

A. Abbreviations

AAL	ATM Adaption Layer
ACF	Active Collections Framework
API	Application Programming Interface
CORBA	Common Object Request Brokerage Architecture
DCE	Distributed Computing Environment
DHT	Distributed Hash Table
DKS	Distributed K-ary Search
DOLR	Decentralized Object Location and Routing
DOM	Distributed Object Memory
DSM	Distributed Shared Memory
J2SE	Java 2 Standard Edition
JCF	Java Collections Framework
JDBC	Java DataBase Connectivity
JDO	Java Data Objects
JVM	Java Virtual Machine
RMI	Java Remote Method Invocation
RPC	Remote Procedure Call
SQL	Structured Query Language
STL	Standard Template Library
SVM	Shared Virtual Memory
UDP	User Datagram Protocol
XML	eXtensible Markup Language

B. Class Diagrams on other Collection Frameworks

Note that classes and method has been excluded from these images when deemed irrelevant for our presentation to save some space. Some .NET constructs does not map well to UML constructs, e.g. properties has with a few exceptions been modelled as operations. Types, especially in C++, have been somewhat simplified and adapted to more Java like conventions for greater consistency.

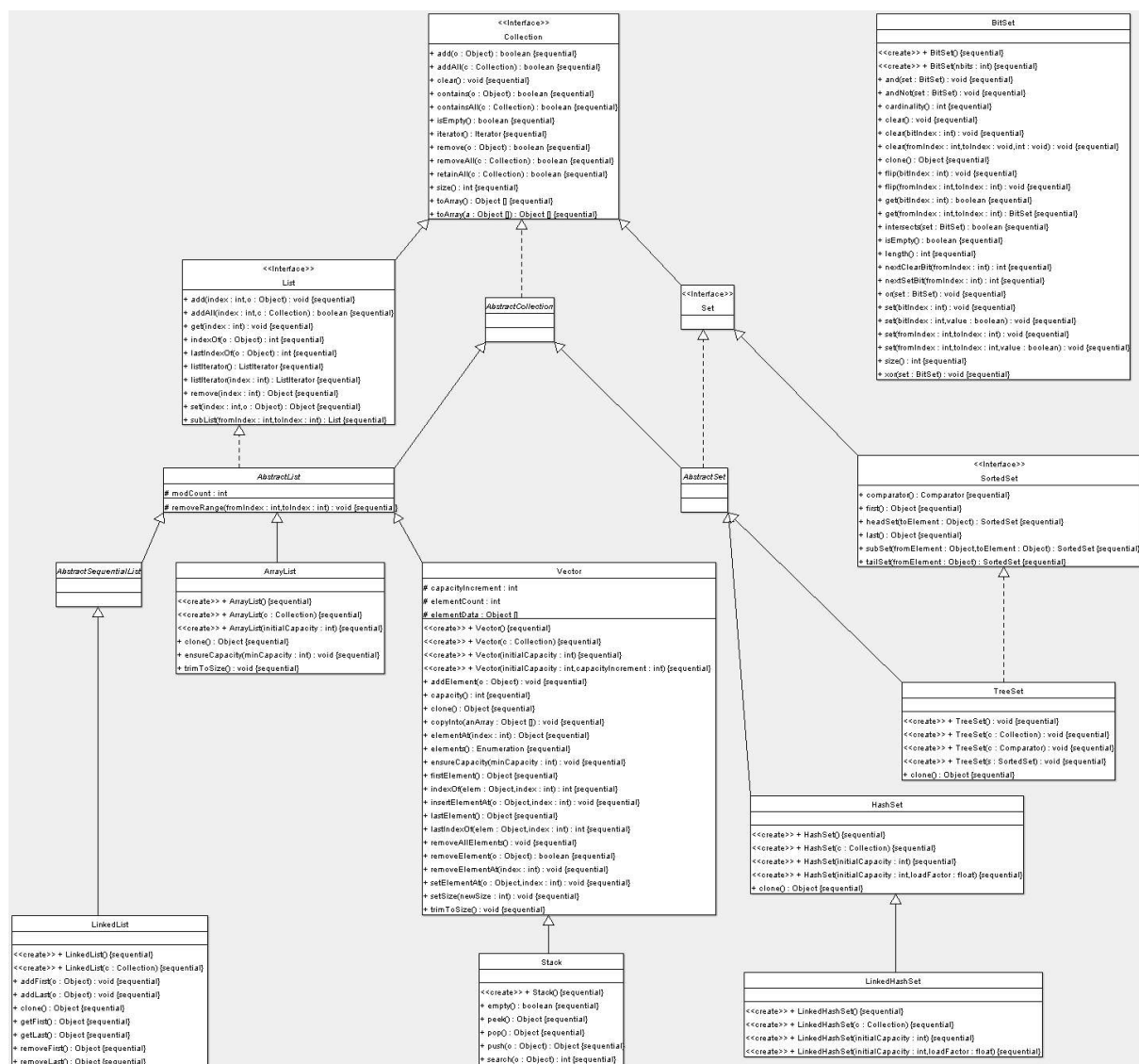


Figure 33 Sun's Java 1.4 collection classes descending from Collection, and BitSet

