the holding document instance. It returns the located element, if found, or `null` if no matching element was found.

Parameters:

`elementName` - A string defining the name of the referenced element

Returns:

An element that was located in the passed element - `null` if not found

getElementWithNameOrIDAttributeInElement

```
public org.jdom.Element getElementWithNameOrIDAttributeInElement(java.lang.String elementName,
                                                                    org.jdom.
                                                                    Element searchElement)
```

This method returns an `element` object that has a 'name' or 'id' attribute that matches the passed in `elementName` variable if found in the passed element. It returns the located element, or `null` if an element was not found.

Parameters:

elementName - A string defining the name of the element to be located
searchElement - The element that is to be searched for the named element

Returns:

A named element that was located in the passed element - null if not found

getElementWithAttributeValueInElement

```
public org.jdom.Element getElementWithAttributeValueInElement( java.lang.  
String searchAttributeName,                                     java.lang.  
String searchAttributeValue,                                  org.jdom.  
Element searchElement)
```

This method returns an element object that has an attribute that matches the passed in searchAttributeName, and a value in the named attribute that matches the searchAttributeValue variable. It returns the located element, or null if an element was not found.

Parameters:

searchAttributeName - The attribute name used to identify the appropriate attribute
searchAttributeValue - The passed value used to check if the named attribute value provides a match
searchElement - The element that is to be searched for the element with the named attribute/attributeValue

Returns:

An element that was located in the passed element - null if not found

getElementWithNameInElement

```
public org.jdom.Element getElementWithNameInElement( java.lang.String elementName,  
org.jdom.Element searchElement)
```

This method returns an element object with a name (not name attribute) that matches the passed in elementName variable if found in the passed element. It returns the located element, or null if an element was not found.

Parameters:

elementName - A string defining the name of the element to be located
searchElement - The element that is to be searched for the named element

Returns:

A named element that was located in the passed element - null if not found

returnXPathElementFromThisSchema

```
public org.jdom.Element returnXPathElementFromThisSchema( org.jdom.  
Element modifyingElement,                                     boolean generateAlertMessage)
```

This method fetches an Element object from the holding STTSchemaSuperclass object, using an XPath expression to identify the element. The XPath expression is extracted from the passed-in modifying element. It will return a single Element if found - it will return null if an error is encountered.

Parameters:

modifyingElement - The Element holding the XPath expression

generateAlertMessage - A boolean flag controlling if a 'cannot find' message is generated or not

Returns:

an Element targeted by the XPath expression - null if no element found or if an error is encountered.

returnXPathElementFromElement

```
public org.jdom.Element returnXPathElementFromElement(org.jdom.Element targetElement,
org.jdom.
Element modifyingElement,
boolean generateAlertMessage)
```

This method fetches an Element object from a Element object, using an XPath expression to identify the element. The XPath expression is extracted from the passed-in modifying element. It will return a single Element if found - it will return null if an error is encountered.

Parameters:

targetElement - The Element to be targeted by the XPath expression

modifyingElement - The Element holding the XPath expression

generateAlertMessage - A boolean flag controlling if a 'cannot find' message is generated or not

Returns:

an Element targeted by the XPath expression - null if no element found or if an error is encountered.

getIndexValueForFirstElement

```
public java.lang.Integer getIndexValueForFirstElement(org.jdom.
Element passedParentElement)
```

This method receives an element, and works out the index value of the first child element object it contains.

Parameters:

passedParentElement - The element that is to have an annotation element added to it

Returns:

An Integer index value of the child element in the parent content array

updateVariablesForModification

```
public void updateVariablesForModification()
```

This method coordinates the updating of a number of attributes to reflect the changes introduced by producing a localised schema that has been derived from another schema. It adds a derivation comment to the top of the schema, updates the namespace of the document to reflect the fact that it has been modified, and updates the version string, again, to reflect the fact that it have been modified.

addDefinitionDocumentIncludeElement

```
public void addDefinitionDocumentIncludeElement()
```

This method inserts a new include element in to the modified schema as a definition document that holds new simpleType declarations from the Application Profile has been created.

updateNamespace

```
public void updateNamespace()
```

This method updates the targetnamespace attribute to reflect that this has been modified (localised) during the application of the Application Profile data. It then updates the default xmlns attribute with the same updated string value.

updateVersionString

```
public void updateVersionString()
```

This method updates the version attribute with a data string to reflect that this has been modified (localised) during the application of the Application Profile data.

getUpdatedFileName

```
public java.lang.String getUpdatedFileName()
```

This method returns a string holding an updated filename for use when the schema is to be written out after modifications have been applied.

Returns:

A string holding the modified schema filename.

getFilenameWithoutExtension

```
public java.lang.String getFilenameWithoutExtension()
```

This method returns a string holding the filename of the schema without the .xsd extension.

Returns:

A string holding the schema filename without the '.xsd' extension.

getName

```
public java.lang.String getName()
```

Returns:

Returns the name.

containsImportOrIncludeStatements

```
public boolean containsImportOrIncludeStatements()
```

This method evaluates the schema to determine if it contains either include or import elements.

Returns:

true if elements found, or false otherwise.

getImportAndIncludeStatements

```
public java.util.Vector getImportAndIncludeStatements()
```

This method returns a Vector containing any include or import elements from this schema.

Returns:

a Vector of elements if any are found, an empty Vector if none are found

org.TELCERT.CRT.STT.schema

Class STTResultantSchema

java.lang.Object

└ org.jdom.Document

└ [org.TELCERT.CRT.STT.schema.STTSchemaSuperclass](#)

└ **org.TELCERT.CRT.STT.schema.STTResultantSchema**

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

public class **STTResultantSchema**

extends [STTSchemaSuperclass](#)

This class provides functionality related to the produced schema as a result of the merge process. It extends the STTSchemaSuperclass class to provide functionality addressing the addition of annotation elements, the insertion of elements within the schema, Principle of Reference Tracing processing and the updating of modified referenced schema strings.

Author:

wluxton

See Also:

[Serialized Form](#)

Constructor Summary

[STTResultantSchema](#)(org.jdom.Document baseDocument)

Method Summary

org.jdom.Element	applyPrincipleOfReferenceTracingProcessing (org.jdom.Element elementForModification, STTBaseSchema baseSchema) This method receives an element for modification, and an STTBaseSchema object.
void	checkAndAddAnnotationElement (org.jdom.Element passedElement) This method receives an element and checks if it contains an annotation element 1 level deep.
static void	checkUsageCompliance (java.lang.String modifyingUsageValue, java.lang.String targetedUsageValue) This static method receives a usage value from the modifying element, and a usage value from the element that is to be modified.
void	insertElementInElement (org.jdom.Element insertionElement, org.jdom.Element targetElement) This method is designed to add a passed in element to a passed in target element.
void	insertElementInElementAtIndex (org.jdom.Element insertionElement, org.jdom.Element targetElement, java.lang.Integer indexValue) This method is designed to add a passed in element to a passed in target element at a specified index position.
void	insertElementWithNameInElement (org.jdom.Element insertionElement, org.jdom.Element targetElement, java.lang.String insertedElementName) This method is designed rename a passed in element to the provided name, and then to add it a passed in target element.

void	insertElementWithNameInElementAtIndex (org.jdom.Element insertionElement, org.jdom.Element targetElement, java.lang.Integer indexValue, java.lang.String insertedElementName) This method is designed designed rename a passed in element to the provided name if given (updating either the 'id' or 'name' attributes if they exist - with no update if they don't), and then to add it to a passed in target element at a specified index position.
boolean	isFirstPRTModification ()
org.jdom.Element	returnXPathElement (STTSchemaSuperclass targetedSchema, org.jdom.Element targetedElement, org.jdom.Element modifyingElement, boolean generateAlertMessage) This method overrides the returnXPathElement method defined in STTSchemaSuperclass.
void	setFirstPRTModification (boolean isFirstPRTModification)
void	updateReferencingSchemaStatements (STTBaseSchema baseSchema) This method iterates through all modified schema and updates the referencing include or import element schemaLocation attribute strings of their parent schema to refer to the modified schema.

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchemaSuperclass](#)

[addDefinitionDocumentIncludeElement](#), [containsImportOrIncludeStatements](#), [getElementWithAttributeValueInElement](#), [getElementWithNameInElement](#), [getElementWithNameOrIDAttribute](#), [getElementWithNameOrIDAttributeInElement](#), [getElementWithReferencingAttribute](#), [getFilenameWithoutExtension](#), [getImportAndIncludeStatements](#), [getIndexValueForFirstElement](#), [getName](#), [getUpdatedFileName](#), [returnXPathElementFromElement](#), [returnXPathElementFromThisSchema](#), [updateNamespace](#), [updateVariablesForModification](#), [updateVersionString](#)

Methods inherited from class org.jdom.Document

[addContent](#), [addContent](#), [addContent](#), [addContent](#), [clone](#), [cloneContent](#), [detachRootElement](#), [equals](#), [getBaseURI](#), [getContent](#), [getContent](#), [getContent](#), [getContentSize](#), [getDescendants](#), [getDescendants](#), [getDocType](#), [getDocument](#), [getParent](#), [getProperty](#), [getRootElement](#), [hashCode](#), [hasRootElement](#), [indexOf](#), [removeContent](#), [removeContent](#), [removeContent](#), [removeContent](#), [setBaseURI](#), [setContent](#), [setContent](#), [setContent](#), [setContent](#), [setDocType](#), [setProperty](#), [setRootElement](#), [toString](#)

Methods inherited from class java.lang.Object

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

STTResultantSchema

```
public STTResultantSchema(org.jdom.Document baseDocument)
```

Method Detail

checkUsageCompliance

```
public static void checkUsageCompliance(java.lang.String modifyingUsageValue,
java.lang.String targetedUsageValue)
```

This static method receives a usage value from the modifying element, and a usage value from the element that is to be modified. If the usage is made less restrictive as a result of the proposed modification i.e: prohibited is changed to optional, a warning message is outputted

Parameters:

modifyingUsageValue - The usage value of the modifying element
targetedUsageValue - The usage value of the targeted element

returnXPathElement

```
public org.jdom.Element returnXPathElement(STTSchemaSuperclass targetedSchema,  
                                             org.jdom.Element targetedElement,  
                                             org.jdom.Element modifyingElement,  
                                             boolean generateAlertMessage)
```

This method overrides the returnXPathElement method defined in STTSchemaSuperclass. It added XPath expression validation checking to ensure that alterations to the resultant schema are targeting the right element for the type of modification being performed. all other functionality is the same. The superclass functionality fetches an Element object from either a STTSchemaSuperclass or an Element object, using an XPath expression to identify the elements. The XPath expression is extracted from the passed in modifying element. It will return an element if found - it will return null if not found or if an error is encountered. If the Xpath string is not relative, and a target element has been passed in, the XPath expression will point to the holding document of that element. When using this method only one of a targetedSchema or a targetedElement should be passed in at a time - the other variable should be null

Parameters:

targetedSchema - The STTSchemaSuperclass object to be targeted by the XPath expression
targetedElement - The Element to be targeted by the XPath expression
modifyingElement - The Element holding the XPath expression
generateAlertMessage - A boolean flag controlling if a 'cannot find' message is generated or not

Returns:

an Element targeted by the XPath expression - null if no element found or if an error is encountered.

insertElementInElement

```
public void insertElementInElement(org.jdom.Element insertionElement,  
                                    org.jdom.Element targetElement)
```

This method is designed to add a passed in element to a passed in target element. The insertion position is not specified. By default it will add it to the top of the element list

Parameters:

insertionElement - The element that is to be added
targetElement - The element that is to have the element added to it

insertElementInElementAtIndex

```
public void insertElementInElementAtIndex(org.jdom.Element insertionElement,  
                                             org.jdom.Element targetElement,  
                                             java.lang.Integer indexValue)
```

This method is designed to add a passed in element to a passed in target element at a specified index position.

