

# Ice Object Adapter Properties

On this page:

- `adapter.ACm`
- `adapter.AdapterId`
- `adapter.Endpoints`
- `adapter.Locator`
- `adapter.ProxyOptions`
- `adapter.PublishedEndpoints`
- `adapter.ReplicaGroupId`
- `adapter.Router`
- `adapter.ThreadPool.Serialize`
- `adapter.ThreadPool.Size`
- `adapter.ThreadPool.SizeMax`
- `adapter.ThreadPool.SizeWarn`
- `adapter.ThreadPool.StackSize`
- `adapter.ThreadPool.ThreadIdleTime`
- `adapter.ThreadPool.ThreadPriority`

## `adapter.ACm`

### Synopsis

```
adapter.ACm=num
```

### Description

If `num` is set to a value larger than zero, Active Connection Management (ACM) is enabled for the adapter, causing it to automatically close incoming connections after they have been idle for `num` seconds. The default value is the setting of `Ice.ACm.Server`.

Note that ACM can cause incoming oneway requests to be silently discarded.

## `adapter.AdapterId`

### Synopsis

```
adapter.AdapterId=id
```

### Description

Specifies an identifier for the object adapter with the name `adapter`. This identifier must be unique among all object adapters using the same `locator` instance. If a locator proxy is defined using `adapter.Locator` or `Ice.Default.Locator`, this object adapter sets its endpoints with the locator registry upon activation.

## `adapter.Endpoints`

### Synopsis

```
adapter.Endpoints=endpoints
```

### Description

Sets the `endpoints` for the object adapter `adapter` to `endpoints`. These endpoints specify the network interfaces on which the object adapter receives requests. Proxies created by the object adapter contain these endpoints, unless the `adapter.PublishedEndpoints` property is also specified.

## `adapter.Locator`

## Synopsis

```
adapter.Locator=locator
```

## Description

Specifies a [locator](#) for the object adapter with the name `adapter`. The value is a stringified proxy to the Ice locator interface.

As a proxy property, you can configure additional [aspects of the proxy](#) using properties.

## `adapter.ProxyOptions`

### Synopsis

```
adapter.ProxyOptions=options
```

### Description

Specifies the [proxy options](#) for proxies created by the object adapter. The value is a string representing the proxy options as they would be specified in a stringified proxy. The default value is `"-t"`, that is, proxies created by the object adapter are configured to use two-way invocations by default.

## `adapter.PublishedEndpoints`

### Synopsis

```
adapter.PublishedEndpoints=endpoints
```

### Description

When creating a proxy, the object adapter `adapter` normally includes the endpoints defined by `adapter.Endpoints`. If `adapter.PublishedEndpoints` is defined, the object adapter [publishes these endpoints](#) instead. This is useful in many situations, such as when a server resides behind a port-forwarding firewall, in which case the object adapter's public endpoints must specify the address and port of the firewall. The `adapter.ProxyOptions` property also influences the proxies created by an object adapter.

## `adapter.ReplicaGroupId`

### Synopsis

```
adapter.ReplicaGroupId=id
```

### Description

Identifies the group of [replicated object adapters](#) to which this adapter belongs. The replica group is treated as a virtual object adapter, so that an indirect proxy of the form `identity@id` refers to the object adapters in the group. During binding, a client will attempt to establish a connection to an endpoint of one of the participating object adapters, and automatically try others until a connection is successfully established or all attempts have failed. Similarly, an outstanding request will, when permitted, automatically fail over to another object adapter of the replica group upon connection failure. The set of endpoints actually used by the client during binding is determined by the locator's configuration policies.

Defining a value for this property has no effect unless `adapter.AdapterId` is also defined. Furthermore, the locator registry may require replica groups to be defined in advance (see [IceGrid.Registry.DynamicRegistration](#)), otherwise `Ice.NotRegisteredException` is raised upon adapter activation. Regardless of whether an object adapter is replicated, it can always be addressed individually in an indirect proxy if it defines a value for `adapter.AdapterId`.

## `adapter.Router`

### Synopsis

```
adapter.Router=router
```

## Description

Specifies a [router](#) for the object adapter with the name *adapter*. The value is a stringified proxy to the Ice router control interface. Defining a router allows the object adapter to receive callbacks from the router over a [bidirectional connection](#), thereby avoiding the need for the router to establish a connection back to the object adapter.

A router can only be assigned to one object adapter. Specifying the same router for more than one object adapter results in undefined behavior. The default value is no router.

As a proxy property, you can configure additional [aspects of the proxy](#) using properties.

## *adapter.ThreadPool.Serialize*

### Synopsis

```
adapter.ThreadPool.Serialize=num
```

### Description

If *num* is a value greater than zero, the adapter's thread pool serializes all messages from each connection. It is not necessary to enable this feature in a thread pool whose maximum size is one thread. In a [multi-threaded pool](#), enabling serialization allows requests from different connections to be dispatched concurrently while preserving the order of messages on each connection. Note that serialization has a significant impact on latency and throughput. If not defined, the default value is zero.

## *adapter.ThreadPool.Size*

### Synopsis

```
adapter.ThreadPool.Size=num
```

### Description

A communicator creates a default server thread pool that dispatches requests to its object adapters. An object adapter can also be configured with its own [thread pool](#). This is useful in avoiding deadlocks due to thread starvation by ensuring that a minimum number of threads is available for dispatching requests to certain Ice objects.

*num* is the initial number of threads in the thread pool. The default value is zero, meaning that an object adapter by default uses the communicator's server thread pool. See [Ice.ThreadPool.name.Size](#) for more information.

## *adapter.ThreadPool.SizeMax*

### Synopsis

```
adapter.ThreadPool.SizeMax=num
```

### Description

*num* is the maximum number of threads for the [thread pool](#). See [Ice.ThreadPool.name.SizeMax](#) for more information.

The default value is the value of [adapter.ThreadPool.Size](#), meaning the thread pool can never grow larger than its initial size.

## *adapter.ThreadPool.SizeWarn*

### Synopsis

```
adapter.ThreadPool.SizeWarn=num
```

### Description

Whenever *num* threads are active in a [thread pool](#), a "low on threads" warning is printed. The default value is zero, which disables the warning.

## adapter.ThreadPool.StackSize

### Synopsis

```
adapter.ThreadPool.StackSize=num
```

### Description

*num* is the stack size (in bytes) of threads in the [thread pool](#). The default value is zero, meaning the operating system's default is used.

## adapter.ThreadPool.ThreadIdleTime

### Synopsis

```
adapter.ThreadPool.ThreadIdleTime=num
```

### Description

In a dynamically-sized [thread pool](#), Ice reaps a thread after it is idle for *num* seconds. Setting this property to zero disables idle thread reaping. If not specified, the default value is 60 seconds. See [Ice.ThreadPool.name.ThreadIdleTime](#) for more information.

## adapter.ThreadPool.ThreadPriority

### Synopsis

```
adapter.ThreadPool.ThreadPriority=num
```

### Description

*num* specifies a thread priority for the object adapter's [thread pool](#). The object adapter creates its threads with the specified priority. Leaving this property unset causes the adapter to create threads with the priority specified by [Ice.ThreadPool.Server.ThreadPriority](#) or, if that property is unset, the priority specified by [Ice.ThreadPriority](#).