

Contents

- [Generated Methods](#)
 - [Notation](#)
 - [Default Methods](#)
 - [ID Element](#)
 - [Property Element](#)
 - [ManyToOne Element](#)
 - [OneToMany Element](#)
 - [ManyToManyElement](#)

Generated Methods

These are the methods that are available for a given TransferObject, depending on its configuration.

Notation

The following notation is used in this document.

Notation	Description
<i>property</i>	An item in italics denotes a XML configuration element, such as <i>id</i> , or <i>property</i> , or <i>manytoone</i>
[@name]	The value of that @ named attribute, i.e in this instance it would be the value of the 'name' attribute.

For example, if I had the following in my xml:

```
<property name="Foo" column="foo" />
```

The notation of `getproperty[@name]` would translate to `getFoo`, as it is the attribute value of *name*, in the *property* element which corresponds to `property[@name]`

Default Methods

TransferObject	Name	Arguments	Returns
Object	getClassName	none	string: The class name of the TransferObject as defined by the config file.
	equalsTransfer	transfer.com.TransferObject : the TransferObject to check equality with	boolean : Whether or not two TransferObjects have the same class, and the same primary key.
	sameTransfer	transfer.com.TransferObject : the TransferObject to check equality with	boolean : Whether or not two TransferObjects are the same object.
	clone	none	transfer.com.TransferObject : A deep clone of this object, that is outside of the Transfer cache.
	getIsDirty	none	boolean : Whether or not the data on this object as changed since it was last committed to the database.
	getIsPersisted	none	boolean : Whether or not this TransferObject has been committed to the database.
	getIsProxy	none	boolean: whether or not this object is a TransferObject Proxy object.
	getIsLoaded	none	boolean: whether or not the underlying object is loaded. For non-Proxy objects, this will always be true. For TransferObject Proxies , it depends on if it has been loaded.

	getLoadedObject	none	transfer.com.TransferObject : For non-Proxy objects, returns <i>this</i> . For TransferObject Proxies , it loads the Proxied object, and returns it.
--	-----------------	------	--

ID Element

TransferObject	Name	Arguments	Returns
Object	setid[@name]	<i>id[@type]</i> : The id to set it to, of the type specified by the attribute 'type'	void : This will generally only be used by Transfer, however if the ID is set before the TransferObject is create()'ed, it will be inserted with the set ID, overwriting any generation or retrieval of the ID.
	getid[@name]	none	<i>id[@type]</i> : The primary key id, of the type specified by the attribute 'type'. If the transferObject has yet to inserted into the database it will return the configured null value for its type.

Property Element

TransferObject	Name	Arguments	Returns
Object	setProperty[@name] (If <i>property[@set]</i> == 'true')	<i>property[@type]</i> : The id to set it to, of the type specified by the attribute 'type'	void
	getProperty[@name] (If <i>property[@get]</i> == 'true')	none	<i>property[@type]</i> : The property to set, of the type specified by the attribute 'type'.
	getProperty[@name]IsNull (If <i>property[@nullable]</i> == 'true')	none	boolean : if the property set is equal to the null value specified in configuration.
	setProperty[@name]Null (If <i>property[@nullable]</i> == 'true')	none	void : Sets the value to the configured NULL value.

ManyToOne Element

TransferObject	Name	Arguments	Returns
Object	setmanytoone[@name]	transfer.com.TransferObject : A TransferObject of the class defined in the <i>link[@to]</i> child element.	void
	getmanytoone[@name]	none	transfer.com.TransferObject : The TransferObject that is currently set.
	removemanytoone[@name]	none	void : Removes the set ManyToOne Child. This will cause Transfer to use NULL as the value for the foreign key in the database.
	hasmanytoone[@name]	none	boolean : Whether it as a ManyTOOne set or not

OneToMany Element

TransferObject	Name	Arguments	Returns
----------------	------	-----------	---------

Object	<code>getonetomany[@name]</code> (if <code>collection[@type] == 'array'</code>)	numeric : index of the array to get an item at.	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>
	<code>getonetomany[@name]</code> (if <code>collection[@type] == 'struct'</code>)	key : The unique key the TransferObject is stored under.	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>
	<code>getonetomany[@name]Array</code> (if <code>collection[@type] == 'array'</code>)	none	array : a shallow copy of the array of child TransferObjects.
	<code>getonetomany[@name]Struct</code> (if <code>collection[@type] == 'struct'</code>)	none	struct : a shallow copy of the struct of child TransferObjects.
	<code>getonetomany[@name]Iterator</code>	none	java.util.Iterator : creates a Iterator on a shallow copy of the collection. If <code>collection[@type] == 'array'</code> then the items will be sorted by the specified order, if it has one.
	<code>sortonetomany[@name]</code> (if <code>collection[@type] == 'array'</code>)	none	void : Performs a software sort on the collection. Usually used after multiple <code>setParentobject[@name]()</code> statements.
	<code>containsonetomany[@name]</code>	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>	boolean : whether or not the collection contains this object.
	<code>findonetomany[@name]</code>	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>	(if <code>collection[@type] == 'array'</code>) numeric : The index of the object in the array (if <code>collection[@type] == 'struct'</code>) string : the key the object is stored under.
Child	<code>setParentobject[@name]</code>	transfer.com.TransferObject : TransferObject of the type of the Object.	void : This will add the Child to the Parent.
	<code>getParentobject[@name]</code>	none	transfer.com.TransferObject : The TransferObject that is the parent.
	<code>removeParentobject[@name]</code>	none	void : Sets the object to having no parent, and removes the Child from the Parent. This will cause Transfer to use NULL as the value for the foreign key in the database.
	<code>hasParentobject[@name]</code>	none	boolean : If this TransferObject has a parent or not

ManyToManyElement

TransferObject	Name	Arguments	Returns
	<code>addmanytomany[@name]</code>	transfer.com.TransferObject : The TransferObject of type defined by <code>link[@to]</code> to add to the collection	void
	<code>getmanytomany[@name]</code> (if <code>collection[@type] == 'array'</code>)	numeric : index of the array to get an item at.	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>
	<code>getmanytomany[@name]</code> (if <code>collection[@type] == 'struct'</code>)	key : The unique key the TransferObject is stored under.	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>

Object	<code>removemanytomany[@name]</code>	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code> to be removed from the collection.	void
	<code>clearmanytomany[@name]</code>	none	void : Removes all children from the collection.
	<code>sortmanytomany[@name]</code> (if <code>collection[@type] == 'array'</code>)	none	void : Performs a software sort on the collection. Usually used after multiple <code>addcollection@name</code> statements.
	<code>getmanytomany[@name]Array</code> (if <code>collection[@type] == 'array'</code>)	none	array : a shallow copy of the array of child TransferObjects.
	<code>getmanytomany[@name]Struct</code> (if <code>collection[@type] == 'struct'</code>)	none	struct : a shallow copy of the struct of child TransferObjects.
	<code>getmanytomany[@name]Iterator</code>	none	java.util.Iterator : creates a Iterator on a shallow copy of the collection. If <code>collection[@type] == 'array'</code> then the items will be sorted by the specified order, if it has one.
	<code>containsmanytomany[@name]</code>	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>	boolean : whether or not the collection contains this object.
	<code>findmanytomany[@name]</code>	transfer.com.TransferObject : A TransferObject of type defined by <code>link[@to]</code>	(if <code>collection[@type] == 'array'</code>) numeric : The index of the object in the array (if <code>collection[@type] == 'struct'</code>) string : the key the object is stored under.