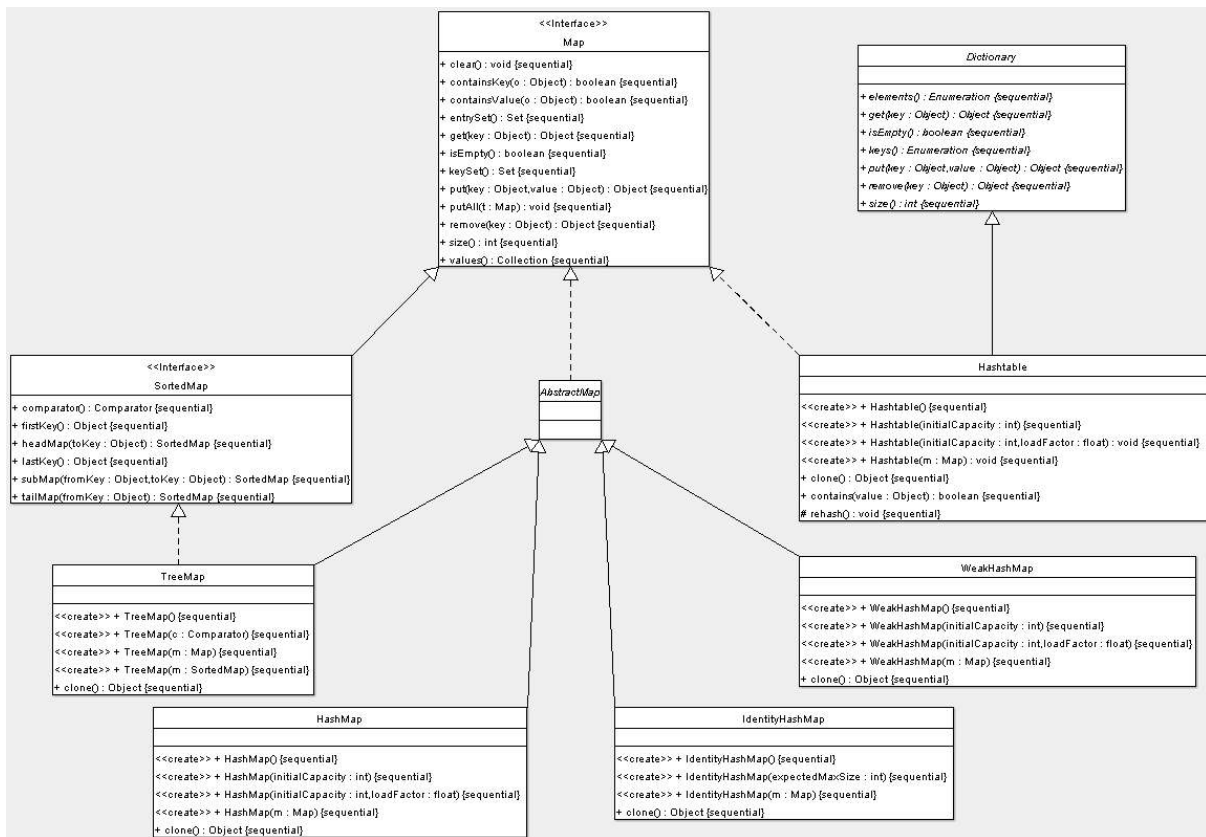


Figure 34 Sun's Java 1.4 collection classes descending from Map

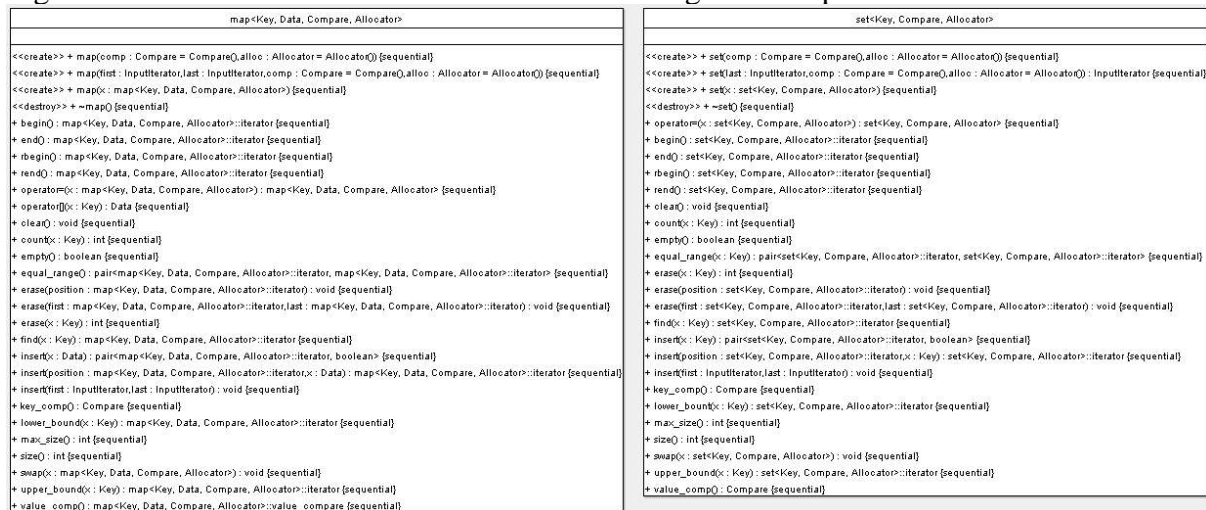


Figure 35 STL relation classes

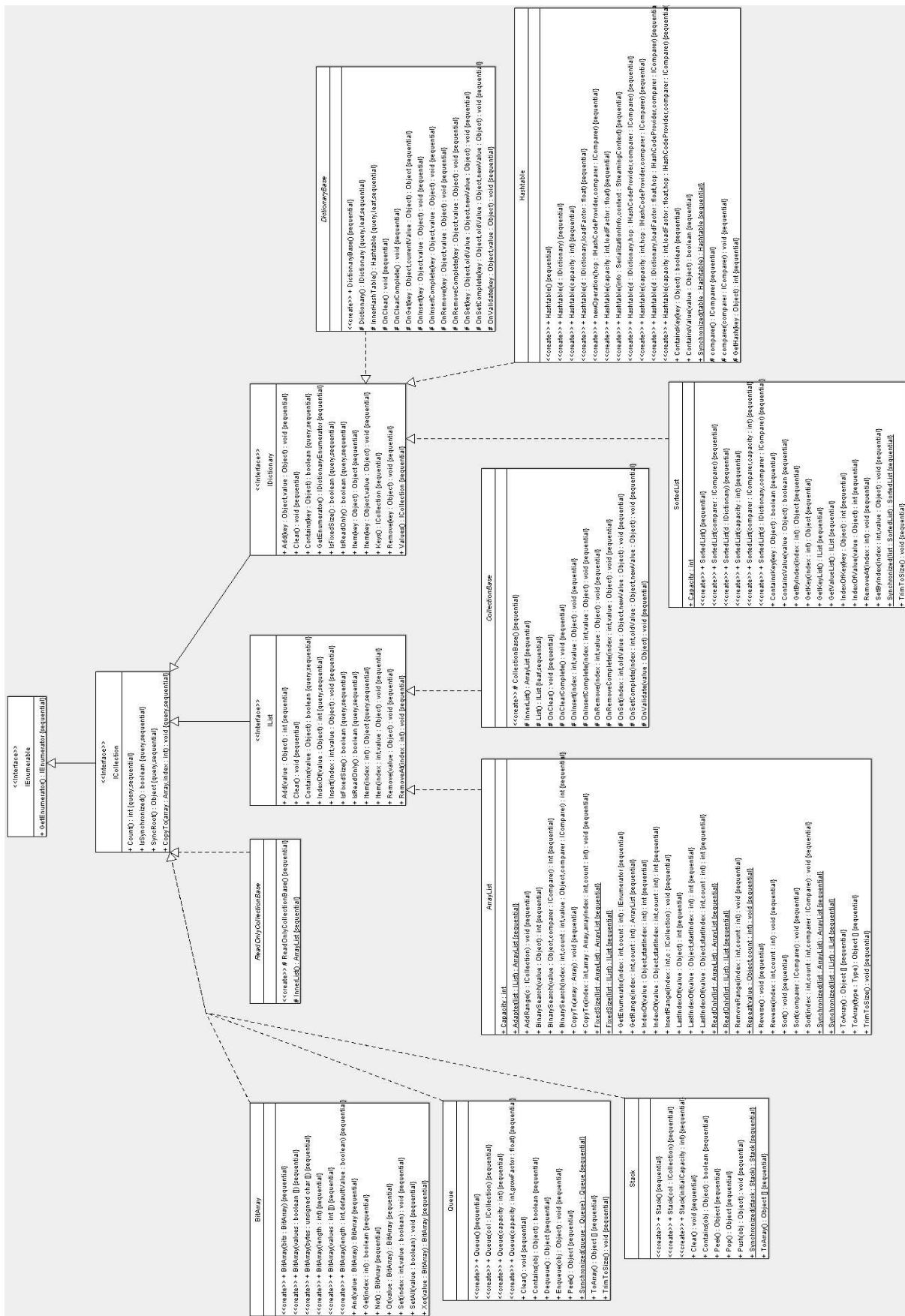


Figure 39 Microsoft's regular .Net collection classes

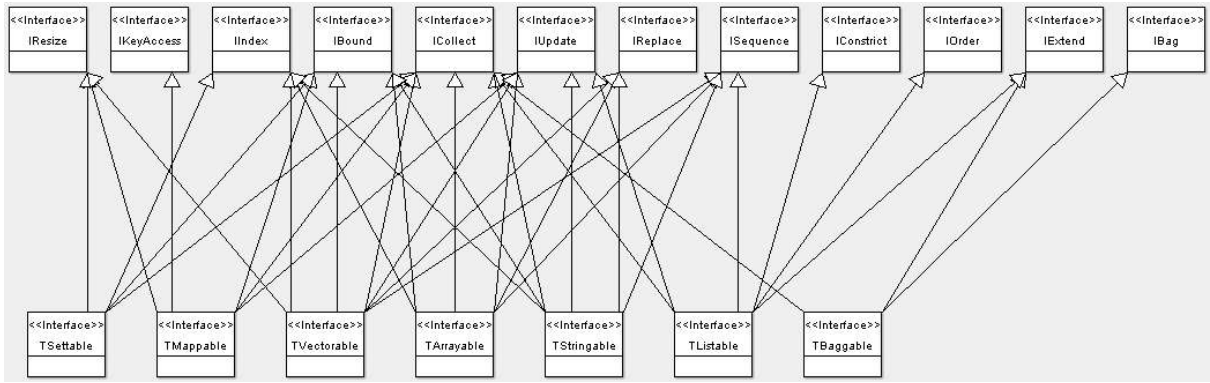


Figure 40 Mirza's interface hierarchy

C. Object Framework API

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED | FIELD | CONSTR | METHOD](#) [DETAIL: FIELD | CONSTR | METHOD](#)

[se.kth.p2p.pjo](#)

Interface GlobalObject

All Superinterfaces:

Serializable

All Known Subinterfaces:

GlobalCollection , GlobalIterator , GlobalList , GlobalListIterator , GlobalMap , GlobalSet

All Known Implementing Classes:

GlobalAggregateList , GlobalArrayList , GlobalDecoupledListIterator , GlobalFakeCollection , GlobalFakeIterator , GlobalFakeList , GlobalFakeListIterator , GlobalFakeSet , GlobalHashMap

public interface GlobalObject

extends Serializable

The interface all distributed control objects will implement. It provides access to control functions common amongst all distributed objects.

All control objects must be serializable, but this does not mean that the semantic object must or even should move with it, since any other class must be able to hold a reference to a control object and they might still want to be serializable.

Method Summary

boolean	<code>equals(Object obj)</code>
boolean	<code>getDoLocal()</code> Queries this control object to see if it is allowed to call the semantic object without interacting with the replication object.
NetworkingObject	<code>getNetworkingObject()</code> Returns the NetworkingObject this object uses to communicate with its environment.
ReplicationObject	<code>getReplicationObject()</code> Returns the ReplicationObject governing the distribution of this global object.
Object	<code>getSemanticObject()</code> Gets the semantic object pointer from the control object.

int	<code>hashCode()</code>
InvocationResult	<code>invoke(String name, Object [] args)</code> Up-call method for replication objects to invoke methods on the semantic object.
void	<code>setDoLocal(boolean value)</code> Informs this control object of whether if it can invoke methods on the local semantic object without consulting with the replication object.
