

RELATIONSHIP AND DEPENDENCY AMONG RESOURCES

There may oftentimes be relationships existing among many instances of manageable resources. This relationship model is not specific to the CMM; the service relationships exist in the grid and Web service environments. Let us take a look at these ideas in order to better familiarize ourselves with the two core concepts here:

1. Relationships describe which resources are connected to each other and what type of connection exists; however, they do not describe the details of how one resource depends on the other.
2. Dependencies add additional information to the relationship on exactly how one resource depends on another. For example, a database resource indicates that it uses a storage device and provides more details on the needs such as storage space.

We will begin our discussion on the CMM relationship model with a simple example on relationships. Processes are created by operating system and the operating system is hosted by the computer system, which is a part of a cluster. CMM provides mechanisms to model these types of resource relationships (see Table 7.2).

In CMM, the `BaseManageableResource` port type provides two service data definitions (`relatedInstance` and `relatedType`) to deal with a relationship.

Table 7.2. CMM-defined relationship types.

Relationship Type	Description
--------------------------	--------------------

Hosts	Any resource "A" hosts another resource "B" if resource "A" provides an environment in which resource "B" is created and runs. The lifecycle of resource "B" is a subset of the lifecycle of resource "A" and resource "B" cannot exist without resource "A." For example, a database hosts the table within it.
Contains	Any resource may consist of a number of other resources. Any contained resource has the same lifetime as the containing resource. If resource "A"

Table 7.2. CMM-defined relationship types.

Relationship Type	Description
	contains resource "B," then if "A" installs "A," "B" gets installed and if "A" stopped "B" gets stopped. For example, a deployed J2EE application containing various modules.
Federates	Any numbers of resources are in different hosting environments and are utilized together to form another resource. For example, an application includes a database and queue and they do not know each other but work together in the application.
Aggregates	A number of resources are grouped together. For example, a resource that represents all computers in a department.
Uses	A resource uses another resource. It is different from federates. For example, a security system uses a LDAP registry to hold user information.
Implements	One resource is utilized to implement the function of another. For example, a database server is implemented as a Linux or Windows service.

These types of relationships exist in current programming environments and are explained quite well by the UML relationship model and dependency graphs. We believe that this CMM-specific relationship model should be elevated to the grid service and Web service worlds, with the appropriate modifications.

We have now discussed the new concepts and canonical information provided by CMM. The resource's manageability information can be implemented using any of the existing management instrumentation methods, such as Common Information Model (CIM), Simple Network Management Protocol (SNMP), and Lightweight Directory Access Protocol (LDAP).

The CMM resource model and management grid service are independent of the underlying service implementation and resource instrumentation. One important question that may come across is: What is the value CMM provides over a normal service interface to any existing resource instrumentation models? The answer lies in the fact that CMM is not just an algorithmic mapping from a grid service to resource instrumentation. Instead, CMM contains a more behavior-specific and self-contained resource management model. Also, the same resource model may map to multiple instrumentation choices, and this is a binding choice.