Parameters:

insertionElement - The element that is to be added

targetElement - The element that is to have the element added to it
indexValue - The insertion position of the element to add

insertElementWithNameInElement

```
public void insertElementWithNameInElement(org.jdom.Element insertionElement,  
                                           org.jdom.Element targetElement,  
                                           java.lang.String insertedElementName)
```

This method is designed to rename a passed-in element to the provided name, and then to add it to a passed-in target element. The insertion position is not specified. By default, it will add it to the top of the element list.

Parameters:

insertionElement - The element that is to be added
targetElement - The element that is to have the element added to it
insertedElementName - The string used to update the 'id' or 'name' attribute for the insertionElement

insertElementWithNameInElementAtIndex

```
public void insertElementWithNameInElementAtIndex(org.jdom.Element insertionElement,  
                                                  org.jdom.Element targetElement,  
                                                  java.lang.Integer indexValue,  
                                                  java.lang.String insertedElementName)
```

This method is designed to rename a passed-in element to the provided name if given (updating either the 'id' or 'name' attributes if they exist - with no update if they don't), and then to add it to a passed-in target element at a specified index position. If the index position is not specified, it will add it to the front of the element list.

Parameters:

insertionElement - The element that is to be added
targetElement - The element that is to have the insertion element added to it
indexValue - The insertion position of the element to add
insertedElementName - The string used to update the 'id' or 'name' attribute for the insertionElement

checkAndAddAnnotationElement

```
public void checkAndAddAnnotationElement(org.jdom.Element passedElement)
```

This method receives an element and checks if it contains an annotation element 1 level deep. If so, it retrieves the targeted element (via the XPath expression in the parent element if the XPath expression is relative) and then calls `insertElementWithNameInElementAtIndex` to add the annotation element to the element retrieved from the base schema.

Parameters:

passedElement - The element to check for a 1-level deep annotation element

applyPrincipleOfReferenceTracingProcessing

```
public org.jdom.Element applyPrincipleOfReferenceTracingProcessing(org.jdom.  
Element elementForModification,  
                                                                    STTBaseSchema baseSchema)
```

This method receives an element for modification, and an `STTBaseSchema` object. It assesses the element for modification, and returns it unchanged for further processing if there is no `ref` or `type` attribute. If a `ref` or `type` attribute is present the referenced definition element is copied from either the `*base schema*` (to ensure the original condition of the definition element is acted on) or from one of the referenced schema that base schema refers to through `include` or `import` elements (if the element cannot be found in the base schema, and inserted into the targeted schema (either the resultant schema, or the reference schema it was located in) after the original definition element. It is inserted back with a new name derived from the original name + the index position of the parent of the element for modification + the index position of the passed element for modification itself. The `ref / type` attribute of the referencing element is updated to point to the newly copied definition element. Finally, the copied definition element is returned in the place of the passed element for modification, so that any AP defined modifications will be applied to the copied definition element.

Parameters:

`elementForModification` - The element to be assessed for a `ref` or `type` attribute

`baseSchema` - The schema that any referenced definition elements are copied from, or that defines referenced schema

Returns:

The original `elementForModification` if no `ref` or `type` attribute or if there were problems encountered during processing, otherwise the copied definition element.

updateReferencingSchemaStatements

```
public void updateReferencingSchemaStatements(STTBaseSchema baseSchema)
```

This method iterates through all modified schema and updates the referencing `include` or `import` element `schemaLocation` attribute strings of their parent schema to refer to the modified schema. It then updates any `import` or `include` element within the resultant schema that references a schema that has been modified as part of the AP process to refer to the new filename for the modified schema.

Parameters:

`baseSchema` - The schema holding information about all of the modified referenced schemas

isFirstPRTModification

```
public boolean isFirstPRTModification()
```

Returns:

Returns the `isFirstPRTModification`.

setFirstPRTModification

```
public void setFirstPRTModification(boolean isFirstPRTModification)
```

Parameters:

`isFirstPRTModification` - The `isFirstPRTModification` to set.

Serialized Form

Package **org.TELCERT.CRT.STT.schema**

Class [org.TELCERT.CRT.STT.schema.STTAppProfile](#) extends [STTSchemaSuperclass](#) implements Serializable

Serialized Fields

messageVehicle

[STTMessages](#) messageVehicle

Class [org.TELCERT.CRT.STT.schema.STTBaseSchema](#) extends [STTSchema](#) implements Serializable

Serialized Fields

allReferencedSchema

java.util.Vector allReferencedSchema

Class [org.TELCERT.CRT.STT.schema.STTConstraintsDocument](#) extends [org.jdom.Document](#) implements Serializable

Serialized Fields

messageVehicle

[STTMessages](#) messageVehicle

constraintsDocument

org.jdom.Document **constraintsDocument**

appProfile

[STTAppProfile](#) **appProfile**

contentAdded

boolean **contentAdded**

**Class [org.TELCERT.CRT.STT.schema.STTDefinitionSchema](#)
extends org.jdom.Document implements Serializable**

Serialized Fields

messageVehicle

[STTMessages](#) **messageVehicle**

definitionDocument

org.jdom.Document **definitionDocument**

documentName

java.lang.String **documentName**

containsNewDefinitions

boolean `containsNewDefinitions`

Class [org.TELCERT.CRT.STT.schema.STTReferencedSchema](#) extends [STTSchema](#) implements `Serializable`

Serialized Fields

isModified

boolean `isModified`

parentSchemaVector

java.util.Vector `parentSchemaVector`

Class [org.TELCERT.CRT.STT.schema.STTResultantSchema](#) extends [STTSchemaSuperclass](#) implements `Serializable`

Serialized Fields

isFirstPRTModification

boolean `isFirstPRTModification`

Class [org.TELCERT.CRT.STT.schema.STTSchema](#) extends [STTSchemaSuperclass](#) implements `Serializable`

Serialized Fields

referencedSchemaVector

java.util.Vector `referencedSchemaVector`

Class [org.TELCERT.CRT.STT.schema.STTSchemaSuperclass](#)
extends `org.jdom.Document` implements `Serializable`

Serialized Fields

messageVehicle

[STTMessages](#) messageVehicle

name

`java.lang.String` name

org.TELCERT.CRT.STT.utilities

Class STTMessages

java.lang.Object

└ **org.TELCERT.CRT.STT.utilities.STTMessages**

public class **STTMessages**

extends java.lang.Object

This is a singleton class that is designed to hold any information or warning strings that are outputted as a result of STT processing. It also holds the capability to display its messages on the command line

Author:

wluxton

Method Summary

void	addErrorMessage (java.lang.String errorMessage) This method allows the addition of an error message to the errorMessages vector
void	addModificationMessage (java.lang.String modificationMessage) This method allows the addition of a modification message to the modificationMessages vector
void	clearAllMessageLogs () This method clears all message entries held in the errorMessages and conditionMessages vectors
void	clearErrorMessages () This method clears all message entries held in the errorMessages vector
void	clearModificactionMessages () This method clears all message entries held in the conditionMessages vector
java.util.Vector	getErrorMessages ()

static STTMessages	getMessageInstance () This method returns the sole instance of the STTMessages singleton class.
java.util.Vector	getModificationMessages ()
void	printAllMesages () This method prints the all message entries held in the errorMessages and conditionMessages vectors
void	printErrorMesages () This method prints the message entries held in the errorMessages vector
void	printModificationMesages () This method prints the message entries held in the modificationMessages vector

Methods inherited from class java.lang.Object

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

Method Detail

getMessageInstance

```
public static STTMessages getMessageInstance()
```

This method returns the sole instance of the STTMessages singleton class.

Returns:

The singleton instance of the STTMessages class

addErrorMessage

```
public void addErrorMessage(java.lang.String errorMessage)
```

This method allows the addition of an error message to the errorMessages vector

Parameters:

errorMessage -

addModificationMessage

```
public void addModificationMessage(java.lang.  
String modificationMessage)
```

This method allows the addition of a modification message to the `modificationMessages` vector

Parameters:

modificationMessage -

getErrorMessages

```
public java.util.Vector getErrorMessages()
```

Returns:

Returns the `errorMessagees` vector.

getModificationMessages

```
public java.util.Vector getModificationMessages()
```

Returns:

Returns the `modificationMessages` vector.

printErrorMesages

```
public void printErrorMesages()
```

This method prints the message entries held in the `errorMessagees` vector

printModificationMessages

```
public void printModificationMessages()
```

This method prints the message entries held in the `modificationMessages` vector

printAllMessages

```
public void printAllMessages()
```

This method prints the all message entries held in the `errorMessages` and `conditionMessages` vectors

clearModificationMessages

```
public void clearModificationMessages()
```

This method clears all message entries held in the `conditionMessages` vector

clearErrorMessages

```
public void clearErrorMessages()
```

This method clears all message entries held in the `errorMessages` vector

clearAllMessageLogs

```
public void clearAllMessageLogs()
```

This method clears all message entries held in the `errorMessages` and `conditionMessages` vectors

org.TELCERT.CRT.STT.schema

Class STTBaseSchema

java.lang.Object

└ org.jdom.Document

└ [org.TELCERT.CRT.STT.schema.STTSchemaSuperclass](#)

└ [org.TELCERT.CRT.STT.schema.STTSchema](#)

└ **org.TELCERT.CRT.STT.schema.STTBaseSchema**

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

public class **STTBaseSchema**

extends [STTSchema](#)

This class is the type for the base schema. It extends the STTSchema class. It is able to track all of the STTReferencedSchema objects that have been pulled in through import or include statements. It can return these as a Vector, or if a schema name is give, it can return an individual schema. In addition it can also return on those referenced schema that have been modified.

