

LIFECYCLE OF GRID SERVICE INSTANCE

Life Cycle of a Grid Service Instance

The hosting environment manages the lifecycle of a grid service by determining how and when a service is created, and how and when a service is destroyed. It is important to understand that the OGSF specification does not direct that behavior. For example, a grid service based on an Enterprise Java Bean (EJB) has the same lifecycle properties and manageability interfaces and policies as defined by that EJB specification.

The following describes lifecycle service creation patterns, as defined by the OGSF, to assist the clients of a grid service:

- A common grid service creation pattern is defined through the grid service factory interface (as discussed in the next sections) and through a createService operation. The service can decide whether or not to support this behavior, based upon the respective policies defined for the service.
- Two destruction mechanisms are defined:
 1. Calling an explicit destruction operation (i.e., destroy the operation on a specific GridService portType)
 2. Using a service-supported soft-state mechanism, based on the termination time attribute of a service

Service Lifecycle Management Using a Soft-State Approach

The soft-state lifetime management approach is a recommended method in the grid service lifecycle management process. Every grid service has a termination time set by the service creator or factory. A client device with appropriate authorization can use this termination time information to check the availability (i.e., lease period) of the service, and can request to extend the current lease time by sending a keep-alive message to the service with a new termination time. If the service accepts this request, the lease time can be extended to the new termination time requested by the client.

This soft-state lifecycle is controlled by appropriate security and policy decisions of the service, and the service has the authority to control this behavior. For example, a service can arbitrarily terminate a service, or a service can extend its termination time, even while the client holds a service reference. Another point to keep in mind is whether or not to actually destroy a service, or just to make it unavailable. This is a service hosting environment decision. Some of the behaviors that we can infer from the specification are:

- A grid service can send lifetime event notifications using the standard grid service notification process.
- A grid service's support for the lifetime is implementation dependent, and one must consult the service documentation for the details on this behavior.
- A Client with proper authority can request for an early termination of a grid service.
- A grid service can extend the service termination time, or terminate itself, at any time during its lifecycle.

Service Operation Extensibility Features of Grid Services

Another interesting feature introduced by the specification is the concept of extensible operations through the use of untyped parameters using the XML schema `xsd:any` construct. This feature allows grid service operations maximum flexibility by allowing "any" parameters (i.e., XML fragments) for that operation. However, this feature introduces confusion on the client side regarding the kind of parameters it can pass to the service. To avoid that confusion, the specification provides the client base with a mechanism (i.e., query capabilities) to ask the service about the supported extensible parameter and its type (XML schema) information.

This example will make the preceding idea clearer. We have already discussed that we can use `findServiceData` of `GridService` to query the service data of a service instance. A service can support different types of queries, including query by service data names and query by XPath. The specification defines the input parameter of `findServiceData` as extensible, utilizing `ogsi:ExtensibilityType`. In this example, one can retrieve all the possible queries supported by the service instance from the SDE values of the service's "findserviceDataExtensibility" service data element. By default, a service provides a static query type value called "queryByServiceDataNames" defined by utilizing the `staticServiceDataValue`. These static SDE values are defined for `GridService portType` in the `ogsi.gwsdl`. We will see more details on the operation extensibility features when we discuss the grid service portTypes.