NXProtocolChecker

Inherits From:	Object
Declared In:	machkit/NXProtocolChecker.h

Class Description

The NXProtocolChecker class defines an object that restricts the messages that can be sent to another object (referred to as the checker's delegate). This can be particularly useful when an object with many methods, only a few of which ought to be remotely accessible, is made available using the Distributed Objects system.

A protocol checker acts as a kind of proxy; when it receives a message that is in its designated protocol, it forwards the message to its delegate, and consequently appears to be the delegate itself. However, when it receives a message not in its protocol, it raises an NX_restrictionEnforcedException exception to indicate that the message isn't allowed, whether or not the delegate implements the method.

Typically, an object that is to be distributed (yet must restrict messages) creates an NXProtocolChecker for itself and returns the checker rather than returning itself in response to any messages. The object might also register the checker as the root object of an NXConnection.

The object should be careful about vending references to **self**; the protocol checker will convert a return value of **self** to indicate the checker rather than the object for any messages that were forwarded by the checker, but direct references to the object (bypassing the checker) could be passed around by other objects.

Instance Variables

id **target**;
Protocol ***protocol**;

target	The checker's delegate
protocol	Indicates the messages the checker will forward

Method Types

Initializing a checker	– initWithObject:forProtocol:
Reimplemented Object methods	– forward:: – descriptionForMethod: – free

Instance Methods

descriptionForMethod:

– (struct objc_method_description *)**descriptionForMethod:**(SEL)*aSelector*

Returns an Objective C description for a method in the checker's protocol, or NULL if *aSelector* isn't declared as an instance method in the protocol.

See also: – **descriptionForInstanceMethod:**(Protocol class)

forward::

– **forward:**(SEL)*aSelector* **:**(void *)*args*

Forwards any message to the delegate if the method is declared in the checker's protocol; otherwise raises an NX_restrictionEnforcedException exception. If a delegate method returns **self**, the checker substitutes its own **id** for the return value so that the sender doesn't gain direct access (bypassing the checker) to the delegate.

free

– **free**

If the **free** method is not part of the protocol that the delegate responds to, this message simply frees the checker. If the **free** method is part of the protocol that the delegate responds to, the **free** message is forwarded to the delegate, and the checker is freed if the delegate returns **nil**. This ensures that that the checker is closely coupled to the delegate whether the delegate implements reference counting or not; see the NXReference protocol for more information.

initWithObject:forProtocol:

– initWithObject:*anObject* **forProtocol:**(Protocol *)*aProtocol*

Initializes a newly allocated NXProtocolChecker instance that will forward any messages in the *aProtocol* protocol to *anObject*, its delegate. Thus, the checker can be vended in lieu of *anObject* to restrict the messages that can be sent to *anObject*. If *anObject* is allowed to be freed or dereferenced by clients, the **free** method should be included in *aProtocol*. Returns the new instance.