Author:

wluxton

See Also:

[Serialized Form](#)

Constructor Summary

[STTBaseSchema](#)(org.jdom.Document baseDocument)

Method Summary

java.util.Vector	getAllReferencedSchema ()
------------------	---

java.util.Vector	<p><u>getModifiedReferencedSchema</u>()</p> <p>This method returns a Vector of STTReferencedSchema that have been modified during the Application Profile modification process.</p>
<u>STTReferencedSchema</u>	<p><u>getReferenceSchemaByName</u>(java.lang.String modReferredSchemaString)</p> <p>This method receives a schema identifier, processes it to get just the schema name, and returns an STTReferencedSchema if one is found.</p>
java.util.Vector	<p><u>getUnmodifiedReferencedSchema</u>()</p> <p>This method returns a Vector of STTReferencedSchema that have been not been modified during the Application Profile modification process.</p>
void	<p><u>setAllReferencedSchema</u>(java.util.Vector allReferencedSchema)</p>

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchema](#)

[getReferenceSchemaVector](#), [resolveReferencedSchema](#),
[setReferencedSchemaVector](#)

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchemaSuperclass](#)

[addDefinitionDocumentIncludeElement](#),
[containsImportOrIncludeStatements](#),
[getElementWithAttributeValueInElement](#),
[getElementWithNameInElement](#), [getElementWithNameOrIDAttribute](#),
[getElementWithNameOrIDAttributeInElement](#),
[getElementWithReferencingAttribute](#), [getFilenameWithoutExtension](#),
[getImportAndIncludeStatements](#), [getIndexValueForFirstElement](#),
[getName](#), [getUpdatedFileName](#), [returnXPathElementFromElement](#),
[returnXPathElementFromThisSchema](#), [updateNamespace](#),
[updateVariablesForModification](#), [updateVersionString](#)

Methods inherited from class org.jdom.Document

```
addContent, addContent, addContent, addContent, clone,
cloneContent, detachRootElement, equals, getBaseURI, getContent,
getContent, getContent, getContentSize, getDescendants,
getDescendants, getDocType, getDocument, getParent, getProperty,
getRootElement, hashCode, hasRootElement, indexOf, removeContent,
removeContent, removeContent, removeContent, setBaseURI,
setContent, setContent, setContent, setContent, setDocType,
setProperty, setRootElement, toString
```

Methods inherited from class java.lang.Object

```
getClass, notify, notifyAll, wait, wait, wait
```

Constructor Detail

STTBaseSchema

```
public STTBaseSchema(org.jdom.Document baseDocument)
```

Parameters:

baseDocument -

Method Detail

getAllReferencedSchema

```
public java.util.Vector getAllReferencedSchema()
```

Returns:

Returns the allReferencedSchema.

setAllReferencedSchema

```
public void setAllReferencedSchema(java.util.
Vector allReferencedSchema)
```

Parameters:

allReferencedSchema - The allReferencedSchema to set.

getReferencedSchemaByName

```
public STTReferencedSchema getReferencedSchemaByName( java.lang.  
String modReferredSchemaString)
```

This method receives a schema identifier, processes it to get just the schema name, and returns an `STTReferencedSchema` if one is found.

Parameters:

`modReferredSchemaString` - A schema identifying string from the `baseSchema` attribute of a modification element.

Returns:

An `STTReferencedSchema` object if found, `null` if not found

getModifiedReferencedSchema

```
public java.util.Vector getModifiedReferencedSchema()
```

This method returns a `Vector` of `STTReferencedSchema` that have been modified during the Application Profile modification process.

Returns:

A `Vector` of modified `STTReferencedSchema` objects

getUnmodifiedReferencedSchema

```
public java.util.Vector getUnmodifiedReferencedSchema()
```

This method returns a `Vector` of `STTReferencedSchema` that have not been modified during the Application Profile modification process. Only schema that have not been modified will be present in this `Vector`.

Returns:

A `Vector` of unmodified `STTReferencedSchema` objects

org.TELCERT.CRT.STT.schema

Class STTSchema

java.lang.Object

└ org.jdom.Document

└ [org.TELCERT.CRT.STT.schema.STTSchemaSuperclass](#)

└ org.TELCERT.CRT.STT.schema.STTSchema

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

Direct Known Subclasses:

[STTBaseSchema](#), [STTReferencedSchema](#)

public abstract class **STTSchema**

extends [STTSchemaSuperclass](#)

Author:

wluxton This abstract class is the superclass for [STTBaseSchema](#) and [STTReferencedSchema](#). It has the functionality to retrieve referenced schemas in `include` and `import` elements within the schema. These schema are held in a vector of `referencedSchemaVector`. It also knows if it contains either `include` and `import` elements.

See Also:

[Serialized Form](#)

Constructor Summary

[STTSchema](#)(org.jdom.Document baseDocument)

Method Summary

java.util.Vector	getReferencedSchemaVector ()
------------------	--

void	resolveReferencedSchema () This method processes a schema's root element.
void	setReferencedSchemaVector (java.util. Vector referencedSchemaVector)

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchemaSuperclass](#)

[addDefinitionDocumentIncludeElement](#),
[containsImportOrIncludeStatements](#),
[getElementWithAttributeValueInElement](#),
[getElementWithNameInElement](#), [getElementWithNameOrIDAttribute](#),
[getElementWithNameOrIDAttributeInElement](#),
[getElementWithReferencingAttribute](#), [getFilenameWithoutExtension](#),
[getImportAndIncludeStatements](#), [getIndexValueForFirstElement](#),
[getName](#), [getUpdatedFileName](#), [returnXPathElementFromElement](#),
[returnXPathElementFromThisSchema](#), [updateNamespace](#),
[updateVariablesForModification](#), [updateVersionString](#)

Methods inherited from class org.jdom.Document

[addContent](#), [addContent](#), [addContent](#), [addContent](#), [clone](#),
[cloneContent](#), [detachRootElement](#), [equals](#), [getBaseURI](#), [getContent](#),
[getContent](#), [getContent](#), [getContentSize](#), [getDescendants](#),
[getDescendants](#), [getDocType](#), [getDocument](#), [getParent](#), [getProperty](#),
[getRootElement](#), [hashCode](#), [hasRootElement](#), [indexOf](#), [removeContent](#),
[removeContent](#), [removeContent](#), [removeContent](#), [setBaseURI](#),
[setContent](#), [setContent](#), [setContent](#), [setContent](#), [setDocType](#),
[setProperty](#), [setRootElement](#), [toString](#)

Methods inherited from class java.lang.Object

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

STTSchema

```
public STTSchema(org.jdom.Document baseDocument)
```

Method Detail

resolveReferencedSchema

```
public void resolveReferencedSchema()
```

This method processes a schema's root element. It generates a list of included or imported elements. If any are found - it iterates through the list, and attempts to retrieve each referenced schema. For each successfully retrieved schema, it creates a `HashMap` that holds the schema and a string that indicates if the schema has been modified during the resultant schema generation process. Each retrieved schema is parsed for include and import statements, and processed accordingly if any are found, with the referenced schema added to the static `HashMap` `referencedSchemaVector`. If a referenced schema is unable to be located or retrieved, a modification message is created (will not terminate processing) informing the user. If no referenced schema are found in the base schema, the `referencedSchemaVector` `HashMap` is set to null

getReferencedSchemaVector

```
public java.util.Vector getReferencedSchemaVector()
```

Returns:

Returns the `referencedSchemaVector`.

setReferencedSchemaVector

```
public void setReferencedSchemaVector(java.util.  
Vector referencedSchemaVector)
```

Parameters:

`referencedSchemaVector` - The `referencedSchemaVector` to set.

org.TELCERT.CRT.STT.schema

Class STTReferencedSchema

java.lang.Object

└ org.jdom.Document

└ [org.TELCERT.CRT.STT.schema.STTSchemaSuperclass](#)

└ [org.TELCERT.CRT.STT.schema.STTSchema](#)

└ **org.TELCERT.CRT.STT.schema.STTReferencedSchema**

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

public class **STTReferencedSchema**

extends [STTSchema](#)

This class is the type for schema objects pulled in through `import` or `include` statements in the base schema. It extends the `STTSchema` class, and includes functionality to track any schema that directly references it though `import` or `include` statements, whether it has been modified as part of the AP process, and also to update the referencing statements of any parent schema one it has been modified.

Author:

wluxton

See Also:

[Serialized Form](#)

Constructor Summary

[STTReferencedSchema](#)(org.jdom.Document baseDocument,
[STTSchema](#) passedParentSchema)

Method Summary

void	<u>addParentSchema</u> (<u>STTSchema</u> parentSchema) This method allows the addition of parent schema's to the vector holding those schema that reference this schema through an import or include element.
java.util. Vector	<u>getParentSchemaVector</u> ()
<u>STTBaseSchema</u>	<u>getReferencingBaseSchema</u> () This method returns the ultimate base schema that held the original included/imported schema.
boolean	<u>isModified</u> ()
void	<u>setIsModified</u> (boolean isModified) This method sets the isModified variable, and if it is true, will update the namespace and version string of the referenced schema
void	<u>setParentSchemaVector</u> (java.util. Vector newParentSchemaVector)
void	<u>updateParentReferencingSchemaLocationStrings</u> () This method iterates through all of the parent STTReferenceSchema objects that directly reference this schema and changes the the schemaLocation attribute of the appropriate include or import elements to reflect the new filename the modified schema will have when it is printed out at the end of the modification process.

