

The Ice Threading Model



Ice is inherently a multi-threaded platform. There is no such thing as a single-threaded server in Ice. As a result, you must concern yourself with concurrency issues: if a thread reads a data structure while another thread updates the same data structure, havoc will ensue unless you protect the data structure with appropriate locks. In order to build Ice applications that behave correctly, it is important that you understand the threading semantics of the Ice run time. Here we discuss Ice's *thread pool* concurrency model and provide guidelines for writing thread-safe Ice applications.

Topics

- [Thread Pools](#)
- [Object Adapter Thread Pools](#)
- [Thread Pool Design Considerations](#)
- [Concurrent Proxy Invocations](#)
- [Nested Invocations](#)
- [Thread Safety](#)
- [Dispatching Requests to User Threads](#)
- [Blocking API Calls](#)

[Back to Top ^](#)

