

Ice-Connection

Ice::Connection

Overview

local interface Connection

The user-level interface to a connection.

Used By

- [Ice::Current::con](#)

Operation Index

[close](#) — Close a connection, either gracefully or forcefully.
[createProxy](#) — Create a special proxy that always uses this connection.
[setAdapter](#) — Explicitly set an object adapter that dispatches requests that are received over this connection.
[getAdapter](#) — Get the object adapter that dispatches requests for this connection.
[getEndpoint](#) — Get the endpoint from which the connection was created.
[flushBatchRequests](#) — Flush any pending batch requests for this connection.
[type](#) — Return the connection type.
[timeout](#) — Get the timeout for the connection.
[toString](#) — Return a description of the connection as human readable text, suitable for logging or error messages.
[getInfo](#) — Returns the connection information.

Operations

void close(bool force)

Close a connection, either gracefully or forcefully. If a connection is closed forcefully, it closes immediately, without sending the relevant close connection protocol messages to the peer and waiting for the peer to acknowledge these protocol messages.

Parameters

`force` — If true, close forcefully. Otherwise the connection is closed gracefully.

Object* createProxy([Ice::Identity](#) id)

Create a special proxy that always uses this connection. This can be used for callbacks from a server to a client if the server cannot directly establish a connection to the client, for example because of firewalls. In this case, the server would create a proxy using an already established connection from the client.

Parameters

`id` — The identity for which a proxy is to be created.

Return Value

A proxy that matches the given identity and uses this connection.

See Also

- [setAdapter](#)

void setAdapter([Ice::ObjectAdapter](#) adapter)

Explicitly set an object adapter that dispatches requests that are received over this connection. A client can invoke an operation on a server using a proxy, and then set an object adapter for the outgoing connection that is used by the proxy in order to receive callbacks. This is useful if the server cannot establish a connection back to the client, for example because of firewalls.

Parameters

`adapter` — The object adapter that should be used by this connection to dispatch requests. The object adapter must be activated. When the object adapter is deactivated, it is automatically removed from the connection.

See Also

- [createProxy](#)
- [setAdapter](#)

Ice::ObjectAdapter `getAdapter()`

Get the object adapter that dispatches requests for this connection.

Return Value

The object adapter that dispatches requests for the connection, or null if no adapter is set.

See Also

- [setAdapter](#)

Ice::Endpoint `getEndpoint()`

Get the endpoint from which the connection was created.

Return Value

The endpoint from which the connection was created.

["async"] void flushBatchRequests()

Flush any pending batch requests for this connection. This causes all batch requests that were sent via proxies that use this connection to be sent to the server.

string type()

Return the connection type. This corresponds to the endpoint type, i.e., "tcp", "udp", etc.

Return Value

The type of the connection.

int timeout()

Get the timeout for the connection.

Return Value

The connection's timeout.

string toString()

Return a description of the connection as human readable text, suitable for logging or error messages.

Return Value

The description of the connection as human readable text.

Ice::ConnectionInfo `getInfo()`

Returns the connection information.

Return Value

The connection information.