

Object Adapter Thread Pools



The default behavior of an [object adapter](#) is to share the [thread pools](#) of its communicator and, for many applications, this behavior is entirely sufficient. However, the ability to configure an object adapter with its own thread pool is useful in certain situations:

- When the concurrency requirements of an object adapter does not match those of its communicator. In a server with multiple object adapters, the configuration of the communicator's client and server thread pools may be a good match for some object adapters, but others may have different requirements. For example, the servants hosted by one object adapter may not support concurrent access, in which case limiting that object adapter to a single-threaded pool eliminates the need for synchronization in those servants. On the other hand, another object adapter might need a multi-threaded pool for better performance.
- To ensure that a minimum number of threads is available for dispatching requests to an adapter's servants. This is especially important for eliminating the possibility of deadlocks when using [nested invocations](#).

An object adapter's thread pool supports all of the properties described in [Configuring Thread Pools](#). For configuration purposes, the name of an adapter's thread pool is `adapter.ThreadPool`, where `adapter` is the name of the adapter.

An adapter creates its own thread pool when at least one of the following properties has a value greater than zero:

- `adapter.ThreadPool.Size`
- `adapter.ThreadPool.SizeMax`

These properties have the same semantics as those described earlier except they both have a default value of zero, meaning that an adapter uses the communicator's thread pools by default.

As an example, the properties shown below configure a thread pool for the object adapter named `PrinterAdapter`:

```
PrinterAdapter.ThreadPool.Size=3
PrinterAdapter.ThreadPool.SizeMax=15
PrinterAdapter.ThreadPool.SizeWarn=14
```

[Back to Top ^](#)

See Also

- [Thread Pools](#)
- [Object Adapters](#)
- [Nested Invocations](#)
- [Object Adapter Properties](#)