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchema](#)

[getReferencedSchemaVector](#), [resolveReferencedSchema](#),
[setReferencedSchemaVector](#)

Methods inherited from class org.TELCERT.CRT.STT.schema.[STTSchemaSuperclass](#)

[addDefinitionDocumentIncludeElement](#),
[containsImportOrIncludeStatements](#),
[getElementWithAttributeValueInElement](#),
[getElementWithNameInElement](#), [getElementWithNameOrIDAttribute](#),
[getElementWithNameOrIDAttributeInElement](#),
[getElementWithReferencingAttribute](#), [getFilenameWithoutExtension](#),
[getImportAndIncludeStatements](#), [getIndexValueForFirstElement](#),
[getName](#), [getUpdatedFileName](#), [returnXPathElementFromElement](#),
[returnXPathElementFromThisSchema](#), [updateNamespace](#),
[updateVariablesForModification](#), [updateVersionString](#)

Methods inherited from class org.jdom.Document

addContent, addContent, addContent, addContent, clone, cloneContent, detachRootElement, equals, getBaseURI, getContent, getContent, getContent, getContentSize, getDescendants, getDescendants, getDocType, getDocument, getParent, getProperty, getRootElement, hashCode, hasRootElement, indexOf, removeContent, removeContent, removeContent, removeContent, setBaseURI, setContent, setContent, setContent, setContent, setDocType, setProperty, setRootElement, toString

Methods inherited from class java.lang.Object

getClass, notify, notifyAll, wait, wait, wait

Constructor Detail

STTReferencedSchema

```
public STTReferencedSchema(org.jdom.Document baseDocument,
                           STTSchema passedParentSchema)
```

Parameters:

baseDocument -

passedParentSchema -

Method Detail

isModified

```
public boolean isModified()
```

Returns:

Returns the isModified.

setIsModified

```
public void setIsModified(boolean isModified)
```

This method sets the `isModified` variable, and if it is `true`, will update the namespace and version string of the referenced schema

Parameters:

`isModified` - The `isModified` to set.

getParentSchemaVector

```
public java.util.Vector getParentSchemaVector()
```

Returns:

Returns the `newParentSchemaVector`.

setParentSchemaVector

```
public void setParentSchemaVector(java.util.  
Vector newParentSchemaVector)
```

Parameters:

`newParentSchemaVector` - The `parentSchema` to set.

addParentSchema

```
public void addParentSchema(STTSchema parentSchema)
```

This method allows the addition of parent schema's to the vector holding those schema that reference this schema through an `import` or `include` element. Only one schema of name 'a' can be added.

Parameters:

`parentSchema` - The schema object to add as a `parentSchema` entry

getReferencingBaseSchema

```
public STTBaseSchema getReferencingBaseSchema( )
```

This method returns the ultimate base schema that held the original included/imported schema.

Returns:

The base schema at the top of the parent schema heirarchy

updateParentReferencingSchemaLocationStrings

```
public void updateParentReferencingSchemaLocationStrings( )
```

This method iterates through all of the parent `STTReferenceSchema` objects that directly reference this schema and changes the the `schemaLocation` attribute of the appropriate include or import elements to reflect the new filename the modified schema will have when it is printed out at the end of the modification process. It also updates the `isModified` variable to reflect the change. The method then invokes the same changes on the selected parent schema so that the changes are cascaded up to the base schema.

org.TELCERT.CRT.STT.schema

Class STTConstraintsDocument

java.lang.Object

└ org.jdom.Document

└ org.TELCERT.CRT.STT.schema.STTConstraintsDocument

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

```
public class STTConstraintsDocument
```

```
extends org.jdom.Document
```

Author:

wluxton This singleton class holds processing related to the construction of a constraints document that holds conditional elements defined in an application profile as Schematron rules. It will accept a condition element, create the schematron rule and write it out to a jdom Document. Then it will modify the targeted element in the resultant schema to produce an attribute value set that is a merge of those held in the base schema, and those held in the application profile. and a second The conditional element processing was modelled on the approach taken by the Apricot application developed by CNR/ISTI (Institute of Information Science and Technologies (ISTI), an institute of the Italian National Research Council (CNR)) in 2004.

See Also:

[Serialized Form](#)

Method Summary

void	createFixedTextNodeRule (org.jdom.Element modifyingElement, org.jdom.Element targetedElement) This method receives a modification element that is attempting to fix the value of a text node.
------	--

void	<p>generateSchematronRuleFromConditional (org.jdom.Element modifyingElement, org.jdom.Element passedBaseSchemaElement)</p> <p>This method is designed to process passed modification elements that have cnd attributes and their associated condition elements, converting the conditions to Schematron rules that reference the resultant schema.</p>
java.lang.String	<p>generateXPathForElement(java.lang.Object passedNode)</p> <p>This method generates an XPath expression identifying the passedNode in the resultant schema by walking its way back up the tree and appending the namespace prefix and element name to the path string.</p>
static STTConstraintsDocument	<p>getConstraintInstance()</p> <p>This method returns the sole instance of the STTConstraintsDocument singleton class.</p>
org.jdom.Document	<p>getConstraintsDocument()</p> <p>This method returns the constraints document - which holds condition elements from the Application Profile expressed as Schematron rules.</p>
boolean	<p>isContentAdded()</p>

Methods inherited from class org.jdom.Document

addContent, addContent, addContent, addContent, clone, cloneContent, detachRootElement, equals, getBaseURI, getContent, getContent, getContent, getContentSize, getDescendants, getDescendants, getDocType, getDocument, getParent, getProperty, getRootElement, hashCode, hasRootElement, indexOf, removeContent, removeContent, removeContent, removeContent, setBaseURI, setContent, setContent, setContent, setContent, setDocType, setProperty, setRootElement, toString

Methods inherited from class java.lang.Object

getClass, notify, notifyAll, wait, wait, wait

Method Detail

getConstraintInstance

```
public static STTConstraintsDocument getConstraintInstance()
```

This method returns the sole instance of the `STTConstraintsDocument` singleton class.

Returns:

The singleton instance of the `STTConstraintsDocument` class

generateSchematronRuleFromConditional

```
public void generateSchematronRuleFromConditional(org.jdom.  
Element modifyingElement,  
org.jdom.  
Element passedBaseSchemaElement)
```

This method is designed to process passed modification elements that have `cond` attributes and their associated `condition` elements, converting the conditions to Schematron rules that reference the resultant schema. These rules populate the `constraints` document. Once a rule has been created and added to the `constraints` document, the target element's attribute values are merged with attribute values from the modification element in the `mergeBaseAndConditionalAttributeValue` method. The approach taken to create the Schematron rules follows that devised for the Apricot application, which was developed by CNR/ISTI (Institute of Information Science and Technologies (ISTI), an institute of the Italian National Research Council (CNR)) in 2004.

Parameters:

`modifyingElement` - The element modifying the base schema that is constrained by a condition.
`passedBaseSchemaElement` - The base schema element that holds the elements to be assessed.

getConstraintsDocument

```
public org.jdom.Document getConstraintsDocument()
```

This method returns the constraints document - which holds `condition` elements from the Application Profile expressed as Schematron rules.

Returns:

A Jdom Document object holding Schematron rules for condition element validation.

createFixedTextNodeRule

```
public void createFixedTextNodeRule(org.jdom.  
Element modifyingElement,  
                                     org.jdom.  
Element targetedElement)
```

This method receives a modification element that is attempting to fix the value of a text node. This is not permitted by XML schema, so a Schematron rule is generated to allow the validate of the specified text node value. The rule is written to the constraints document.

Parameters:

`modifyingElement` - A modification element that is used as the base for the generated Schematron rule.

`targetedElement` - An element object that is the target of the modification

generateXPathForElement

```
public java.lang.String generateXPathForElement(java.lang.  
Object passedNode)
```

This method generates an XPath expression identifying the `passedNode` in the resultant schema by walking its way back up the tree and appending the namespace prefix and element name to the path string. If the passed Node has siblings of the same name, and index value is generated.

Parameters:

`passedNode` - The object used to generate the XPath expression for.

Returns:

The generated Xpath expression as a `String`

isContentAdded

```
public boolean isContentAdded()
```

Returns:

Returns the contentAdded boolean - indicates whether constraint rules have been added to base document or not.

org.TELCERT.CRT.STT.schema

Class STTDefinitionSchema

java.lang.Object

└ org.jdom.Document

└ org.TELCERT.CRT.STT.schema.STTDefinitionSchema

All Implemented Interfaces:

java.lang.Cloneable, org.jdom.Parent, java.io.Serializable

public class **STTDefinitionSchema**

extends org.jdom.Document

Author:

wluxton This object type is specifically designed to only hold type elements defined in an Application Profile and referenced by the modified, resultant schema. To this end it has functionality that is scoped to creating a new document object, inserting elements within it, and populating include elements for insertion in to modified schema.

See Also:

[Serialized Form](#)

Method Summary

void	addBaseSchemaImportElement (STTBaseSchema passedBaseSchema) This method inserts a new import element in to the definition document that includes the result schema as often defined elements in the AP are extending the base schema elements....
boolean	addNewDefinitionToDocument (org.jdom.Element definitionElement) This method receives an Element object, and adds it to the definition document.

void	<u>clearAddedDefintions()</u> This method removes all of the definition elements, and resets the containsNewDefintions boolean to false ready for a new transformation run.
boolean	<u>containsNewDefinitions()</u>
org.jdom.Document	<u>getDefinitionDocument()</u>
static <u>STTDefinitionSchema</u>	<u>getDefinitionSchemaInstance()</u> This method returns the sole instance of the STTDefinitionSchema singleton class.
static <u>STTDefinitionSchema</u>	<u>getDefinitionSchemaInstance</u> (<u>STTBaseSchema</u> passedBaseSchema) This method returns the sole instance of the STTDefinitionSchema singleton class.
java.lang.String	<u>getDocumentName()</u>
org.jdom.Element	<u>getIncludeElementForDefinitionDocument()</u> This method returns an include Element for insertion in to a referencing schema.

Methods inherited from class org.jdom.Document

addContent, addContent, addContent, addContent, clone, cloneContent, detachRootElement, equals, getBaseURI, getContent, getContent, getContent, getContentSize, getDescendants, getDescendants, getDocType, getDocument, getParent, getProperty, getRootElement, hashCode, hasRootElement, indexOf, removeContent, removeContent, removeContent, removeContent, setBaseURI, setContent, setContent, setContent, setContent, setDocType, setProperty, setRootElement, toString

Methods inherited from class java.lang.Object

getClass, notify, notifyAll, wait, wait, wait

Method Detail

getDefinitionSchemaInstance

```
public static STTDefinitionSchema getDefinitionSchemaInstance()
```

This method returns the sole instance of the STTDefinitionSchema singleton class.

