NSValueTransformer Class Reference

Cocoa > Data Management

2007-07-24

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Simultaneously published in the United States and Canada.

Contents

NSValueTransformer Class Reference

Inherits from	NSObject
Conforms to	NSObject (NSObject)
Framework	/System/Library/Frameworks/Foundation.framework
Availability	Available in Mac OS X v10.3 and later.
Companion guide	Value Transformer Programming Guide
Declared in	NSValueTransformer.h
Related sample code	BindingsJoystick CustomAtomicStoreSubclass NewsReader RGB ValueTransformers TemperatureTester

Overview

`NSValueTransformer` is an abstract class that is used by the Cocoa Bindings technology to transform values from one representation to another.

An application creates a subclass of `NSValueTransformer`, overriding the necessary methods to provide the required custom transformation.

Example

A relatively trivial value transformer takes an object of type `id` and returns a string based on the object's class type. This transformer is not reversible as it's probably unreasonable to transform a class name into an object. The value transformer class you write to accomplish this simple task could look like:

```
@interface ClassNameTransformer: NSValueTransformer {}
@end
@implementation ClassNameTransformer
+ (Class)transformedValueClass { return [NSString class]; }
+ (BOOL)allowsReverseTransformation { return NO; }
- (id)transformedValue:(id)value {
    return (value == nil) ? nil : NSStringFromClass([value class]);
}
@end
```

Tasks

Using Name-based Registry

`+ setValueTransformer:forName:` (page 7)
Registers the value transformer a given transformer with a given identifier.

`+ valueTransformerForName:` (page 8)
Returns the value transformer identified by a given identifier.

`+ valueTransformerNames` (page 8)
Returns an array of all the registered value transformers.

Getting Information About a Transformer

`+ allowsReverseTransformation` (page 6)
Returns a Boolean value that indicates whether the receiver can reverse a transformation.

`+ transformedValueClass` (page 7)
Returns the class of the value returned by the receiver for a forward transformation.

Using Transformers

`- transformedValue:` (page 9)
Returns the result of transforming a given value.

`- reverseTransformedValue:` (page 9)
Returns the result of the reverse transformation of a given value.

Class Methods

allowsReverseTransformation

Returns a Boolean value that indicates whether the receiver can reverse a transformation.

```
+ (BOOL)allowsReverseTransformation
```

Return Value
`YES` if the receiver supports reverse value transformations, otherwise `NO`.

The default is `NO`.

Discussion
A subclass should override this method to return `YES` if it supports reverse value transformations.

Availability
Available in Mac OS X v10.3 and later.

Related Sample Code
CoreRecipes
DerivedProperty

Declared In
NSValueTransformer.h

setValueTransformer:forName:

Registers the value transformer a given transformer with a given identifier.

```
+ (void)setValueTransformer:(NSValueTransformer *)transformer forName:(NSString
    *)name
```

Parameters

transformer
The transformer to register.

name
The name for *transformer*.

Availability
Available in Mac OS X v10.3 and later.

See Also
+ valueTransformerForName: (page 8)

Related Sample Code
CoreRecipes
GridCalendar
NewsReader
RGB ValueTransformers
TemperatureTester

Declared In
NSValueTransformer.h

transformedValueClass

Returns the class of the value returned by the receiver for a forward transformation.

```
+ (Class)transformedValueClass
```

Return Value
The class of the value returned by the receiver for a forward transformation.

Discussion
A subclass should override this method to return the appropriate class.

Availability
Available in Mac OS X v10.3 and later.

Related Sample Code
DerivedProperty
GridCalendar
RGB ValueTransformers
StickiesExample

Declared In
`NSValueTransformer.h`

valueTransformerForName:

Returns the value transformer identified by a given identifier.

```
+ (NSValueTransformer *)valueTransformerForName:(NSString *)name
```

Parameters

name

The transformer identifier.

Return Value
The value transformer identified by *name* in the shared registry, or `nil` if not found.

Discussion
If `valueTransformerForName:` does not find a registered transformer instance for *name*, it will attempt to find a class with the specified name. If a corresponding class is found an instance will be created and initialized using its `init:` method and then automatically registered with `name`.