void	<code>setSemanticObject(Object value)</code> Up-call from the replication object to set the semantic object pointer in the control object.

[Overview](#)
[Package](#)
[Class](#)
[Tree](#)
[Deprecated](#)
[Index](#)
[Help](#)

[PREV CLASS](#)
[NEXT CLASS](#)
[FRAMES](#)
[NO FRAMES](#)
[All Classes](#)
[SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

[se.kth.p2p.pjo](#)

Interface NetworkingObject

All Superinterfaces:

Serializable

All Known Implementing Classes:

P2PNetworkingObject

public interface **NetworkingObject**

extends Serializable

This is the basic interface that NetworkingObjects must implement.

NetworkingObjects must be serializable, since these are integral parts of GlobalObjects and they must be serializable.

Method Summary

NetworkAddress	<code>getNetworkAddress()</code> Returns the <code>NetworkAddress</code> of this global object, ie its globally unique name.
NetworkAddress	<code>getNode()</code> Returns the id of the node we're currently running on.
ReplicationObject	<code>getReplicationObject()</code> Returns the <code>ReplicationObject</code> this <code>NetworkingObject</code> is connected to.
void	<code>messageReceived(NetworkAddress address, ReplicationMessage message)</code>

	Call-back for the object manager to notify the network object that a message has been received directed to it.
<code>NetworkAddress</code>	<code>registerObject()</code> Obtains a unique id and names this object.
<code>void</code>	<code>registerObject(NetworkAddress id)</code> Names this object the given identifier.
<code>void</code>	<code>sendMessage(NetworkAddress addr, ReplicationMessage message)</code> Sends an invoke message
<code>void</code>	<code>setNetworkAddress(NetworkAddress address)</code> This is a call-back method for an object manager.
<code>void</code>	<code>setReplicationObject(ReplicationObject obj)</code> Sets the ReplicationObject of this NetworkingObject.

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

[se.kth.p2p.pjo](#)

Interface ReplicationMessage

All Superinterfaces:

Message , Serializable

All Known Implementing Classes:

RemoteInvoke , RemoteInvokeResponse

`public interface ReplicationMessage`

extends Message

An interface for Messages sent between ReplicationObjects. Such Messages will be used to create different replication protocols (with a much wider meaning than just that semantic objects are copied back and forth across the network).