Returns:

The singleton instance of the STTDefinitionSchema class

getDefinitionSchemaInstance

```
public static STTDefinitionSchema getDefinitionSchemaInstance  
(STTBaseSchema passedBaseSchema)
```

This method returns the sole instance of the STTDefinitionSchema singleton class.

Parameters:

passedBaseSchema - The base schema object referenced through an impot statement by the definition document

Returns:

The singleton instance of the STTDefinitionSchema class

addBaseSchemaImportElement

```
public void addBaseSchemaImportElement  
(STTBaseSchema passedBaseSchema)
```

This method inserts a new import element in to the definition document that includes the result schema as often defined elements in the AP are extending the base schema elements... The import namespace value is taken from the targetnamespace attribute if found, or the default namespace. If neither are found, then the filename of the passed document is used . The schemaLocation attribute is taken from the URI of the base schema.

Parameters:

passedBaseSchema - The base schema object referenced through an impot statement by the definition document

addNewDefinitionToDocument

```
public boolean addNewDefinitionToDocument(org.jdom.  
Element definitionElement)
```

This method receives an `Element` object, and adds it to the definition document. If the definition document the new type is to be added to has no name, the name of the AP document holding the new type is used as a base for the definition document name.

Parameters:

`definitionElement` - The `Element` to add

Returns:

`true` if addition of definition element to document was successful, `false` otherwise.

getIncludeElementForDefinitionDocument

```
public org.jdom.Element getIncludeElementForDefinitionDocument()
```

This method returns an include `Element` for insertion in to a referencing schema. The include element points to this document object, which will in the same directory as the modified, referring schema.

Returns:

An include `Element` object.

getDocumentName

```
public java.lang.String getDocumentName()
```

Returns:

Returns the `documentName` `String`.

containsNewDefinitions

```
public boolean containsNewDefinitions()
```

Returns:

Returns the `containsNewDefinitions` `boolean`.

getDefinitionDocument

```
public org.jdom.Document getDefinitionDocument()
```

Returns:

Returns the definitionDocument.

clearAddedDefintions

```
public void clearAddedDefintions()
```

This method removes all of the defintion elements, and resets the containsNewDefintions boolean to false ready for a new transformation run.

org.TELCERT.CRT.STT.utilities

Class STTConstants

java.lang.Object

└ org.TELCERT.CRT.STT.utilities.STTConstants

public class **STTConstants**

extends java.lang.Object

Cross-class constants for the Schema Tranform Tool

Author:

wluxton

Field Summary

static java. lang.String[]	<u>ALLOWABLE_ATTRIBUTE_PROCESS_CONTENTS_VALUES_LIST</u>
static java. lang.String[]	<u>ALLOWABLE_ATTRIBUTE_USE_VALUES_LIST</u>
static java. lang.String	<u>ATTRIBUTE_BASE</u>
static java. lang.String	<u>ATTRIBUTE_BASE_SCHEMA</u>
static java. lang.String	<u>ATTRIBUTE_CHILD_XPATH</u>
static java. lang.String	<u>ATTRIBUTE_CONDITION</u>
static java. lang.String	<u>ATTRIBUTE_DEFAULT</u>
static java. lang.String	<u>ATTRIBUTE_FIXED</u>

static java. lang.String	<u>ATTRIBUTE_ID</u>
static java. lang.String	<u>ATTRIBUTE_MAPPING</u>
static java. lang.String	<u>ATTRIBUTE_MAX_OCCURS</u>
static java. lang.String	<u>ATTRIBUTE_MIN_OCCURS</u>
static java. lang.String	<u>ATTRIBUTE_NAME</u>
static java. lang.String	<u>ATTRIBUTE_NAMESPACE</u>
static java. lang.String	<u>ATTRIBUTE_NEXT_SIBLING</u>
static java. lang.String	<u>ATTRIBUTE_OCCURS_UNBOUNDED</u>
static java. lang.String	<u>ATTRIBUTE_PARENT_XPATH</u>
static java. lang.String	<u>ATTRIBUTE_PROCESS_CONTENTS</u>
static java. lang.String	<u>ATTRIBUTE_PROCESS_CONTENTS_LAX</u>
static java. lang.String	<u>ATTRIBUTE_PROCESS_CONTENTS_SKIP</u>
static java. lang.String	<u>ATTRIBUTE_PROCESS_CONTENTS_STRICT</u>
static java. lang.String[]	<u>ATTRIBUTE_PROPERTIES_VALUES_LIST</u>
static java. lang.String	<u>ATTRIBUTE_REF</u>
static java. lang.String	<u>ATTRIBUTE_SCHEMA_LOCATION</u>
static java. lang.String	<u>ATTRIBUTE_SCHEMA_VERSION</u>

static java. lang.String	<u>ATTRIBUTE_TARGET_NAMESPACE</u>
static java. lang.String	<u>ATTRIBUTE_TYPE</u>
static java. lang.String	<u>ATTRIBUTE_USE</u>
static java. lang.String	<u>ATTRIBUTE_USE_OPTIONAL</u>
static java. lang.String	<u>ATTRIBUTE_USE_PROHIBITED</u>
static java. lang.String	<u>ATTRIBUTE_USE_REQUIRED</u>
static java. lang.String[]	<u>CARDINALITY_VALUES_LIST</u>
static java. lang.String	<u>DERIVED_SCHEMA_NAMESPACE</u>
static boolean	<u>DOCUMENT_VALIDATION</u>
static java. lang.String	<u>ELEMENT_ANNOTATION</u>
static java. lang.String	<u>ELEMENT_CONDITION</u>
static java. lang.String	<u>ELEMENT_DEFINITION_HOLDING_ELEMENT</u>
static java. lang.String	<u>ELEMENT_EXTENSION</u>
static java. lang.String	<u>ELEMENT_IMPORT</u>
static java. lang.String	<u>ELEMENT_INCLUDE</u>
static java. lang.String	<u>ELEMENT_MAPPING_HOLDING_ELEMENT</u>
static java. lang.String	<u>ELEMENT_MODIFICATION</u>

static java. lang.String	<u>ELEMENT_MODIFICATION_HOLDING_ELEMENT</u>
static java. lang.String[]	<u>EXTENSION_VALUES_LIST</u>
static java. lang.String	<u>FILENAME_CONDITION_ONLY</u>
static java. lang.String	<u>FILENAME_CONSTRAINT</u>
static java. lang.String	<u>FILENAME_DEFINITION_DOCUMENT</u>
static java. lang.String	<u>FILENAME_LOCALISED</u>
static java. lang.String	<u>MOD_ATTRIBUTE_EXTENSION</u>
static java. lang.String	<u>MOD_ATTRIBUTE_PROPERTIES</u>
static java. lang.String	<u>MOD_CARDINALITY</u>
static java. lang.String	<u>MOD_ELEMENT_EXTENSION</u>
static java. lang.String	<u>MOD_NEW_ATTRIBUTE_EXTENSION</u>
static java. lang.String	<u>MOD_NEW_ELEMENT_EXTENSION</u>
static java. lang.String	<u>NAMESPACE_XS_XSD</u>
static java. lang.String	<u>NAMESPACE_XSI</u>
static java. lang.String	<u>REFERENCED_SCHEMA_STRING</u>
static java. lang.String	<u>SAX_BUILDER_CLASS</u>
static java. lang.String	<u>TYPE_ANY</u>

static java. lang.String	<u>TYPE_ANY_ATTRIBUTE</u>
static java. lang.String	<u>TYPE_ATTRIBUTE</u>
static java. lang.String	<u>TYPE_COMPLEX_TYPE</u>
static java. lang.String	<u>TYPE_ELEMENT</u>
static java. lang.String	<u>TYPE_SIMPLE_CONTENT</u>
static java. lang.String	<u>TYPE_SIMPLE_TYPE</u>

Constructor Summary

[STTConstants](#) ()

Methods inherited from class java.lang.Object

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

Field Detail

DOCUMENT_VALIDATION

```
public static boolean DOCUMENT_VALIDATION
```

SAX_BUILDER_CLASS

```
public static final java.lang.String SAX_BUILDER_CLASS
```

See Also:

[Constant Field Values](#)

FILENAME_LOCALISED

```
public static final java.lang.String FILENAME_LOCALISED
```

See Also:

[Constant Field Values](#)

FILENAME_CONSTRAINT

```
public static final java.lang.String FILENAME_CONSTRAINT
```

See Also:

[Constant Field Values](#)

FILENAME_CONDITION_ONLY

```
public static final java.lang.String FILENAME_CONDITION_ONLY
```

See Also:

[Constant Field Values](#)

FILENAME_DEFINITION_DOCUMENT

```
public static final java.lang.String FILENAME_DEFINITION_DOCUMENT
```

See Also:

[Constant Field Values](#)

DERIVED_SCHEMA_NAMESPACE

```
public static final java.lang.String DERIVED_SCHEMA_NAMESPACE
```

See Also:

REFERENCED_SCHEMA_STRING

```
public static final java.lang.String REFERENCED_SCHEMA_STRING
```

See Also:

[Constant Field Values](#)

NAMESPACE_XS_XSD

```
public static final java.lang.String NAMESPACE_XS_XSD
```

See Also:

[Constant Field Values](#)

NAMESPACE_XSI

```
public static final java.lang.String NAMESPACE_XSI
```

See Also:

[Constant Field Values](#)

ELEMENT_DEFINITION_HOLDING_ELEMENT

```
public static final java.lang.String  
ELEMENT_DEFINITION_HOLDING_ELEMENT
```

See Also:

[Constant Field Values](#)

ELEMENT_MODIFICATION_HOLDING_ELEMENT

```
public static final java.lang.String  
ELEMENT_MODIFICATION_HOLDING_ELEMENT
```

See Also:

[Constant Field Values](#)

ELEMENT_MAPPING_HOLDING_ELEMENT

```
public static final java.lang.String ELEMENT_MAPPING_HOLDING_ELEMENT
```

See Also:

[Constant Field Values](#)

ELEMENT_MODIFICATION

```
public static final java.lang.String ELEMENT_MODIFICATION
```

See Also:

