Servant Locators

Previous Next

In a nutshell, a servant locator is a local object that you implement and attach to an object adapter. Once an adapter has a servant locator, it consults its active servant map (ASM) to locate a servant for an incoming request as usual. If a servant for the request can be found in the ASM, the request is dispatched to that servant. However, if the ASM does not have an entry for the object identity of the request, the object adapter calls back into the servant locator to ask it to provide a servant for the request. The servant locator either:

- locates an existing servant or instantiates a new servant and passes it to the Ice run time, in which case the request is dispatched to that servant, or
- the servant locator indicates failure to locate a servant to the Ice run time, in which case the client receives an `ObjectNotExistException`.

This simple mechanism allows us to scale servers to provide access to an unlimited number of Ice objects. For example, instead of instantiating a separate servant for each and every Ice object in existence, the server can instantiate servants only for those Ice objects that are actually used by clients. Similar to a Default Servant, the server can also instantiate only one single servant that the locator returns for a group of Ice objects with similar properties.

Servant locators are most commonly used by servers that provide access to databases: typically, the number of entries in the database is far larger than what the server can hold in memory. Servant locators allow the server to only instantiate servants for those Ice objects that are actually used by clients.

Another common use for servant locators is in servers that are used for process control or network management: in that case, there is no database but, instead, there is a potentially very large number of devices or network elements that must be controlled via the server. Otherwise, this scenario is the same as for large databases: the number of Ice objects exceeds the number of servants that the server can hold in memory and, therefore, requires an approach that allows the number of instantiated servants to be less than the number of Ice objects.

Topics

- The ServantLocator Interface
- Threading Guarantees for Servant Locators
- Registering a Servant Locator
- Servant Locator Example
- Using Identity Categories with Servant Locators
- Using Cookies with Servant Locators

Back to Top ^

See Also

- Object Adapters
- The Active Servant Map
- Object Identity
- Object Life Cycle
- Default Servants

Previous Next