Method Summary

<code>NetworkAddress</code>	<code>getObjectId()</code> Returns the identifier of the object this message concerns.
-----------------------------	---

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

Interface ReplicationObject

All Superinterfaces:

Serializable

All Known Implementing Classes:

RPCReplicationObject

public interface **ReplicationObject**

extends Serializable


The interface all replication objects must implement. A replication object implements some replication protocol for the purpose of maintaining the appearance that the local representatives of a global object really just are views on a single logical object.

Method Summary	
NetworkAddress	created() Call this function to notify the ReplicationObject after a new GlobalObject has been created and everything else initialized.
void	created(NetworkAddress id) Call this function to notify the ReplicationObject after a new GlobalObject has been created and everything else initialized.
GlobalObject	getControlObject() XXX This is a debugging function.
InvocationResult	invoke(String name, Object [] params) An implementation of this method is responsible for deciding whether to invoke the method locally, somehow synchronize with other hosts and then invoke the method locally or to invoke the method remotely.
void	messageReceived(NetworkAddress sender, ReplicationMessage message) Up-call method intended for NetworkingObjects.
void	setNetworkingObject(NetworkingObject obj) Sets the NetworkingObject of this ReplicationObject.

[Overview](#)
[Package](#)
[Class](#)
[Tree](#)
[Deprecated](#)
[Index](#)
[Help](#)

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes SUMMARY: NESTED | FIELD | CONSTR | METHOD
 METHODDETAIL: FIELD | CONSTR | METHOD

Class InvocationResult

java.lang.Object  se.kth.p2p.pjo.InvocationResult

All Implemented Interfaces:

Serializable

public abstract class InvocationResult

extends Object

implements Serializable

A class representing the result of a method invocation. You should not create instances of this class directly but rather use a public static method, like `booleanResult`, `intResult`, etc. It is serializable if the contained return value is serializable, which is always true for primitive types, should always be true for Exceptions and generally up to the programmer if needed and appropriate for objects in general.

Results can be either of primitive type, Object type or exceptions. The receiver of the result is assumed to know which type to expect, with the exception that an exception might have occurred, and thus it is a runtime exception condition to access the result with the wrong get-method.

See Also:

Serialized Form

Method Summary	
<code>static InvocationResult</code>	<code>booleanResult(boolean result)</code> Returns an InvocationResult representing a boolean return.
<code>boolean</code>	<code>booleanValue()</code> Returns the value of this Result.
<code>static InvocationResult</code>	<code>doubleResult(double result)</code> Returns an InvocationResult representing a double return.
<code>double</code>	<code>doubleValue()</code> Returns the value of this Result.
<code>static InvocationResult</code>	<code>exceptionResult(Throwable result)</code> Returns an InvocationResult representing an exception return.
<code>static InvocationResult</code>	<code>intResult(int result)</code> Returns an InvocationResult representing an int return.
<code>int</code>	<code>intValue()</code> Returns the value of this Result.
<code>boolean</code>	<code>isException()</code> Checks if this result is an exception result.
<code>static</code>	<code>longResult(long result)</code>

InvocationResult	Returns an InvocationResult representing a long return.
long	longValue() Returns the value of this Result.
static InvocationResult	objectResult(Object result) Returns an InvocationResult representing an Object return.
Object	objectValue() Returns the value of this Result.
static InvocationResult	voidResult() Returns an InvocationResult representing a void return.
void	voidValue() Returns the value of this Result.

Methods inherited from class java.lang. Object


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

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

se.kth.p2p.pjo

Class ObjectManager

java.lang.Object  se.kth.p2p.pjo.ObjectManager

public final class **ObjectManager**

extends Object

See Also:

getManager()

Method Summary

void	deregisterObject(NetworkAddress id) Deregisters a GlobalObject registered with this ObjectManager.
Collection	getLocalObjects() Returns a Collection containing the GlobalObjects that currently exist on this node.
static ObjectManager	getManager() Returns a reference to an ObjectManager

void	registerObject (NetworkAddress id, NetworkingObject obj) Registers a GlobalObject with a specific name.
void	registerObject (NetworkingObject obj) Registers a GlobalObject globally.
GlobalObject	resolve (NetworkAddress id, NetworkingObject fallback, boolean global) Returns the GlobalObject with the given id, or null if it cannot find the object.
NetworkingObject	resolveNetworkingObject (NetworkAddress id, NetworkingObject fallback, boolean global) Returns the NetworkingObject of the GlobalObject with the given id, or null if it cannot find the object.
void	start (NetworkAddress node, Runnable runnable) Sends a Runnable to a node within the network and executes it there.
void	start (Runnable runnable) Sends a Runnable to some node within the network and executes it there.
void	startLocal (Runnable runnable) Executes a Runnable locally.

Methods inherited from class java.lang. Object


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

Overview Package Class Tree Deprecated Index Help

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

se.kth.p2p.pjo

Class P2PNetworkingObject

java.lang.Object  se.kth.p2p.pjo.P2PNetworkingObject

All Implemented Interfaces:

NetworkingObject , Serializable

public class **P2PNetworkingObject**

extends Object

implements NetworkingObject

A NetworkingObject that interfaces with ObjectManager and an OverlayNetwork to fulfill its duties.

See Also:

ObjectManager, OverlayNetwork, Serialized Form

Constructor Summary

`P2PNetworkingObject()`
A constructor.

Method Summary

<code>NetworkAddress</code>	<code>getNetworkAddress()</code> Returns the <code>NetworkAddress</code> of this global object, ie its globally unique name.
<code>NetworkAddress</code>	<code>getNode()</code> Returns the id of the node we're currently running on.
<code>ReplicationObject</code>	<code>getReplicationObject()</code> Returns the <code>ReplicationObject</code> this <code>NetworkingObject</code> is connected to.
<code>void</code>	<code>messageReceived(NetworkAddress addr, ReplicationMessage message)</code> Call-back for the object manager to notify the network object that a message has been received directed to it.
<code>NetworkAddress</code>	<code>registerObject()</code> Obtains a unique id and names this object.
<code>void</code>	<code>registerObject(NetworkAddress id)</code> Names this object the given identifier.
<code>void</code>	<code>sendMessage(NetworkAddress addr, ReplicationMessage message)</code> Sends an invoke message
<code>void</code>	<code>setNetworkAddress(NetworkAddress address)</code> This is a call-back method for an object manager.
<code>void</code>	<code>setReplicationObject(ReplicationObject obj)</code> Sets the <code>ReplicationObject</code> of this <code>NetworkingObject</code> .
<code>String</code>	<code>toString()</code> Returns a <code>String</code> representation of this object.