[Constant Field Values](#)

ELEMENT_CONDITION

```
public static final java.lang.String ELEMENT_CONDITION
```

See Also:

[Constant Field Values](#)

ELEMENT_ANNOTATION

```
public static final java.lang.String ELEMENT_ANNOTATION
```

See Also:

[Constant Field Values](#)

ELEMENT_INCLUDE

```
public static final java.lang.String ELEMENT_INCLUDE
```

See Also:

[Constant Field Values](#)

ELEMENT_IMPORT

```
public static final java.lang.String ELEMENT_IMPORT
```

See Also:

[Constant Field Values](#)

ELEMENT_EXTENSION

```
public static final java.lang.String ELEMENT_EXTENSION
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_TARGET_NAMESPACE

```
public static final java.lang.String ATTRIBUTE_TARGET_NAMESPACE
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_SCHEMA_LOCATION

```
public static final java.lang.String ATTRIBUTE_SCHEMA_LOCATION
```

See Also:

ATTRIBUTE_SCHEMA_VERSION

```
public static final java.lang.String ATTRIBUTE_SCHEMA_VERSION
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_BASE_SCHEMA

```
public static final java.lang.String ATTRIBUTE_BASE_SCHEMA
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_PARENT_XPATH

```
public static final java.lang.String ATTRIBUTE_PARENT_XPATH
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_CHILD_XPATH

```
public static final java.lang.String ATTRIBUTE_CHILD_XPATH
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_NAME

```
public static final java.lang.String ATTRIBUTE_NAME
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_ID

```
public static final java.lang.String ATTRIBUTE_ID
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_REF

```
public static final java.lang.String ATTRIBUTE_REF
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_NEXT_SIBLING

```
public static final java.lang.String ATTRIBUTE_NEXT_SIBLING
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_CONDITION

```
public static final java.lang.String ATTRIBUTE_CONDITION
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_MAX_OCCURS

```
public static final java.lang.String ATTRIBUTE_MAX_OCCURS
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_MIN_OCCURS

```
public static final java.lang.String ATTRIBUTE_MIN_OCCURS
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_TYPE

```
public static final java.lang.String ATTRIBUTE_TYPE
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_USE

```
public static final java.lang.String ATTRIBUTE_USE
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_DEFAULT

```
public static final java.lang.String ATTRIBUTE_DEFAULT
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_FIXED

```
public static final java.lang.String ATTRIBUTE_FIXED
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_MAPPING

```
public static final java.lang.String ATTRIBUTE_MAPPING
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_NAMESPACE

```
public static final java.lang.String ATTRIBUTE_NAMESPACE
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_PROCESS_CONTENTS

```
public static final java.lang.String ATTRIBUTE_PROCESS_CONTENTS
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_OCCURS_UNBOUNDED

```
public static final java.lang.String ATTRIBUTE_OCCURS_UNBOUNDED
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_BASE

```
public static final java.lang.String ATTRIBUTE_BASE
```

See Also:

[Constant Field Values](#)

MOD_CARDINALITY

```
public static final java.lang.String MOD_CARDINALITY
```

See Also:

[Constant Field Values](#)

MOD_ELEMENT_EXTENSION

```
public static final java.lang.String MOD_ELEMENT_EXTENSION
```

See Also:

[Constant Field Values](#)

MOD_ATTRIBUTE_EXTENSION

```
public static final java.lang.String MOD_ATTRIBUTE_EXTENSION
```

See Also:

[Constant Field Values](#)

MOD_ATTRIBUTE_PROPERTIES

```
public static final java.lang.String MOD_ATTRIBUTE_PROPERTIES
```

See Also:

[Constant Field Values](#)

MOD_NEW_ELEMENT_EXTENSION

```
public static final java.lang.String MOD_NEW_ELEMENT_EXTENSION
```

See Also:

[Constant Field Values](#)

MOD_NEW_ATTRIBUTE_EXTENSION

```
public static final java.lang.String MOD_NEW_ATTRIBUTE_EXTENSION
```

See Also:

[Constant Field Values](#)

TYPE_SIMPLE_CONTENT

```
public static final java.lang.String TYPE_SIMPLE_CONTENT
```

See Also:

[Constant Field Values](#)

TYPE_COMPLEX_TYPE

```
public static final java.lang.String TYPE_COMPLEX_TYPE
```

See Also:

[Constant Field Values](#)

TYPE_ATTRIBUTE

```
public static final java.lang.String TYPE_ATTRIBUTE
```

See Also:

[Constant Field Values](#)

TYPE_ANY

```
public static final java.lang.String TYPE_ANY
```

See Also:

[Constant Field Values](#)

TYPE_ANY_ATTRIBUTE

```
public static final java.lang.String TYPE_ANY_ATTRIBUTE
```

See Also:

[Constant Field Values](#)

TYPE_ELEMENT

```
public static final java.lang.String TYPE_ELEMENT
```

See Also:

[Constant Field Values](#)

TYPE_SIMPLE_TYPE

```
public static final java.lang.String TYPE_SIMPLE_TYPE
```

See Also:

[Constant Field Values](#)

CARDINALITY_VALUES_LIST

```
public static final java.lang.String[] CARDINALITY_VALUES_LIST
```

ATTRIBUTE_PROPERTIES_VALUES_LIST

```
public static final java.lang.String[]  
ATTRIBUTE_PROPERTIES_VALUES_LIST
```

ATTRIBUTE_USE_PROHIBITED

```
public static final java.lang.String ATTRIBUTE_USE_PROHIBITED
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_USE_OPTIONAL

```
public static final java.lang.String ATTRIBUTE_USE_OPTIONAL
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_USE_REQUIRED

```
public static final java.lang.String ATTRIBUTE_USE_REQUIRED
```

See Also:

[Constant Field Values](#)

ALLOWABLE_ATTRIBUTE_USE_VALUES_LIST

```
public static final java.lang.String[]  
ALLOWABLE_ATTRIBUTE_USE_VALUES_LIST
```

EXTENSION_VALUES_LIST

```
public static final java.lang.String[] EXTENSION_VALUES_LIST
```

ATTRIBUTE_PROCESS_CONTENTS_LAX

```
public static final java.lang.String ATTRIBUTE_PROCESS_CONTENTS_LAX
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_PROCESS_CONTENTS_STRICT

```
public static final java.lang.String  
ATTRIBUTE_PROCESS_CONTENTS_STRICT
```

See Also:

[Constant Field Values](#)

ATTRIBUTE_PROCESS_CONTENTS_SKIP

```
public static final java.lang.String ATTRIBUTE_PROCESS_CONTENTS_SKIP
```

See Also:

[Constant Field Values](#)

ALLOWABLE_ATTRIBUTE_PROCESS_CONTENTS_VALUES_LIST

```
public static final java.lang.String[]  
ALLOWABLE_ATTRIBUTE_PROCESS_CONTENTS_VALUES_LIST
```

Constructor Detail

STTConstants

```
public STTConstants()
```

Constant Field Values

Contents

- [org.TELCERT.*](#)

org.TELCERT.*

org.TELCERT.CRT.STT.utilities.[STTConstants](#)

public static final java.lang.String	ATTRIBUTE_BASE	"base"
public static final java.lang.String	ATTRIBUTE_BASE_SCHEMA	"baseSchema"
public static final java.lang.String	ATTRIBUTE_CHILD_XPATH	"subelement"
public static final java.lang.String	ATTRIBUTE_CONDITION	"cnd"
public static final java.lang.String	ATTRIBUTE_DEFAULT	"default"
public static final java.lang.String	ATTRIBUTE_FIXED	"fixed"
public static final java.lang.String	ATTRIBUTE_ID	"id"
public static final java.lang.String	ATTRIBUTE_MAPPING	"mapping"
public static final java.lang.String	ATTRIBUTE_MAX_OCCURS	"maxOccurs"
public static final java.lang.String	ATTRIBUTE_MIN_OCCURS	"minOccurs"
public static final java.lang.String	ATTRIBUTE_NAME	"name"
public static final java.lang.String	ATTRIBUTE_NAMESPACE	"namespace"
public static final java.lang.String	ATTRIBUTE_NEXT_SIBLING	"nextSibling"
public static final java.lang.String	ATTRIBUTE_OCCURS_UNBOUNDED	"unbounded"
public static final java.lang.String	ATTRIBUTE_PARENT_XPATH	"element"
public static final java.lang.String	ATTRIBUTE_PROCESS_CONTENTS	"processContents"
public static final java.lang.String	ATTRIBUTE_PROCESS_CONTENTS_LAX	"lax"
public static final java.lang.String	ATTRIBUTE_PROCESS_CONTENTS_SKIP	"skip"
public static final java.lang.String	ATTRIBUTE_PROCESS_CONTENTS_STRICT	"strict"
public static final java.lang.String	ATTRIBUTE_REF	"ref"

public static final java.lang.String	ATTRIBUTE_SCHEMA_LOCATION	"schemaLocation"
public static final java.lang.String	ATTRIBUTE_SCHEMA_VERSION	"version"
public static final java.lang.String	ATTRIBUTE_TARGET_NAMESPACE	"targetNamespace"
public static final java.lang.String	ATTRIBUTE_TYPE	"type"
public static final java.lang.String	ATTRIBUTE_USE	"use"
public static final java.lang.String	ATTRIBUTE_USE_OPTIONAL	"optional"
public static final java.lang.String	ATTRIBUTE_USE_PROHIBITED	"prohibited"
public static final java.lang.String	ATTRIBUTE_USE_REQUIRED	"required"
public static final java.lang.String	DERIVED_SCHEMA_NAMESPACE	"localised_Schema"
public static final java.lang.String	ELEMENT_ANNOTATION	"annotation"
public static final java.lang.String	ELEMENT_CONDITION	"condition"
public static final java.lang.String	ELEMENT_DEFINITION_HOLDING_ELEMENT	"definitions"
public static final java.lang.String	ELEMENT_EXTENSION	"extension"
public static final java.lang.String	ELEMENT_IMPORT	"import"
public static final java.lang.String	ELEMENT_INCLUDE	"include"
public static final java.lang.String	ELEMENT_MAPPING_HOLDING_ELEMENT	"mappings"
public static final java.lang.String	ELEMENT_MODIFICATION	"modification"
public static final java.lang.String	ELEMENT_MODIFICATION_HOLDING_ELEMENT	"modifications"
public static final java.lang.String	FILENAME_CONDITION_ONLY	"_conditionElements.xml"
public static final java.lang.String	FILENAME_CONSTRAINT	"_constraintsDocument.xml"
public static final java.lang.String	FILENAME_DEFINITION_DOCUMENT	"_definition.xsd"
public static final java.lang.String	FILENAME_LOCALISED	"_localised.xsd"
public static final java.lang.String	MOD_ATTRIBUTE_EXTENSION	"attribute_extension"
public static final java.lang.String	MOD_ATTRIBUTE_PROPERTIES	"attribute_properties"
public static final java.lang.String	MOD_CARDINALITY	"cardinality"
public static final java.lang.String	MOD_ELEMENT_EXTENSION	"element_extension"

