

# IceGrid Slice API

---

## IceGrid

### Overview

#### module IceGrid

IceGrid is a server activation and deployment tool. IceGrid, simplifies the complex task of deploying applications in a heterogeneous computer network.

### Class Index

[AdaptiveLoadBalancingPolicy](#) — Adaptive load balancing policy.  
[BoxedDistributionDescriptor](#) — A "boxed" distribution descriptor.  
[BoxedString](#) — A "boxed" string.  
[CommunicatorDescriptor](#) — A communicator descriptor.  
[IceBoxDescriptor](#) — An IceBox server descriptor.  
[LoadBalancingPolicy](#) — A base class for load balancing policies.  
[OrderedLoadBalancingPolicy](#) — Ordered load balancing policy.  
[RandomLoadBalancingPolicy](#) — Random load balancing policy.  
[RoundRobinLoadBalancingPolicy](#) — Round robin load balancing policy.  
[ServerDescriptor](#) — An Ice server descriptor.  
[ServiceDescriptor](#) — An IceBox service descriptor.

### Interface Index

[AdapterObserver](#) — This interface allows applications to monitor the state of object adapters that are registered with IceGrid.  
[Admin](#) — The IceGrid administrative interface.  
[AdminSession](#) — Used by administrative clients to view, update, and receive observer updates from the IceGrid registry.  
[ApplicationObserver](#) — The database observer interface.  
[FileIterator](#) — This interface provides access to IceGrid log file contents.  
[FileParser](#) — `icegridadmin` provides a [FileParser](#) object to transform XML files into [ApplicationDescriptor](#) objects.  
[Locator](#) — The IceGrid locator interface provides access to the [Query](#) and [Registry](#) object of the IceGrid registry.  
[NodeObserver](#) — The node observer interface.  
[ObjectObserver](#) — This interface allows applications to monitor IceGrid well-known objects.  
[Query](#) — The IceGrid query interface.  
[Registry](#) — The IceGrid registry allows clients create sessions directly with the registry.  
[RegistryObserver](#) — This interface allows applications to monitor changes the state of the registry.  
[Session](#) — A session object is used by IceGrid clients to allocate and release objects.  
[UserAccountMapper](#) — A user account mapper object is used by IceGrid nodes to map session identifiers to user accounts.

### Exception Index

[AccessDeniedException](#) — This exception is raised if a registry lock wasn't acquired or is already held by a session.

[AdapterNotExistException](#) — This exception is raised if an adapter does not exist.

[AllocationException](#) — This exception is raised if the allocation of an object failed.

[AllocationTimeoutException](#) — This exception is raised if the request to allocate an object times out.

[ApplicationNotExistException](#) — This exception is raised if an application does not exist.

[BadSignalException](#) — This exception is raised if an unknown signal was sent to a server.

[DeploymentException](#) — An exception for deployment errors.

[FileNotAvailableException](#) — This exception is raised if a file is not available.

[NodeNotExistException](#) — This exception is raised if a node does not exist.

[NodeUnreachableException](#) — This exception is raised if a node could not be reached.

[ObjectExistsException](#) — This exception is raised if an object already exists.

[ObjectNotRegisteredException](#) — This exception is raised if an object is not registered.

[ObserverAlreadyRegisteredException](#) — This exception is raised if an observer is already registered with the registry.

[ParseException](#) — This exception is raised if an error occurs during parsing.

[PatchException](#) — This exception is raised if a patch failed.

[PermissionDeniedException](#) — This exception is raised if a client is denied the ability to create a session with IceGrid.

[RegistryNotExistException](#) — This exception is raised if a registry does not exist.

[RegistryUnreachableException](#) — This exception is raised if a registry could not be reached.

[ServerNotExistException](#) — This exception is raised if a server does not exist.

[ServerStartException](#) — This exception is raised if a server failed to start.

[ServerStopException](#) — This exception is raised if a server failed to stop.

[ServerUnreachableException](#) — This exception is raised if a server could not be reached.

[UserAccountNotFoundException](#) — This exception is raised if a user account for a given session identifier can't be found.

## Structure Index

[AdapterDescriptor](#) — An Ice object adapter descriptor.

[AdapterDynamicInfo](#) — Dynamic information about the state of an adapter.

[AdapterInfo](#) — Information about an adapter registered with the IceGrid registry.

[ApplicationDescriptor](#) — An application descriptor.

[ApplicationInfo](#) — Information about an IceGrid application.

[ApplicationUpdateDescriptor](#) — An application update descriptor to describe the updates to apply to a deployed application.

[ApplicationUpdateInfo](#) — Information about updates to an IceGrid application.

[DbEnvDescriptor](#) — A Freeze database environment descriptor.