Methods inherited from class java.lang. Object

`clone` , `equals` , `finalize` , `getClass` , `hashCode` , `notify` , `notifyAll` , `wait` , `wait` , `wait`

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#) [SUMMARY](#): [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)

Class RPCReplicationObject

java.lang.Object  se.kth.p2p.pjo.RPCReplicationObject

All Implemented Interfaces:

ReplicationObject , Serializable

public class RPCReplicationObject

extends Object

implements ReplicationObject

A replication object that only uses remote procedure calls. The semantic object thus never leaves the node it was created on, which might be necessary for object communicating with special hardware or using other special resources or desirable for exceptionally big objects. At the same time it offers no protection if the node goes down, then the object would (at best) be inaccessible until the node comes up again or (at worst) all references to it are invalidated and the node creates a new logical object when it comes up again (or not).

See Also:

Serialized Form

Field Summary	
<small>protected</small> <small>waiterServant</small>	servant The handler of blocking threads waiting for responses from the network.

Constructor Summary	
<small>RPCReplicationObject</small> (GlobalObject obj)	A constructor.

Method Summary	
<small>NetworkAddress</small>	created() Call this function to notify the ReplicationObject after a new GlobalObject has been created and everything else initialized.
<small>void</small>	created(NetworkAddress id) Call this function to notify the ReplicationObject after a new GlobalObject has been created and everything else initialized.
<small>GlobalObject</small>	getControlObject() XXX This is a debugging function.

InvocationResult	invoke (String name, Object [] args) An implementation of this method is responsible for deciding whether to invoke the method locally, somehow synchronize with other hosts and then invoke the method locally or to invoke the method remotely.
void	messageReceived (NetworkAddress sender, ReplicationMessage message) Up-call method intended for NetworkingObjects.
protected void	result (RemoteInvokeResponse response) Handles a RemoteInvokeResponse passed to messageReceived.
void	setNetworkingObject (NetworkingObject obj) Sets the NetworkingObject of this ReplicationObject.
String	toString () Returns a String representation of this object.

Methods inherited from class java.lang. Object	
clone , equals , finalize , getClass , hashCode , notify , notifyAll , wait , wait , wait	

D. Collection Framework API

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

[se.kth.p2p.util](#)

Interface GlobalCollection

All Superinterfaces:

`GlobalObject` , `Serializable`

All Known Subinterfaces:

`GlobalList` , `GlobalSet`

All Known Implementing Classes:

`GlobalAggregateList` , `GlobalArrayList` , `GlobalFakeCollection` , `GlobalFakeList` , `GlobalFakeSet`

`public interface GlobalCollection`

extends `GlobalObject`

An interface for `GlobalObjects` containing collections of objects.

Note that it would often be a bad idea to put a global object in a collection and a better idea to put the name of the global object in the collection and use

```
ObjectManager.resolve(se.kth.p2p.NetworkAddress,  
se.kth.p2p.pjo.NetworkingObject, boolean).
```

Unless otherwise noted all methods behaves as in `Collection`. Note specifically that this includes unchecked exceptions.

See Also:

`Collection`

Method Summary

<code>boolean</code>	<code>add(Object o)</code> See <code>Collection.add(java.lang.Object)</code> .
<code>boolean</code>	<code>addAll(Collection c)</code> See <code>Collection.addAll(java.util.Collection)</code> .
<code>void</code>	<code>clear()</code> See <code>Collection.clear()</code> .
<code>boolean</code>	<code>contains(Object o)</code> See <code>Collection.contains(java.lang.Object)</code>
<code>boolean</code>	<code>containsAll(Collection c)</code>

	See <code>Collection.containsAll(java.util.Collection)</code> .
<code>boolean</code>	<code>equals(Object o)</code> Throws <code>UnsupportedOperationException</code> .
<code>int</code>	<code>hashCode()</code> Throws <code>UnsupportedOperationException</code> .
<code>boolean</code>	<code>isEmpty()</code> See <code>Collection.isEmpty()</code> .
<code>GlobalIterator</code>	<code>iterator()</code> See <code>Collection.iterator()</code> .
<code>boolean</code>	<code>remove(Object o)</code> See <code>Collection.remove(java.lang.Object)</code> .
<code>boolean</code>	<code>removeAll(Collection c)</code> See <code>Collection.removeAll(java.util.Collection)</code> .
<code>boolean</code>	<code>retainAll(Collection c)</code> See <code>Collection.retainAll(java.util.Collection)</code> .
<code>int</code>	<code>size()</code> See <code>Collection.size()</code> .
<code>Object []</code>	<code>toArray()</code> See <code>Collection.toArray()</code> .
<code>Object []</code>	<code>toArray(Object [] a)</code> Returns an array containing all of the elements in this collection; the runtime type of the returned array is that of the specified array.

Methods inherited from interface `se.kth.p2p.pjo.GlobalObject`

`getDoLocal` , `getNetworkingObject` , `getReplicationObject` , `getSemanticObject` , `invoke` , `setDoLocal` , `setSemanticObject`

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

[se.kth.p2p.util](#)

Interface `GlobalList`

All Superinterfaces:

`GlobalCollection` , `GlobalObject` , `Serializable`

All Known Implementing Classes:

`GlobalAggregateList` , `GlobalArrayList` , `GlobalFakeList`

`public interface GlobalList`

extends GlobalCollection

An interface for GlobalObjects containing an ordered sequence of Objects.

Unless otherwise noted all methods behaves as in List. Note specifically that this includes unchecked exceptions.

See Also:

List

Method Summary	
void	add (int index, Object element) See List.add(int, Object).
boolean	add (Object o) See List.add(Object).
boolean	addAll (Collection c) See List.addAll(Collection).
boolean	addAll (int index, Collection c) See List.addAll(int, Collection).
void	clear () See List.clear().
boolean	contains (Object o) See List.contains(java.lang.Object).
boolean	containsAll (Collection c) See List.containsAll(java.util.Collection).
boolean	equals (Object o) Throws UnsupportedOperationException.
Object	get (int index) See List.get(int).
int	hashCode () Throws UnsupportedOperationException.
int	indexOf (Object o) See List.indexOf(java.lang.Object).
boolean	isEmpty () See List.isEmpty().
GlobalIterator	iterator () See List.iterator().
int	lastIndexOf (Object o) See List.lastIndexOf(java.lang.Object).
GlobalListIterator	listIterator ()

	See <code>List.listIterator()</code> .
<code>GlobalListIterator</code>	<code>listIterator(int index)</code> See <code>List.listIterator(int)</code> .
<code>Object</code>	<code>remove(int index)</code> See <code>List.remove(int)</code> .
<code>boolean</code>	<code>remove(Object o)</code> See <code>List.remove(Object)</code> .
<code>boolean</code>	<code>removeAll(Collection c)</code> See <code>List.removeAll(java.util.Collection)</code> .
<code>boolean</code>	<code>retainAll(Collection c)</code> See <code>List.retainAll(java.util.Collection)</code> .
<code>Object</code>	<code>set(int index, Object element)</code> See <code>List.set(int, java.lang.Object)</code> .
<code>int</code>	<code>size()</code> See <code>List.size()</code> .
<code>GlobalList</code>	<code>subList(int fromIndex, int toIndex)</code> See <code>List.subList(int, int)</code> .
<code>Object []</code>	<code>toArray()</code> Returns an array containing all elements in this list.
<code>Object []</code>	<code>toArray(Object [] a)</code> Returns an array containing all elements in this list.