public static final java.lang.String	MOD_NEW_ATTRIBUTE_EXTENSION	"new_attribute_extension"
public static final java.lang.String	MOD_NEW_ELEMENT_EXTENSION	"new_element_extension"
public static final java.lang.String	NAMESPACE_XS_XSD	"http://www.w3.org/2001/XMLSchema"
public static final java.lang.String	NAMESPACE_XSI	"http://www.w3.org/2001/XMLSchema-instance"
public static final java.lang.String	REFERENCED_SCHEMA_STRING	"baseSchema"
public static final java.lang.String	SAX_BUILDER_CLASS	"org.apache.xerces.parsers.SAXParser"
public static final java.lang.String	TYPE_ANY	":any"
public static final java.lang.String	TYPE_ANY_ATTRIBUTE	"anyAttribute"
public static final java.lang.String	TYPE_ATTRIBUTE	"attribute"
public static final java.lang.String	TYPE_COMPLEX_TYPE	"complexType"
public static final java.lang.String	TYPE_ELEMENT	"element"
public static final java.lang.String	TYPE_SIMPLE_CONTENT	"simpleContent"
public static final java.lang.String	TYPE_SIMPLE_TYPE	"simpleType"

org.TELCERT.CRT.STT.schema

Class STTSchemaProcessor

java.lang.Object

└ `org.TELCERT.CRT.STT.schema.STTSchemaProcessor`

public class **STTSchemaProcessor**

extends java.lang.Object

This class holds all of the modification functionality for the generation of a resultant schema from the merge of a provided Application Profile and a provided Base Schema. It processes the different modification types to process the resultant schema, which it returns to the calling class.

Author:

wluxton

Constructor Summary

[STTSchemaProcessor](#) ()

The default constructor - will create a new instance of the `STTSchemaUtilities` class, and get the `STTMessages` instance for message reporting

Method Summary

void	applyAttributePropertiesModification (org.jdom.Element modifyingElement, org.jdom.Element passedElement) This method changes the attribute properties of the targeted element.
void	applyCardinalityModification (org.jdom.Element modifyingElement, org.jdom.Element passedElement) This method receives an element with modification information, and an element that holds elements that are the modification targets.
void	applyExtensionModification (org.jdom.Element modifyingElement, org.jdom.Element targetElement) This method receives an element that holds modification information and a target element that is to be altered by the information in the modifying element.
void	applyNewExtensionModification (org.jdom.Element modifyingElement, org.jdom.Element targetElement) This method receives an element that holds modification information and a target element that is to be altered by the information in the modifying element.

STTResultantSchema	configureResultantSchema (STTBaseSchema passedBaseSchema, STTAppProfile passedAppProfile, boolean addAnnotations) This method receives a base schema object, an application profile object that holds modification information that is used to alter the base schema to produce a resultant schema, and a boolean that controls whether annotate elements in the application profile should be inserted in the resultant schema or not.
STTAppProfile	getAppProfile ()
STTBaseSchema	getBaseSchema ()
STTConstraintsDocument	getConstraintsDoc ()
STTResultantSchema	getResultantSchema ()
void	processModificationElement (org.jdom.Element modificationElement) This method receives a modification element only - it calls processModificationElement(Element, Element with a null modification element.
void	processModificationElement (org.jdom.Element modificationElement, org.jdom.Element passedBaseSchemaElement) This method receives a modification element and a targetedBaseSchemaElement.
void	setAppProfile (STTAppProfile appProfile)
void	setBaseSchema (STTBaseSchema baseSchema)
void	setConstraintsDoc (STTConstraintsDocument constraintsDoc)
void	setResultantSchema (STTResultantSchema resultantSchema)

Methods inherited from class java.lang.Object

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

Constructor Detail

STTSchemaProcessor

```
public STTSchemaProcessor()
```

The default constructor - will create a new instance of the STTSchemaUtilities class, and get the STTMessages instance for message reporting

Method Detail

configureResultantSchema

```
public STTResultantSchema configureResultantSchema(STTBaseSchema passedBaseSchema,  
                                                STTAppProfile passedAppProfile,  
                                                boolean addAnnotations)
```

This method receives a base schema object, an application profile object that holds modification information that is used to alter the base schema to produce a resultant schema, and a boolean that controls whether `annotate` elements in the application profile should be inserted in the resultant schema or not. The method iterates through the modification elements of the app profile and calls `processModificationElement` on each one to apply the changes. It then returns the resultant schema.

Parameters:

`passedBaseSchema` - The schema that is used as the modification base.

`passedAppProfile` - The application profile that holds the modification elements that are to be applied to the base schema.

`addAnnotations` - A boolean that determines if `annotate` elements should be added to the resultant schema

Returns:

A copy of the the base schema that has had the modifications held in the app profile applied to it.

processModificationElement

```
public void processModificationElement(org.jdom.Element modificationElement)
```

This method receives a modification element only - it calls `processModificationElement(Element, Element)` with a null modification element.

Parameters:

`modificationElement` - The modification element to that is to be processed

processModificationElement

```
public void processModificationElement(org.jdom.Element modificationElement,  
                                       org.jdom.Element passedBaseSchemaElement)
```

This method receives a modification element and a `targetedBaseSchemaElement`. If the passed target element is null, it fetches a target element from the base schema using the XPath expression in the modification element's `element` attribute. If the `passedBaseSchemaElement` is not null, it uses this element as the context for the XPath expression to fetch a target element. Then it gets any children elements of the modification element - it first checks to see if it contains any children modification elements - which will restrict the scope of the applied modifications, then processes any modification sub-elements that exist. Then it processes any other children elements to apply their modifications.

Parameters:

`modificationElement` - The modification element to that is to be processed

applyCardinalityModification

```
public void applyCardinalityModification(org.jdom.Element modifyingElement,  
                                         org.jdom.Element passedElement)
```

This method receives an element with modification information, and an element that holds elements that are the modification targets. It retrieves the target element identified by the XPath expression in the modifying element, and sets the minOccurs and maxOccurs attributes to the values present in the modifying element.

Parameters:

`modifyingElement` - The element holding the XPath expression and the modification information
`passedElement` - The element that holds the element targeted for change by the modification element.

applyAttributePropertiesModification

```
public void applyAttributePropertiesModification(org.jdom.  
Element modifyingElement,  
                                                org.jdom.Element passedElement)
```

This method changes the attribute properties of the targeted element. It can change the type, the use, the default and the fixed attributes of an element. It receives an element with modification information, and an element that holds elements that are the modification targets. It retrieves the target element identified by the XPath expression in the modifying element, and sets the appropriate attributes to the values present in the modifying element. If the use attribute is changed to relax usage restrictions, a warning message is outputted.

Parameters:

`modifyingElement` - The element holding the XPath expression and the modification information
`passedElement` - The element that holds the element targeted for change by the modification element.

applyExtensionModification

```
public void applyExtensionModification(org.jdom.Element modifyingElement,  
                                       org.jdom.Element targetElement)
```

This method receives an element that holds modification information and a target element that is to be altered by the information in the modifying element. It can apply changes to existing `xs:any` and `xs:anyAttribute` elements in the target document. It checks the modification type is `element_extension` before assessing if minOccurs and maxOccur attribute modifications are to be applied.

Parameters:

`modifyingElement` - The element holding the XPath expression and the modification information
`targetElement` - The element that holds the element targeted for change by the modification element.

applyNewExtensionModification

```
public void applyNewExtensionModification(org.jdom.Element modifyingElement,  
                                           org.jdom.Element targetElement)
```

This method receives an element that holds modification information and a target element that is to be altered by the information in the modifying element. It will add in a new `xs:any` or `xs:anyAttribute` element to the targeted element, depending on the passed in modification element, in front of a specified element if the `nextSibling` attribute is populated, or at the end if there is no populated `nextSibling` attribute, or the value cannot be resolved. The added elements will be populated with data from the modification element.

Parameters:

`modifyingElement` - The element holding the XPath expression and the modification information
`targetElement` - The element that holds the element targeted for change by the modification element.

getResultantSchema

```
public STTResultantSchema getResultantSchema()
```

setResultantSchema

```
public void setResultantSchema(STTResultantSchema resultantSchema)
```

getAppProfile

```
public STTAppProfile getAppProfile()
```

setAppProfile

```
public void setAppProfile(STTAppProfile appProfile)
```

getBaseSchema

```
public STTBaseSchema getBaseSchema()
```

setBaseSchema

```
public void setBaseSchema(STTBaseSchema baseSchema)
```

getConstraintsDoc

```
public STTConstraintsDocument getConstraintsDoc()
```

setConstraintsDoc

```
public void setConstraintsDoc(STTConstraintsDocument constraintsDoc)
```

org.TELCERT.CRT.STT

Class STTTransformer

java.lang.Object

└ org.TELCERT.CRT.STT.STTTransformer

public class **STTTransformer**

extends java.lang.Object

This class holds the processing functionality for validating input arguments, executing the AP/base schema merge process and outputting the resultant schema and condition-related documents.

