

Dynamic Ice

The Ice streaming API allows you to serialize and deserialize Slice types using either the Ice encoding or an encoding of your choice (for example, XML). This is useful, for example, if you want to store Slice types in a data base.

The dynamic invocation and dispatch interfaces allow you to write generic clients and servers that need not have compile-time knowledge of the Slice types used by an application. This makes it possible to create applications such as object browsers, protocol analyzers, or protocol bridges. In addition, the dynamic invocation and dispatch interfaces permit services such as IceStorm to be implemented without the need to unmarshal and remarshal every message, with considerable performance improvements.

Keep in mind that applications that use dynamic invocation and dispatch are tedious to implement and harder to prove correct (because what normally would be a compile-time error appears only as a run-time error with dynamic invocation and dispatch). Therefore, you should use the dynamic interfaces only if your application truly benefits from this trade-off.

Topics

- [Streaming Interfaces](#)
- [Dynamic Invocation and Dispatch](#)
- [Asynchronous Dynamic Invocation and Dispatch](#)