[DistributionDescriptor](#) — A distribution descriptor defines an IcePatch2 server and the directories to retrieve from the patch server.

[LoadInfo](#) — Information about the load of a node.

[NodeDescriptor](#) — A node descriptor.

[NodeDynamicInfo](#) — Dynamic information about the state of a node.

[NodeInfo](#) — Information about an IceGrid node.

[NodeUpdateDescriptor](#) — A node update descriptor to describe the updates to apply to a node of a deployed application.

[ObjectDescriptor](#) — An Ice object descriptor.

[ObjectInfo](#) — Information about an Ice object.

[PropertyDescriptor](#) — Property descriptor.

[PropertySetDescriptor](#) — A property set descriptor.

[RegistryInfo](#) — Information about an IceGrid registry replica.

[ReplicaGroupDescriptor](#) — A replica group descriptor.

[ServerDynamicInfo](#) — Dynamic information about the state of a server.

[ServerInfo](#) — Information about a server managed by an IceGrid node.

[ServerInstanceDescriptor](#) — A server template instance descriptor.

[ServiceInstanceDescriptor](#) — A service template instance descriptor.

[TemplateDescriptor](#) — A template descriptor for server or service templates.

## Sequence Index

[AdapterDescriptorSeq](#) — A sequence of adapter descriptors.

[AdapterDynamicInfoSeq](#) — A sequence of adapter dynamic information structures.

[AdapterInfoSeq](#) — A sequence of adapter information structures.

[ApplicationDescriptorSeq](#) — A sequence of application descriptors.

[ApplicationInfoSeq](#) — A sequence of [ApplicationInfo](#) structures.

[DbEnvDescriptorSeq](#) — A sequence of database environment descriptors.

[NodeDynamicInfoSeq](#) — A sequence of node dynamic information structures.

[NodeUpdateDescriptorSeq](#) — A sequence of node update descriptors.

[ObjectDescriptorSeq](#) — A sequence of object descriptors.

[ObjectInfoSeq](#) — A sequence of object information structures.

[PropertyDescriptorSeq](#) — A sequence of property descriptors.

[RegistryInfoSeq](#) — A sequence of [RegistryInfo](#) structures.

[ReplicaGroupDescriptorSeq](#) — A sequence of replica groups.

[ServerDescriptorSeq](#) — A sequence of server descriptors.

[ServerDynamicInfoSeq](#) — A sequence of server dynamic information structures.

[ServerInstanceDescriptorSeq](#) — A sequence of server instance descriptors.

[ServiceDescriptorSeq](#) — A sequence of service descriptors.

[ServiceInstanceDescriptorSeq](#) — A sequence of service instance descriptors.

## Dictionary Index

[NodeDescriptorDict](#) — Mapping of node name to node descriptor.  
[PropertySetDescriptorDict](#) — A mapping of property set name to property set descriptor.  
[StringObjectProxyDict](#) — A dictionary of proxies.  
[StringStringDict](#) — A mapping of string to string.  
[TemplateDescriptorDict](#) — A mapping of template identifier to template descriptor.

## Enumeration Index

[LoadSample](#) — Determines which load sampling interval to use.  
[ServerState](#) — An enumeration representing the state of the server.

## Sequences

### [ "java:type:java.util.LinkedList<AdapterDescriptor>" ] sequence<[AdapterDescriptor](#)> AdapterDescriptorSeq

A sequence of adapter descriptors.

Used By

- [CommunicatorDescriptor::adapters](#)

### [ "java:type:java.util.LinkedList<AdapterDynamicInfo>" ] sequence<[AdapterDynamicInfo](#)> AdapterDynamicInfoSeq

A sequence of adapter dynamic information structures.

Used By

- [NodeDynamicInfo::adapters](#)

### sequence<[AdapterInfo](#)> AdapterInfoSeq

A sequence of adapter information structures.

Used By

- [AdapterObserver::adapterInit](#)
- [Admin::getAdapterInfo](#)

### [ "java:type:java.util.LinkedList<ApplicationDescriptor>" ] sequence<[ApplicationDescriptor](#)> ApplicationDescriptorSeq

A sequence of application descriptors.

### [ "java:type:java.util.LinkedList<ApplicationInfo>" ] sequence<[ApplicationInfo](#)> ApplicationInfoSeq

A sequence of [ApplicationInfo](#) structures.

Used By

- [ApplicationObserver::applicationInit](#)

### [ "java:type:java.util.LinkedList<DbEnvDescriptor>" ] sequence<[DbEnvDescriptor](#)> DbEnvDescriptorSeq

A sequence of database environment descriptors.

Used By

- [CommunicatorDescriptor::dbEnvs](#)

**sequence<[NodeDynamicInfo](#)> NodeDynamicInfoSeq**

A sequence of node dynamic information structures.