Availability
Available in Mac OS X v10.3 and later.

See Also
`+ setValueTransformer:forName:` (page 7)

Related Sample Code
BindingsJoystick
RGB ValueTransformers
TemperatureTester

Declared In
`NSValueTransformer.h`

valueTransformerNames

Returns an array of all the registered value transformers.

```
+ (NSArray *)valueTransformerNames
```

Return Value
An array of all the registered value transformers.

Availability
Available in Mac OS X v10.3 and later.

Declared In
NSValueTransformer.h

Instance Methods

reverseTransformedValue:

Returns the result of the reverse transformation of a given value.

```
- (id)reverseTransformedValue:(id)value
```

Parameters

value

The value to reverse transform.

Return Value
The reverse transformation of *value*.

Discussion
The default implementation raises an exception if `allowsReverseTransformation` (page 6) returns `NO`; otherwise it will invoke `transformedValue:` (page 9) with *value*.

A subclass should override this method if they require a reverse transformation that is not the same as simply reapplying the original transform (as would be the case with negation, for example). For example, if a value transformer converts a value in Fahrenheit to Celsius, this method would converts a value from Celsius to Fahrenheit.

Availability
Available in Mac OS X v10.3 and later.

See Also
`- transformedValue:` (page 9)

Related Sample Code
TemperatureTester

Declared In
NSValueTransformer.h

transformedValue:

Returns the result of transforming a given value.

```
- (id)transformedValue:(id)value
```

Parameters

value

The value to transform.

Return Value
The result of transforming *value*.

The default implementation simply returns *value*.

Discussion
A subclass should override this method to transform and return an object based on *value*.

Availability
Available in Mac OS X v10.3 and later.

See Also
- reverseTransformedValue: (page 9)

Related Sample Code
BindingsJoystick
CoreRecipes
RGB ValueTransformers
TemperatureTester

Declared In
NSValueTransformer.h

Constants

Named Value Transformers

The following named value transformers are defined by NSValueTransformer:

```
NSString * const NSNegateBooleanTransformerName;
NSString * const NSIsNilTransformerName ;
NSString * const NSIsNotNilTransformerName ;
NSString * const NSUnarchiveFromDataTransformerName ;
NSString * const NSKeyedUnarchiveFromDataTransformerName ;
```

Constants

NSNegateBooleanTransformerName

This value transformer negates a boolean value, transforming YES to NO and NO to YES.

This transformer is reversible.

Available in Mac OS X v10.3 and later.

Declared in NSValueTransformer.h.

NSIsNilTransformerName

This value transformer returns YES if the value is nil.

This transformer is not reversible.

Available in Mac OS X v10.3 and later.

Declared in NSValueTransformer.h.

`NSIsNotNilTransformerName`

This value transformer returns `YES` if the value is non-`nil`.

This transformer is not reversible.

Available in Mac OS X v10.3 and later.

Declared in `NSValueTransformer.h`.

`NSUnarchiveFromDataTransformerName`

This value transformer returns an object created by attempting to unarchive the data in the `NSData` object passed as the value.

The reverse transformation returns an `NSData` instance created by archiving the value. The archived object must implement the `NSCoding` protocol using sequential archiving in order to be unarchived and archived with this transformer.

Available in Mac OS X v10.3 and later.

Declared in `NSValueTransformer.h`.

`NSKeyedUnarchiveFromDataTransformerName`

This value transformer returns an object created by attempting to unarchive the data in the `NSData` object passed as the value. The archived object must be created using keyed archiving in order to be unarchived and archived with this transformer.

The reverse transformation returns an `NSData` instance created by archiving the value using keyed archiving. The archived object must implement the `NSCoding` protocol using keyed archiving in order to be unarchived and archived with this transformer.

Available in Mac OS X v10.5 and later.

Declared in `NSValueTransformer.h`.

Declared In

`NSValueTransformer.h`

Document Revision History

This table describes the changes to *NSValueTransformer Class Reference*.

Date	Notes
2007-07-24	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

Document Revision History

Index