Methods inherited from interface `se.kth.p2p.pjo.GlobalObject`

`getDoLocal` , `getNetworkingObject` , `getReplicationObject` , `getSemanticObject` ,
`invoke` , `setDoLocal` , `setSemanticObject`

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED | FIELD | CONSTR | METHOD](#) [DETAIL: FIELD | CONSTR | METHOD](#)

[se.kth.p2p.util](#)

Interface `GlobalMap`

All Superinterfaces:

`GlobalObject` , `Serializable`

All Known Implementing Classes:

`GlobalHashMap`

`public interface GlobalMap`

extends `GlobalObject`

An interface for global objects that contains a mapping from a set of objects to other objects.

Unless otherwise noted all methods behaves as in `Map`. Note specifically that this includes unchecked exceptions. Note also that the `entrySet()` call is missing since it is currently believed that implementing it will always be less efficient than using the `keySet()`, `get()` calls to iterate the mapping.

See Also:

`Map`

Method Summary	
<code>void</code>	clear() See <code>Map.clear()</code> .
<code>boolean</code>	containsKey (Object key) See <code>Map.containsKey(java.lang.Object)</code> .
<code>boolean</code>	containsValue (Object value) See <code>Map.containsValue(java.lang.Object)</code> .
<code>boolean</code>	equals (Object o) Throws <code>UnsupportedOperationException</code> .
<code>Object</code>	get (Object key) See <code>Map.get(java.lang.Object)</code> .
<code>int</code>	hashCode () Throws <code>UnsupportedOperationException</code> .
<code>boolean</code>	isEmpty () See <code>Map.isEmpty()</code> .
<code>GlobalSet</code>	keySet () See <code>Map.keySet()</code> .
<code>Object</code>	put (Object key, Object value) See <code>Map.put(java.lang.Object, java.lang.Object)</code> .
<code>void</code>	putAll (Map t) See <code>Map.putAll(java.util.Map)</code> .
<code>Object</code>	remove (Object key) See <code>Map.remove(java.lang.Object)</code> .
<code>int</code>	size () See <code>Map.size()</code> .
<code>GlobalCollection</code>	values () See <code>Map.values()</code> .

se.kth.p2p.util

Interface GlobalSet

All Superinterfaces:

GlobalCollection , GlobalObject , Serializable

All Known Implementing Classes:

GlobalFakeSet

public interface **GlobalSet**

extends GlobalCollection

An interface for GlobalObjects containing sets of objects. A set is a collection without duplicates.

Note: Great care must be exercised if mutable objects are used as set elements. The behavior of a set is not specified if the value of an object is changed in a manner that affects equals comparisons while the object is an element in the set. A special case of this prohibition is that it is not permissible for a set to contain itself as an element.

Unless otherwise noted all methods behaves as in `set`. Note specifically that this includes unchecked exceptions.

See Also:

Set

Method Summary	
boolean	add (Object o) See Set.add(java.lang.Object).
boolean	addAll (Collection c) See Set.addAll(java.util.Collection).
void	clear () See Set.clear().
boolean	contains (Object o) See Set.contains(java.lang.Object).
boolean	containsAll (Collection c) See Set.containsAll(java.util.Collection).
boolean	equals (Object o) Throws UnsupportedOperationException.
int	hashCode () Throws UnsupportedOperationException.
boolean	isEmpty ()

	See <code>Set.isEmpty()</code> .
<code>GlobalIterator</code>	<code>iterator()</code> See <code>Set.iterator()</code> .
<code>boolean</code>	<code>remove(Object o)</code> See <code>Set.remove(java.lang.Object)</code> .
<code>boolean</code>	<code>removeAll(Collection c)</code> See <code>Set.removeAll(java.util.Collection)</code> .
<code>boolean</code>	<code>retainAll(Collection c)</code> See <code>Set.retainAll(java.util.Collection)</code> .
<code>int</code>	<code>size()</code> See <code>Set.size()</code> .
<code>Object []</code>	<code>toArray()</code> Returns an array containing all of the elements in this set.
<code>Object []</code>	<code>toArray(Object [] a)</code> Returns an array containing all of the elements in this set; the runtime type of the returned array is that of the specified array.

Methods inherited from interface `se.kth.p2p.pjo.GlobalObject`

`getDoLocal` , `getNetworkingObject` , `getReplicationObject` , `getSemanticObject` ,
`invoke` , `setDoLocal` , `setSemanticObject`

[Overview](#) [Package](#) [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

`se.kth.p2p.util`

Interface `GlobalIterator`

All Superinterfaces:

`GlobalObject` , `Serializable`

All Known Subinterfaces:

`GlobalListIterator`

All Known Implementing Classes:

`GlobalDecoupledListIterator` , `GlobalFakeIterator` , `GlobalFakeListIterator`

public interface `GlobalIterator`

extends `GlobalObject`

An iterator over a collection. This is a direct parallel to `Iterator` although adapted for `GlobalObjects` by adding the possible `RemoteException`.

Unless otherwise noted all methods behaves as in `Iterator`. Note specifically that this includes unchecked exceptions.

See Also:

`Iterator`

Method Summary	
<code>boolean</code>	<code>hasNext()</code> See <code>Iterator.hasNext()</code> .
<code>Object</code>	<code>next()</code> See <code>Iterator.next()</code> .
<code>void</code>	<code>remove()</code> See <code>Iterator.remove()</code> .

Methods inherited from interface <code>se.kth.p2p.pjo.GlobalObject</code>
<code>equals</code> , <code>getDoLocal</code> , <code>getNetworkingObject</code> , <code>getReplicationObject</code> , <code>getSemanticObject</code> , <code>hashCode</code> , <code>invoke</code> , <code>setDoLocal</code> , <code>setSemanticObject</code>

Overview	Package	Class	Tree	Deprecated	Index	Help
----------	---------	-------	------	------------	-------	------

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#) [SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHODDETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

`se.kth.p2p.util`

Interface `GlobalListIterator`

All Superinterfaces:

`GlobalIterator` , `GlobalObject` , `Serializable`

All Known Implementing Classes:

`GlobalDecoupledListIterator` , `GlobalFakeListIterator`

`public interface GlobalListIterator`

extends `GlobalIterator`

An interface for global list iterators. Global list iterators are global iterators that iterate a list and can iterate it in both directions and perhaps also invoke some more methods on it.

