| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ServerRequestInterceptorOperations.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptor.html)   [**NEXT CLASS**](http://docs.google.com/org/omg/PortableInterceptor/SUCCESSFUL.html) | [**FRAMES**](http://docs.google.com/index.html?org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html)    [**NO FRAMES**](http://docs.google.com/ServerRequestInterceptorOperations.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

## **org.omg.PortableInterceptor**

Interface ServerRequestInterceptorOperations

**All Superinterfaces:** [InterceptorOperations](http://docs.google.com/org/omg/PortableInterceptor/InterceptorOperations.html) **All Known Subinterfaces:** [ServerRequestInterceptor](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptor.html)

public interface **ServerRequestInterceptorOperations**extends [InterceptorOperations](http://docs.google.com/org/omg/PortableInterceptor/InterceptorOperations.html)

Server-side request interceptor.

A request Interceptor is designed to intercept the flow of a request/reply sequence through the ORB at specific points so that services can query the request information and manipulate the service contexts which are propagated between clients and servers. The primary use of request Interceptors is to enable ORB services to transfer context information between clients and servers. There are two types of request Interceptors: client-side and server-side.

To write a server-side Interceptor, implement the ServerRequestInterceptor interface.

**See Also:**[ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html)

| **Method Summary** | |
| --- | --- |
| void | [**receive\_request\_service\_contexts**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html#receive_request_service_contexts(org.omg.PortableInterceptor.ServerRequestInfo))([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)            Allows the interceptor to process service context information. |
| void | [**receive\_request**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html#receive_request(org.omg.PortableInterceptor.ServerRequestInfo))([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)            Allows an Interceptor to query request information after all the information, including operation parameters, are available. |
| void | [**send\_exception**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html#send_exception(org.omg.PortableInterceptor.ServerRequestInfo))([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)            Allows an Interceptor to query the exception information and modify the reply service context before the exception is thrown to the client. |
| void | [**send\_other**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html#send_other(org.omg.PortableInterceptor.ServerRequestInfo))([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)            Allows an Interceptor to query the information available when a request results in something other than a normal reply or an exception. |
| void | [**send\_reply**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html#send_reply(org.omg.PortableInterceptor.ServerRequestInfo))([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)            Allows an Interceptor to query reply information and modify the reply service context after the target operation has been invoked and before the reply is returned to the client. |

| **Methods inherited from interface org.omg.PortableInterceptor.**[**InterceptorOperations**](http://docs.google.com/org/omg/PortableInterceptor/InterceptorOperations.html) |
| --- |
| [destroy](http://docs.google.com/org/omg/PortableInterceptor/InterceptorOperations.html#destroy()), [name](http://docs.google.com/org/omg/PortableInterceptor/InterceptorOperations.html#name()) |

| **Method Detail** |
| --- |

### receive\_request\_service\_contexts

void **receive\_request\_service\_contexts**([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)  
 throws [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html)

Allows the interceptor to process service context information.

At this interception point, Interceptors must get their service context information from the incoming request transfer it to PortableInterceptor.Current's slots.

This interception point is called before the servant manager is called. Operation parameters are not yet available at this point. This interception point may or may not execute in the same thread as the target invocation.

This interception point may throw a system exception. If it does, no other Interceptors' receive\_request\_service\_contexts operations are called. Those Interceptors on the Flow Stack are popped and their send\_exception interception points are called.

This interception point may also throw a ForwardRequest exception. If an Interceptor throws this exception, no other Interceptors' receive\_request\_service\_contexts operations are called. Those Interceptors on the Flow Stack are popped and their send\_other interception points are called.

Compliant Interceptors shall properly follow completion\_status semantics if they throw a system exception from this interception point. The completion\_status shall be COMPLETED\_NO.

**Parameters:**ri - Information about the current request being intercepted. **Throws:** [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html) - If thrown, indicates to the ORB that a retry of the request should occur with the new object given in the exception.

### receive\_request

void **receive\_request**([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)  
 throws [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html)

Allows an Interceptor to query request information after all the information, including operation parameters, are available. This interception point shall execute in the same thread as the target invocation.

In the DSI model, since the parameters are first available when the user code calls arguments, receive\_request is called from within arguments. It is possible that arguments is not called in the DSI model. The target may call set\_exception before calling arguments. The ORB shall guarantee that receive\_request is called once, either through arguments or through set\_exception. If it is called through set\_exception, requesting the arguments will result in NO\_RESOURCES being thrown with a standard minor code of 1.

