

How Glacier2 uses Request Contexts

The Glacier2 router examines the [context](#) of an incoming request for special keys that affect how the router forwards the request. These contexts have the same semantics regardless of whether the request is sent from client to server or from server to client.

On this page:

- [The `_fwd` Context](#)
- [The `_ovrd` Context](#)
- [Forwarding Batch Requests](#)
- [Context Forwarding](#)

The `_fwd` Context

The `_fwd` context determines the proxy mode that the router uses when forwarding the request. The value associated with the `_fwd` key must be a string containing one or more of the characters shown in the following table:

Value	Mode
d	datagram
D	Batch datagram
o	Oneway
O	Batch oneway
s	Secure
t	Twoway
z	Compress

Legal values for the `_fwd` context key.

These characters match the corresponding [stringified proxy options](#).

For requests whose `_fwd` context specify a batch mode, the forwarding behavior of the router depends on whether it is [batching requests](#).

If the `_fwd` key is not present in a request context, the mode used by the router to forward that request depends on the mode used by the client's proxy as well as the router's own configuration:

- If the client used twoway mode, the router also uses twoway mode.
- If the client sent the request as a oneway or batch oneway, the router's behavior is determined by its [configuration properties](#).

The `_ovrd` Context

In buffered mode, the router allows a new incoming request to override any pending requests that are still in the router's queue, effectively replacing any pending requests with the new request.

For a new request to override a pending request, both requests must meet the following criteria:

- they specify the `_ovrd` key in the request context with the same value
- they are oneway requests
- they are requests on the same object.

This feature is intended to be used by clients that are sending frequent oneway requests in which the most recent request takes precedence. This feature minimizes the number of requests that are forwarded to the server when requests are sent frequently enough that they accumulate in the router's queue before the router has a chance to process them.

Note that the properties [Glacier2.Client.SleepTime](#) and [Glacier2.Server.SleepTime](#) can be used to add a delay to the router once it has sent all pending requests. Setting a delay increases the likelihood of overrides actually taking effect. These properties are described in the next section.

Forwarding Batch Requests

Clients can direct the router to forward oneway requests in batches by including the `D` or `O` characters in the `_fwd` context. If the router is configured for buffered mode and several such requests accumulate in its queue, the router forwards them together in a [batch](#) rather than as individual requests.

In addition, the properties `Glacier2.Client.AlwaysBatch` and `Glacier2.Server.AlwaysBatch` determine whether oneway requests are always batched regardless of the `_fwd` context. The former property affects requests from clients to servers, while the latter affects requests from servers to clients. If a property is defined with a non-zero value, then all requests whose `_fwd` context includes the `O` character or were sent as oneway invocations are treated as if `O` were specified instead, and are batched when possible. Likewise, requests whose `_fwd` context includes the `D` character or were sent as datagram invocations are treated as if `D` were specified. If a property is not defined, the router only batches requests if specifically directed to do so by the `_fwd` context.

The configuration properties `Glacier2.Client.SleepTime` and `Glacier2.Server.SleepTime` can be used to force the router's delivery threads to sleep for the specified number of milliseconds after the router has sent all of its pending requests. (Incoming requests are queued during this period.) The delay is useful to increase the effectiveness of batching because it makes it more likely for additional requests to accumulate in a batch before the batch is sent. If these properties are not defined, or their value is zero, the corresponding thread does not sleep after sending queued requests.

Context Forwarding

The configuration properties `Glacier2.Client.ForwardContext` and `Glacier2.Server.ForwardContext` determine whether the router includes the context when forwarding a request. The former property affects requests from clients to servers, while the latter affects requests from servers to clients. If a property is not defined or has the value zero, the router does not include the context when forwarding requests.

See Also

- [Request Contexts](#)
- [Proxy and Endpoint Syntax](#)
- [Batched Invocations](#)
- [Glacier2 Properties](#)