Unless otherwise noted all methods behaves as in `ListIterator`. Note specifically that this includes unchecked exceptions.

See Also:

`ListIterator`

Method Summary	
void	add (Object o) See <code>ListIterator.add(java.lang.Object)</code> .
boolean	hasNext () See <code>ListIterator.hasNext()</code> .
boolean	hasPrevious () See <code>ListIterator.hasPrevious()</code> .
Object	next () See <code>ListIterator.next()</code> .
int	nextIndex () See <code>ListIterator.nextIndex()</code> .
Object	previous () See <code>ListIterator.previous()</code> .
int	previousIndex () See <code>ListIterator.previousIndex()</code> .
void	remove () See <code>ListIterator.remove()</code> .
void	set (Object o) See <code>ListIterator.set(java.lang.Object)</code> .


Methods inherited from interface <code>se.kth.p2p.pjo.GlobalObject</code>
<code>equals</code> , <code>getDoLocal</code> , <code>getNetworkingObject</code> , <code>getReplicationObject</code> , <code>getSemanticObject</code> , <code>hashCode</code> , <code>invoke</code> , <code>setDoLocal</code> , <code>setSemanticObject</code>

Overview	Package	Class	Tree	Deprecated	Index	Help
----------	---------	-------	------	------------	-------	------

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes SUMMARY: NESTED | FIELD | CONSTR | METHOD
 METHOD

se.kth.p2p.util

Class `GlobalAggregateList`

`java.lang.Object`  `se.kth.p2p.util.GlobalAggregateList`

All Implemented Interfaces:

`GlobalCollection` , `GlobalList` , `GlobalObject` , `Serializable`

`public class GlobalAggregateList`

extends `Object`

implements `GlobalList`

A `GlobalList` that is made up of the concatenation of other `GlobalList`s. It thus doesn't contain any objects directly, but references to other lists that contain the objects.

This class defines the special operation `append(se.kth.p2p.util.GlobalList)` that lets you append a copy of another list to the end of this list. This lets you build lists that are distributed over multiple nodes, while maintaining clear ownership of the different objects and providing intuitive operational semantics for those used to the Java Collections Framework. In this implementation only `GlobalArrayList` lists can be appended.

See Also:

Serialized Form

Constructor Summary	
<code>GlobalAggregateList()</code> A constructor.	
<code>GlobalAggregateList(NetworkAddress addr)</code> A constructor that allows explicitly naming the object.	

Method Summary	
<code>void</code>	<code>add(int index, Object element)</code> See <code>List.add(int, Object)</code> .
<code>boolean</code>	<code>add(Object o)</code> See <code>List.add(Object)</code> .
<code>boolean</code>	<code>addAll(Collection c)</code> See <code>List.addAll(Collection)</code> .
<code>boolean</code>	<code>addAll(int index, Collection c)</code> See <code>List.addAll(int, Collection)</code> .
<code>void</code>	<code>append(GlobalList list)</code> Appends a clone of <code>list</code> to the end of this list.
<code>void</code>	<code>clear()</code> See <code>List.clear()</code> .
<code>boolean</code>	<code>contains(Object elem)</code> See <code>List.contains(java.lang.Object)</code> .
<code>boolean</code>	<code>containsAll(Collection c)</code> See <code>List.containsAll(java.util.Collection)</code> .
<code>boolean</code>	<code>equals(Object obj)</code> Throws <code>UnsupportedOperationException</code> .

Object	get (int index) See <code>List.get(int)</code> .
boolean	getDoLocal () Queries this control object to see if it is allowed to call the semantic object without interacting with the replication object.
NetworkingObject	getNetworkingObject () Returns the NetworkingObject this object uses to communicate with its environment.
ReplicationObject	getReplicationObject () Returns the ReplicationObject governing the distribution of this global object.
Object	getSemanticObject () Gets the semantic object pointer from the control object.
int	hashCode () Throws <code>UnsupportedOperationException</code> .
int	indexOf (Object elem) See <code>List.indexOf(java.lang.Object)</code> .
InvocationResult	invoke (String name, Object [] args) Up-call method for replication objects to invoke methods on the semantic object.
boolean	isEmpty () See <code>List.isEmpty()</code> .
GlobalIterator	iterator () See <code>List.iterator()</code> .
int	lastIndexOf (Object elem) See <code>List.lastIndexOf(java.lang.Object)</code> .
GlobalListIterator	listIterator () See <code>List.listIterator()</code> .
GlobalListIterator	listIterator (int index) See <code>List.listIterator(int)</code> .
Object	remove (int index) See <code>List.remove(int)</code> .
boolean	remove (Object o) See <code>List.remove(Object)</code> .
boolean	removeAll (Collection c) See <code>List.removeAll(java.util.Collection)</code> .
boolean	retainAll (Collection c) See <code>List.retainAll(java.util.Collection)</code> .
Object	set (int index, Object element) See <code>List.set(int, java.lang.Object)</code> .

<code>void</code>	<code>setDoLocal(boolean val)</code> Informs this control object of whether if it can invoke methods on the local semantic object without consulting with the replication object.
<code>void</code>	<code>setSemanticObject(Object obj)</code> Up-call from the replication object to set the semantic object pointer in the control object.
<code>int</code>	<code>size()</code> See <code>List.size()</code> .
<code>GlobalList</code>	<code>subList(int fromIndex, int toIndex)</code> See <code>List.subList(int, int)</code> .
<code>Object []</code>	<code>toArray()</code> Returns an array containing all elements in this list.
<code>Object []</code>	<code>toArray(Object [] a)</code> Returns an array containing all elements in this list.
<code>String</code>	<code>toString()</code> Returns a String description of this object.


Methods inherited from class java.lang. Object
<code>clone</code> , <code>finalize</code> , <code>getClass</code> , <code>notify</code> , <code>notifyAll</code> , <code>wait</code> , <code>wait</code> , <code>wait</code>

Overview	Package	Class	Tree	Deprecated	Index	Help
-----------------	----------------	--------------	-------------	-------------------	--------------	-------------

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes SUMMARY: NESTED | FIELD | CONSTR | METHOD
 METHOD

[se.kth.p2p.util](#)

Class GlobalArrayList

java.lang.Object  se.kth.p2p.util.GlobalArrayList

All Implemented Interfaces:

Cloneable , GlobalCollection , GlobalList , GlobalObject , Serializable

public final class GlobalArrayList

extends Object

implements Cloneable , GlobalList

A GlobalObject wrapping an ArrayList.

See Also:

Serialized Form

Constructor Summary

<code>GlobalArrayList()</code> A constructor.	
<code>GlobalArrayList(NetworkAddress address)</code> A constructor.	