Author:

wluxton

Constructor Summary

[STTTransformer](#) ()

Method Summary

void	clearAllMessages ()
	This method clears all message entries held in the errorMessages and conditionMessages vectors
java.util.Vector	getAllMessages ()
	This method returns a Vector of all modification and error messages generated during processing.
java.util.Vector	getErrorMessages ()
	This method returns a Vector of error messages generated during processing.
java.util.Vector	getModificationMessages ()
	This method returns a Vector of modification messages generated during processing.

boolean	<pre>processAPMerge(java.lang.String baseSchemaLocation, java.lang.String appProfileLocation, java.lang.String outputDirectoryLocationString, boolean addAnnotate, boolean createConditionOnlyDocument)</pre> <p>This method receives a number of input and output file locations and two booleans.</p>
---------	--

Methods inherited from class java.lang.Object

`equals`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Constructor Detail

STTTransformer

```
public STTTransformer()
```

Method Detail

processAPMerge

```
public boolean processAPMerge(java.lang.String baseSchemaLocation,
                               java.lang.String appProfileLocation,
                               java.lang.String outputDirectoryLocationString,
                               boolean addAnnotate,
                               boolean createConditionOnlyDocument)
```

This method receives a number of input and output file locations and two booleans. The two input file locations define a base schema definition file (which must end in .xsd), and a Application Profile location. The output location defines a directory location for any generated output files - the resultant, modified schema file (also an .xsd file), which is directly derived from the base schema, any modified referenced schema (identified through `import` and `include` statements in the base schema, and any generated conditional documents. It also requires two additional boolean arguments to control if annotation elements in the application profile are to be added to the resultant schema, and if a separate document holding only condition elements extracted from the application profile is to be created if condition elements are present, in addition to the standard constraints document.

Parameters:

`baseSchemaLocation` - The String location of the base schema
`appProfileLocation` - The String location of the application profile
`outputDirectoryLocationString` - The String location of the output directory
`addAnnotate` - A boolean determining if annoate elements in the AP are to be added to the resultant schema
`createConditionOnlyDocument` - A boolean determining if a seperate condition only document is to be created if conditional elements are present in the AP

Returns:

true if no errors were encountered during processing, false if errors were encountered

getErrorMessages

```
public java.util.Vector getErrorMessages()
```

This method returns a Vector of error messages generated during processing.

Returns:

A Vector of error messages

getModificationMessages

```
public java.util.Vector getModificationMessages()
```

This method returns a Vector of modification messages generated during processing.

Returns:

A Vector of modification messages

getAllMessages

```
public java.util.Vector getAllMessages()
```

This method returns a Vector of all modification and error messages generated during processing.

Returns:

clearAllMessages

```
public void clearAllMessages()
```

This method clears all message entries held in the `errorMessages` and `conditionMessages` vectors

org.TELCERT.CRT.STT.utilities

Class STTUtilities

java.lang.Object

└─ org.TELCERT.CRT.STT.utilities.STTUtilities

public class **STTUtilities**

extends java.lang.Object

This is a utility class designed to hold useful miscellaneous methods to allow separation between the controller and the view. It implements the methods as class methods

Author:

wluxton

Constructor Summary

[STTUtilities\(\)](#)

Method Summary

static boolean	checkXSDType (java.lang.String locationString) This method evaluates the file extension of the passed file location string - returns true if the passed string ends in 'xsd', false if it doesn't.
static org.jdom.Document	createJDOMDocumentFromLocation (java.lang.String locationString) This method receives a file location string and converts it to a JDOM document.
static void	printXMLOutput (java.lang.String prefixString, org.jdom.Document printItem) This method formats and displays a passed Document object using System.out.println.

static void	printXMLOutput (java.lang.String prefixString, org.jdom.Element printItem) This method formats and displays a passed Element object using System.out.println.
static java.io.File	validateOutputFileLocation (java.lang.String outputFileLocationString, java.lang.String resultantFilename) This method validates a passed directory path string and produces a File object if the path is valid.
static void	writeDocumentToLocation (org.jdom.Document docToWrite, java.lang.String outputFileName, java.io.File fileLocation, boolean shouldRetainSpaces) This method is used to write out JDOM documents in a pretty (indented) format to a given location, using the provided outputFileName string as the file name.

Methods inherited from class java.lang.Object

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

Constructor Detail

STTUtilities

```
public STTUtilities()
```

Method Detail

createJDOMDocumentFromLocation

```
public static org.jdom.Document createJDOMDocumentFromLocation(java.lang.String locationString)
```

This method receives a file location string and converts it to a JDOM document. All processing and exception error messages are placed in the the STTMessages instance.

Parameters:

locationString -

Returns:

The created JDOM document, or null if errors were encountered in the creation process

boolean shouldRetainSpaces)

This method is used to write out JDOM documents in a pretty (indented) format to a given location, using the provided `outputFileName` string as the file name. It will produce an error message which is written to the `errorMessage` vector of the `STTMessages` instance if an error was encountered.

Parameters:

`docToWrite` - The JDOM document to write out.
`outputFileName` - The string used for the filename
`fileLocation` - The file location to write to.
`shouldRetainSpaces` - A boolean controlling if the original spacing should be retained. If `false` prettyPrining is enabled

printXMLOutput

```
public static void printXMLOutput(java.lang.String prefixString,  
                                   org.jdom.Document printItem)
```

This method formats and displays a passed Document object using `System.out.println`.

Parameters:

`prefixString` - An explanatory string that appears before the item to print
`printItem` - The Document object to format and print to `System.out.println`

printXMLOutput

```
public static void printXMLOutput(java.lang.String prefixString,  
                                   org.jdom.Element printItem)
```

This method formats and displays a passed Element object using `System.out.println`.

Parameters:

`prefixString` - An explanatory string that appears before the item to print
`printItem` - The Element object to format and print to `System.out.println`

org.TELCERT.CRT.STT.utilities

Class STTXMLOutputter

java.lang.Object

└ org.jdom.output.XMLOutputter

└ org.TELCERT.CRT.STT.utilities.STTXMLOutputter

All Implemented Interfaces:

java.lang.Cloneable

public class **STTXMLOutputter**

extends org.jdom.output.XMLOutputter

Author:

wluxton This class extends the Jdom XMLOutputter class to allow the addition of automatic new line entry afer the addition of comments.

Constructor Summary

[STTXMLOutputter](#)()

This will create an STTXMLOutputter with the default format

[STTXMLOutputter](#)(org.jdom.output.Format arg0)

This will create an STTXMLOutputter with the specified format

Methods inherited from class org.jdom.output.XMLOutputter

clone, escapeAttributeEntities, escapeElementEntities, getFormat, output, output, output, output, output, output, output, output, output, output, output, output, outputElementContent, outputElementContent, outputString, outputString, outputString, outputString, outputString, outputString, outputString, outputString, outputString, setFormat, toString

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

STTXMLOutputter

```
public STTXMLOutputter()
```

This will create an `STTXMLOutputter` with the default format

STTXMLOutputter

```
public STTXMLOutputter(org.jdom.output.Format arg0)
```

This will create an `STTXMLOutputter` with the specified format

Parameters:

`arg0` - The format that is to be used by the outputter

org.TELCERT.CRT.STT

Class SchemaTransformTool

java.lang.Object

└ org.TELCERT.CRT.STT.SchemaTransformTool

public class **SchemaTransformTool**

extends java.lang.Object

This is a simple command-line front-end for the Schema Transform Tool. Functionality is held in the STTTransformer class.

Author:

wluxton

Constructor Summary

[SchemaTransformTool](#) ()

Method Summary

static void	main (java.lang.String[] args)
-------------	--

This method receives an array of arguments, checks that the correct number has been received, and converts the two boolean string values to boolean primitives before invoking the STTTransformer.processAPMerge() method.

Methods inherited from class java.lang.Object

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

Constructor Detail

SchemaTransformTool

```
public SchemaTransformTool()
```

Method Detail

main

```
public static void main(java.lang.String[] args)
```

This method receives an array of arguments, checks that the correct number has been received, and converts the two boolean string values to boolean primitives before invoking the `STTTransformer.processAPMerge()` method. This is the method that is responsible for the actual application of the Application profile modifications against a base schema. The argument array is composed of two input file locations (identifying a base schema definition location - must end in `.xsd`, and a `Application Profile()` location) and an output directory location for the output files - the resultant, modified schema file (also an `.xsd` file), and any conditional documents. It also holds two additional boolean arguments to control if annotation elements in the application profile are to be added to the resultant schema, and if a separate document holding only condition elements extracted from the application profile is to be created if condition elements are present, in addition to the standard constraints document.

Parameters:

`args` - BaseSchemaURL ApplicationProfileURL output file directory location.
Boolean to control addition of annotate comments in AP to resultant schema. Boolean to control if separate condition only file is generated if conditional elements present in application profile.

[org.TELCERT.CRT.STT](#)

Classes

[SchemaTransformTool](#)

[STTTransformer](#)

Package org.TELCERT.CRT.STT

Class Summary	
<u>SchemaTransformTool</u>	This is a simple command-line front-end for the Schema Transform Tool.
<u>STTTransformer</u>	This class holds the processing functionality for validating input arguments, executing the AP/base schema merge process and outputting the resultant schema and condition-related documents.

org.TELCERT.CRT.STT.schema

Classes

[STTAppProfile](#)

[STTBaseSchema](#)

[STTConstraintsDocument](#)

[STTDefinitionSchema](#)

[STTReferencedSchema](#)

[STTResultantSchema](#)

[STTSchema](#)

[STTSchemaProcessor](#)

[STTSchemaSuperclass](#)

Package org.TELCERT.CRT.STT.schema

Class Summary	
<u>STTAppProfile</u>	This class is the type for the resultant schema.
<u>STTBaseSchema</u>	This class is the type for the base schema.
<u>STTConstraintsDocument</u>	
<u>STTDefinitionSchema</u>	
<u>STTReferencedSchema</u>	This class is the type for schema objects pulled in through <code>import</code> or <code>include</code> statements in the base schema.
<u>STTResultantSchema</u>	This class provides functionality related to the produced schema as a result of the merge process.
<u>STTSchema</u>	
<u>STTSchemaProcessor</u>	This class holds all of the modification functionality for the generation of a resultant schema from the merge of a provided Application Profile and a provided Base Schema.
<u>STTSchemaSuperclass</u>	This abstract class provides functionality for <code>STTSchemaSuperclass</code> objects.

[org.TELCERT.CRT.STT.utilities](#)

Classes

[STTConstants](#)

[STTMessages](#)

[STTUtilities](#)

[STTXMLOutputter](#)

Package org.TELCERT.CRT.STT.utilities

Class Summary

<u>STTConstants</u>	Cross-class constants for the Schema Tranform Tool
<u>STTMessages</u>	This is a singleton class that is designed to hold any information or warning strings that are outputted as a result of STT processing.
<u>STTUtilities</u>	This is a utility class designed to hold useful miscellaneous methods to allow seperation between the controller and the view.
<u>STTXMLOutputter</u>	