Used By

- [NodeObserver::nodeInit](#)

**[ "java:type:java.util.LinkedList<NodeUpdateDescriptor>" ] sequence<[NodeUpdateDescriptor](#)> NodeUpdateDescriptorSeq**

A sequence of node update descriptors.

Used By

- [ApplicationUpdateDescriptor::nodes](#)

**[ "java:type:java.util.LinkedList<ObjectDescriptor>" ] sequence<[ObjectDescriptor](#)> ObjectDescriptorSeq**

A sequence of object descriptors.

Used By

- [AdapterDescriptor::allocatables](#)
- [AdapterDescriptor::objects](#)
- [ReplicaGroupDescriptor::objects](#)

**sequence<[ObjectInfo](#)> ObjectInfoSeq**

A sequence of object information structures.

Used By

- [Admin::getAllObjectInfos](#)
- [Admin::getObjectInfosByType](#)
- [ObjectObserver::objectInit](#)

**[ "java:type:java.util.LinkedList<PropertyDescriptor>" ] sequence<[PropertyDescriptor](#)> PropertyDescriptorSeq**

A sequence of property descriptors.

Used By

- [DbEnvDescriptor::properties](#)
- [PropertySetDescriptor::properties](#)

**sequence<[RegistryInfo](#)> RegistryInfoSeq**

A sequence of [RegistryInfo](#) structures.

Used By

- [RegistryObserver::registryInit](#)

**[ "java:type:java.util.LinkedList<ReplicaGroupDescriptor>" ] sequence<[ReplicaGroupDescriptor](#)> ReplicaGroupDescriptorSeq**

A sequence of replica groups.

Used By

- [ApplicationDescriptor::replicaGroups](#)
- [ApplicationUpdateDescriptor::replicaGroups](#)

**[ "java:type:java.util.LinkedList<ServerDescriptor>" ] sequence<[ServerDescriptor](#)> ServerDescriptorSeq**

A sequence of server descriptors.

Used By

- [NodeDescriptor::servers](#)
- [NodeUpdateDescriptor::servers](#)

**[ "java:type:java.util.LinkedList<ServerDynamicInfo>" ] sequence<[ServerDynamicInfo](#)> ServerDynamicInfoSeq**

A sequence of server dynamic information structures.

Used By

- [NodeDynamicInfo::servers](#)

**[ "java:type:java.util.LinkedList<ServerInstanceDescriptor>" ] sequence<[ServerInstanceDescriptor](#)> ServerInstanceDescriptorSeq**

A sequence of server instance descriptors.

Used By

- [NodeDescriptor::serverInstances](#)
- [NodeUpdateDescriptor::serverInstances](#)

**[ "java:type:java.util.LinkedList<ServiceDescriptor>" ] sequence<[ServiceDescriptor](#)> ServiceDescriptorSeq**

A sequence of service descriptors.

**[ "java:type:java.util.LinkedList<ServiceInstanceDescriptor>" ] sequence<[ServiceInstanceDescriptor](#)> ServiceInstanceDescriptorSeq**

A sequence of service instance descriptors.

Used By

- [IceBoxDescriptor::services](#)

## Dictionaries

**dictionary<string, [NodeDescriptor](#)> NodeDescriptorDict**

Mapping of node name to node descriptor.

Used By

- [ApplicationDescriptor::nodes](#)

**dictionary<string, [PropertySetDescriptor](#)> PropertySetDescriptorDict**

A mapping of property set name to property set descriptor.

Used By

- [ApplicationDescriptor::propertySets](#)
- [ApplicationUpdateDescriptor::propertySets](#)
- [NodeDescriptor::propertySets](#)
- [NodeUpdateDescriptor::propertySets](#)
- [ServerInstanceDescriptor::servicePropertySets](#)

**dictionary<string, Object\*> StringObjectProxyDict**

A dictionary of proxies.

**dictionary<string, string> StringStringDict**

A mapping of string to string.

Used By

- [ApplicationDescriptor::variables](#)
- [ApplicationUpdateDescriptor::variables](#)
- [NodeDescriptor::variables](#)
- [NodeUpdateDescriptor::variables](#)
- [ServerInstanceDescriptor::parameterValues](#)
- [ServiceInstanceDescriptor::parameterValues](#)
- [TemplateDescriptor::parameterDefaults](#)

**dictionary<string, [TemplateDescriptor](#)> TemplateDescriptorDict**

A mapping of template identifier to template descriptor.

Used By

- [ApplicationDescriptor::serverTemplates](#)
  - [ApplicationDescriptor::serviceTemplates](#)
  - [ApplicationUpdateDescriptor::serverTemplates](#)
  - [ApplicationUpdateDescriptor::serviceTemplates](#)
-