Method Summary

void	<code>add(int index, Object element)</code> See <code>List.add(int, Object)</code> .
boolean	<code>add(Object o)</code> See <code>List.add(Object)</code> .
boolean	<code>addAll(Collection c)</code> See <code>List.addAll(Collection)</code> .
boolean	<code>addAll(int index, Collection c)</code> See <code>List.addAll(int, Collection)</code> .
void	<code>clear()</code> See <code>List.clear()</code> .
Object	<code>clone()</code> Clones this <code>GlobalArrayList</code> , ie creates a new <code>GlobalArrayList</code> with a new unique name and a shallow copy of the backing <code>ArrayList</code> .
boolean	<code>contains(Object elem)</code> See <code>List.contains(java.lang.Object)</code> .
boolean	<code>containsAll(Collection c)</code> See <code>List.containsAll(java.util.Collection)</code> .
void	<code>ensureCapacity(int minCapacity)</code> Increases the capacity of this <code>GlobalArrayList</code> instance, if necessary, to ensure that it can hold at least the number of elements specified by the minimum capacity argument.
boolean	<code>equals(Object obj)</code> Throws <code>UnsupportedOperationException</code> .
Object	<code>get(int index)</code> See <code>List.get(int)</code> .
boolean	<code>getDoLocal()</code> Queries this control object to see if it is allowed to call the semantic object without interacting with the replication object.
NetworkingObject	<code>getNetworkingObject()</code>

	Returns the NetworkingObject this object uses to communicate with its environment.
ReplicationObject	getReplicationObject() Returns the ReplicationObject governing the distribution of this global object.
Object	getSemanticObject() Gets the semantic object pointer from the control object.
int	hashCode() Throws UnsupportedOperationException.
int	indexOf(Object elem) See List.indexOf(java.lang.Object).
InvocationResult	invoke(String name, Object [] args) Up-call method for replication objects to invoke methods on the semantic object.
boolean	isEmpty() See List.isEmpty().
GlobalIterator	iterator() See List.iterator().
int	lastIndexOf(Object elem) See List.lastIndexOf(java.lang.Object).
GlobalListIterator	listIterator() See List.listIterator().
GlobalListIterator	listIterator(int index) See List.listIterator(int).
Object	remove(int index) See List.remove(int).
boolean	remove(Object o) See List.remove(Object).
boolean	removeAll(Collection c) See List.removeAll(java.util.Collection).
boolean	retainAll(Collection c) See List.retainAll(java.util.Collection).
Object	set(int index, Object element) See List.set(int, java.lang.Object).
void	setDoLocal(boolean val) Informs this control object of whether if it can invoke methods on the local semantic object without consulting with the replication object.
void	setSemanticObject(Object obj) Up-call from the replication object to set the semantic object pointer in the control object.

int	size() See <code>List.size()</code> .
GlobalList	subList(int fromIndex, int toIndex) See <code>List.subList(int, int)</code> .
Object []	toArray() Returns an array containing all elements in this list.
Object []	toArray(Object [] a) Returns an array containing all elements in this list.
String	toString() Returns a String description of this object.
void	trimToSize() Trims the capacity of this <code>GlobalArrayList</code> instance to be the list's current size.

Methods inherited from class `java.lang.Object`


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

Overview **Package** **Class** **Tree** **Deprecated** **Index** **Help**

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

se.kth.p2p.util

Class `GlobalHashMap`

`java.lang.Object`  `se.kth.p2p.util.GlobalHashMap`

All Implemented Interfaces:

`GlobalMap` , `GlobalObject` , `Serializable`

public class `GlobalHashMap`

extends `Object`

implements `GlobalMap`

A wrapper around a `HashMap`. This wrapper publishes most of the methods exposed by `HashMap` as a global object.

See Also:

`HashMap`, `Serialized Form`

Constructor Summary

<code>GlobalHashMap()</code> A constructor.	
<code>GlobalHashMap(NetworkAddress address)</code> A constructor that explicitly names the new object.	

Method Summary	
void	<code>clear()</code> See <code>Map.clear()</code> .
boolean	<code>containsKey(Object key)</code> See <code>Map.containsKey(java.lang.Object)</code> .
boolean	<code>containsValue(Object value)</code> See <code>Map.containsValue(java.lang.Object)</code> .
boolean	<code>equals(Object obj)</code> Throws <code>UnsupportedOperationException</code> .
Object	<code>get(Object key)</code> See <code>Map.get(java.lang.Object)</code> .
boolean	<code>getDoLocal()</code> Queries this control object to see if it is allowed to call the semantic object without interacting with the replication object.
NetworkingObject	<code>getNetworkingObject()</code> Returns the <code>NetworkingObject</code> this object uses to communicate with its environment.
ReplicationObject	<code>getReplicationObject()</code> Returns the <code>ReplicationObject</code> governing the distribution of this global object.
Object	<code>getSemanticObject()</code> Gets the semantic object pointer from the control object.
int	<code>hashCode()</code> Throws <code>UnsupportedOperationException</code> .
InvocationResult	<code>invoke(String name, Object [] args)</code> Up-call method for replication objects to invoke methods on the semantic object.
boolean	<code>isEmpty()</code> See <code>Map.isEmpty()</code> .
GlobalSet	<code>keySet()</code> See <code>Map.keySet()</code> .
Object	<code>put(Object key, Object value)</code> See <code>Map.put(java.lang.Object, java.lang.Object)</code> .
void	<code>putAll(Map m)</code>

	<code>See Map.putAll(java.util.Map).</code>
<code>Object</code>	<code>remove(Object key)</code> <code>See Map.remove(java.lang.Object).</code>
<code>void</code>	<code>setDoLocal(boolean val)</code> Informs this control object of whether if it can invoke methods on the local semantic object without consulting with the replication object.
<code>void</code>	<code>setSemanticObject(Object obj)</code> Up-call from the replication object to set the semantic object pointer in the control object.
<code>int</code>	<code>size()</code> <code>See Map.size().</code>
<code>String</code>	<code>toString()</code> Returns a String description of this object.
<code>GlobalCollection</code>	<code>values()</code> <code>See Map.values().</code>

Methods inherited from class java.lang. Object

`clone , finalize , getClass , notify , notifyAll , wait , wait , wait`

E. Tables

Table 7 add-append trace data

Collection size	Nodes	Avg T_{total} (ms)	Avg T_{add} (ms)	Append time ($1-T_{add}/T_{total}$)
660000	1	2105		
740000	1	3400		
	2	4529	4016	11%
	4	3320	2098	37%
	8	4058	1076	73%
820000	1	4700		
	2	4540	4102	10%
	4	3350	2219	34%
	8	3730	1273	66%
900000	1	4662		
	2	5677	4739	17%
	4	5514	2304	58%
	8	3692	1310	65%
980000	1	9151		
	2	5977	4989	17%
	4	4358	2628	40%
	8	4018	1324	67%