NSAttributeDescription Class Reference

Cocoa > Objective-C Language

2007-10-31

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

Simultaneously published in the United States and Canada.

Contents

NSAttributeDescription Class Reference

Inherits from	NSPropertyDescription : NSObject
Conforms to	NSCoding (NSPropertyDescription) NSCopying (NSPropertyDescription) NSObject (NSObject)
Framework	/System/Library/Frameworks/CoreData.framework
Availability	Available in Mac OS X v10.4 and later.
Declared in	NSAttributeDescription.h
Companion guides	Core Data Programming Guide Core Data Utility Tutorial
Related sample code	Core Data HTML Store CoreRecipes CustomAtomicStoreSubclass

Overview

The `NSAttributeDescription` class is used to describe attributes of an entity described by an instance of `NSEntityDescription`.

`NSAttributeDescription` inherits from `NSPropertyDescription`, which provides most of the basic behavior. Instances of `NSAttributeDescription` are used to describe attributes, as distinct from relationships. The class adds the ability to specify the attribute type, and to specify a default value. In a managed object model, you must specify the type of all attributes—you can only use the undefined attribute type (`NSUndefinedAttributeType`) for transient attributes.

Editing Attribute Descriptions

Attribute descriptions are editable until they are used by an object graph manager. This allows you to create or modify them dynamically. However, once a description is used (when the managed object model to which it belongs is associated with a persistent store coordinator), it *must not* (indeed cannot) be changed. This is enforced at runtime: any attempt to mutate a model or any of its sub-objects after the model is associated with a persistent store coordinator causes an exception to be thrown. If you need to modify a model that is in use, create a copy, modify the copy, and then discard the objects with the old model.

Note: Default values set for attributes are retained by a managed object model, not copied. This means that attribute values do not have to implement the `NSCopying` protocol, however it also means that you should not modify any objects after they have been set as default values.

Tasks

Getting and Setting Type Information

- `attributeType` (page 7)
  Returns the type of the receiver.
- `setAttributeType:` (page 8)
  Sets the type of the receiver.
- `attributeValueClassName` (page 7)
  Returns the name of the class used to represent the receiver.
- `setAttributeValueClassName:` (page 8)
  Sets the name of the class used to represent the receiver.

Getting and Setting the Default Value

- `defaultValue` (page 7)
  Returns the default value of the receiver.
- `setDefaultValue:` (page 9)
  Sets the default value of the receiver.

Versioning Support

- `versionHash` (page 10)
  Returns the version hash for the receiver.

Value Transformers

- `valueTransformerName` (page 10)
  Returns the name of the transformer used to transform the attribute value.
- `setValueTransformerName:` (page 9)
  Sets the name of the transformer to use to transform the attribute value.

Instance Methods

attributeType

Returns the type of the receiver.

```
- (NSAttributeType)attributeType
```

Return Value
The type of the receiver.

Availability
Available in Mac OS X v10.4 and later.

See Also
- attributeValueClassName (page 7)
- setAttributeType: (page 8)

Related Sample Code
Core Data HTML Store
CoreRecipes

Declared In
NSAttributeDescription.h

attributeValueClassName

Returns the name of the class used to represent the receiver.

```
- (NSString *)attributeValueClassName
```

Return Value
The name of the class used to represent the receiver, as a string.

Availability
Available in Mac OS X v10.4 and later.

See Also
- attributeType (page 7)
- setAttributeType: (page 8)

Declared In
NSAttributeDescription.h

defaultValue

Returns the default value of the receiver.

```
- (id)defaultValue
```

Return Value
The default value of the receiver.

Availability
Available in Mac OS X v10.4 and later.

See Also
- `setDefaultValue:` (page 9)

Declared In
`NSAttributeDescription.h`

setAttributeType:

Sets the type of the receiver.

```
- (void)setAttributeType:(NSAttributeType)type
```

Parameters

type

An `NSAttributeType` constant that specifies the type for the receiver.

Special Considerations

This method raises an exception if the receiver's model has been used by an object graph manager.

Availability
Available in Mac OS X v10.4 and later.

See Also
- `attributeType` (page 7)
- `attributeValueClassName` (page 7)

Declared In
`NSAttributeDescription.h`

setAttributeValueClassName:

Sets the name of the class used to represent the receiver.

```
- (void)setAttributeValueClassName:(NSString *)className
```

Parameters

className

The name of the class used to represent the receiver.

Discussion
If you set the value class name, Core Data can check the class of any instance set as the value of an attribute.