This interception point may throw a system exception. If it does, no other Interceptors' receive\_request operations are called. Those Interceptors on the Flow Stack are popped and their send\_exception interception points are called.

This interception point may also throw a ForwardRequest exception. If an Interceptor throws this exception, no other Interceptors' receive\_request operations are called. Those Interceptors on the Flow Stack are popped and their send\_other interception points are called.

Compliant Interceptors shall properly follow completion\_status semantics if they throw a system exception from this interception point. The completion\_status shall be COMPLETED\_NO.

**Parameters:**ri - Information about the current request being intercepted. **Throws:** [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html) - If thrown, indicates to the ORB that a retry of the request should occur with the new object given in the exception.

### send\_reply

void **send\_reply**([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)

Allows an Interceptor to query reply information and modify the reply service context after the target operation has been invoked and before the reply is returned to the client. This interception point shall execute in the same thread as the target invocation.

This interception point may throw a system exception. If it does, no other Interceptors' send\_reply operations are called. The remaining Interceptors in the Flow Stack shall have their send\_exception interception point called.

Compliant Interceptors shall properly follow completion\_status semantics if they throw a system exception from this interception point. The completion\_status shall be COMPLETED\_YES.

**Parameters:**ri - Information about the current request being intercepted.

### send\_exception

void **send\_exception**([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)  
 throws [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html)

Allows an Interceptor to query the exception information and modify the reply service context before the exception is thrown to the client. When an exception occurs, this interception point is called. This interception point shall execute in the same thread as the target invocation.

This interception point may throw a system exception. This has the effect of changing the exception which successive Interceptors popped from the Flow Stack receive on their calls to send\_exception. The exception thrown to the client will be the last exception thrown by an Interceptor, or the original exception if no Interceptor changes the exception.

This interception point may also throw a ForwardRequest exception. If an Interceptor throws this exception, no other Interceptors' send\_exception operations are called. The remaining Interceptors in the Flow Stack shall have their send\_other interception points called.

If the completion\_status of the exception is not COMPLETED\_NO, then it is inappropriate for this interception point to throw a ForwardRequest exception. The request's at-most-once semantics would be lost.

Compliant Interceptors shall properly follow completion\_status semantics if they throw a system exception from this interception point. If the original exception is a system exception, the completion\_status of the new exception shall be the same as on the original. If the original exception is a user exception, then the completion\_status of the new exception shall be COMPLETED\_YES.

**Parameters:**ri - Information about the current request being intercepted. **Throws:** [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html) - If thrown, indicates to the ORB that a retry of the request should occur with the new object given in the exception.

### send\_other

void **send\_other**([ServerRequestInfo](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInfo.html) ri)  
 throws [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html)

Allows an Interceptor to query the information available when a request results in something other than a normal reply or an exception. For example, a request could result in a retry (e.g., a GIOP Reply with a LOCATION\_FORWARD status was received). This interception point shall execute in the same thread as the target invocation.

This interception point may throw a system exception. If it does, no other Interceptors' send\_other operations are called. The remaining Interceptors in the Flow Stack shall have their send\_exception interception points called.

This interception point may also throw a ForwardRequest exception. If an Interceptor throws this exception, successive Interceptors' send\_other operations are called with the new information provided by the ForwardRequest exception.

Compliant Interceptors shall properly follow completion\_status semantics if they throw a system exception from this interception point. The completion\_status shall be COMPLETED\_NO.

**Parameters:**ri - Information about the current request being intercepted. **Throws:** [ForwardRequest](http://docs.google.com/org/omg/PortableInterceptor/ForwardRequest.html) - If thrown, indicates to the ORB that a retry of the request should occur with the new object given in the exception.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ServerRequestInterceptorOperations.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/org/omg/PortableInterceptor/ServerRequestInterceptor.html)   [**NEXT CLASS**](http://docs.google.com/org/omg/PortableInterceptor/SUCCESSFUL.html) | [**FRAMES**](http://docs.google.com/index.html?org/omg/PortableInterceptor/ServerRequestInterceptorOperations.html)    [**NO FRAMES**](http://docs.google.com/ServerRequestInterceptorOperations.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).