Availability
Available in Mac OS X v10.5 and later.

See Also
- `attributeValueClassName` (page 7)

Declared In
NSAttributeDescription.h

setDefaultValue:

Sets the default value of the receiver.

```
- (void)setDefaultValue:(id)value
```

Parameters

value

The default value for the receiver.

Discussion
Default values are retained by a managed object model, not copied. This means that attribute values do not have to implement the NSCopying protocol, however it also means that you should not modify any objects after they have been set as default values.

Special Considerations

This method raises an exception if the receiver's model has been used by an object graph manager.

Availability
Available in Mac OS X v10.4 and later.

See Also
- defaultValue (page 7)

Declared In
NSAttributeDescription.h

setValueTransformerName:

Sets the name of the transformer to use to transform the attribute value.

```
- (void)setValueTransformerName:(NSString *)string
```

Parameters

string

The name of the transformer to use to transform the attribute value. The transformer must output an NSData object from transformedValue: and must allow reverse transformations.

Discussion
The receiver must be an attribute of type NSTransformedAttributeType.

If this value is not set, or is set to nil, Core Data will default to using a transformer which uses NSCoding to archive and unarchive the attribute value.

Availability
Available in Mac OS X v10.5 and later.

See Also
- valueTransformerName (page 10)

Declared In
NSAttributeDescription.h

valueTransformerName

Returns the name of the transformer used to transform the attribute value.

```
- (NSString *)valueTransformerName
```

Return Value
The name of the transformer used to transform the attribute value.

Discussion
The receiver must be an attribute of type `NSTransformedAttributeType`.

Availability
Available in Mac OS X v10.5 and later.

See Also
`- setValueTransformerName:` (page 9)

Related Sample Code
CustomAtomicStoreSubclass

Declared In
NSAttributeDescription.h

versionHash

Returns the version hash for the receiver.

```
- (NSData *)versionHash
```

Return Value
The version hash for the receiver.

Discussion
The version hash is used to uniquely identify an attribute based on its configuration. This value includes the `versionHash` information from `NSPropertyDescription` and the attribute type.

Availability
Available in Mac OS X v10.5 and later.

See Also
`- versionHash` (`NSPropertyDescription`)

Declared In
NSAttributeDescription.h

Constants

NSAttributeType

Defines the possible types of `NSAttributeType` properties. These explicitly distinguish between bit sizes to ensure data store independence.

```
typedef enum {
NSUndefinedAttributeType = 0,
NSInteger16AttributeType = 100,
NSInteger32AttributeType = 200,
NSInteger64AttributeType = 300,
NSDecimalAttributeType = 400,
NSDoubleAttributeType = 500,
NSFloatAttributeType = 600,
NSStringAttributeType = 700,
NSBooleanAttributeType = 800,
NSDateAttributeType = 900,
NSBinaryDataAttributeType = 1000,
NSTransformableAttributeType = 1800
} NSAttributeType;
```

Constants

`NSUndefinedAttributeType`

Specifies an undefined attribute type.

`NSUndefinedAttributeType` is valid for *transient* properties—Core Data will still track the property as an `id` value and register undo/redo actions, and so on. `NSUndefinedAttributeType` is illegal for non-transient properties.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSInteger16AttributeType`

Specifies a 16-bit signed integer attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSInteger32AttributeType`

Specifies a 32-bit signed integer attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSInteger64AttributeType`

Specifies a 64-bit signed integer attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSDecimalAttributeType`

Specifies an `NSDecimalNumber` attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSDoubleAttributeType`
Specifies a double attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSFloatAttributeType`
Specifies a float attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSStringAttributeType`
Specifies an `NSString` attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSBooleanAttributeType`
Specifies a Boolean attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSDateAttributeType`
Specifies an `NSDate` attribute.

Times are specified in GMT.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSBinaryDataAttributeType`
Specifies an `NSData` attribute.

Available in Mac OS X v10.4 and later.

Declared in `NSAttributeDescription.h`.

`NSTransformableAttributeType`
Specifies an attribute that uses a value transformer.

Available in Mac OS X v10.5 and later.

Declared in `NSAttributeDescription.h`.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`NSAttributeDescription.h`

Document Revision History

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

Date	Notes
2007-10-31	Updated for Mac OS X v10.5. Corrected definition of NSTransformableAttributeType.
2006-10-26	Updated for Mac OS X v10.5.
2006-05-23	First publication of this content as a separate document.

Document Revision History

Index