211
DBMS_UMF

The `DBMS_UMF` package provides an interface for deploying the Remote Management Framework (RMF) for an Oracle Database. The RMF is used for collecting performance statistics for an Oracle Database.

> **See Also:**
>
> *Oracle Database Performance Tuning Guide* for more information about configuring the RMF for an Oracle Database.

This chapter contains the following topic:

- Summary of DBMS_UMF Subprograms

Summary of DBMS_UMF Subprograms

This topic lists the `DBMS_UMF` subprograms in alphabetical order and briefly describes them.

Table 211-1 DBMS_UMF Package Subprograms

Subprogram	Description
CONFIGURE_NODE Procedure	Configures a node in the RMF topology
CREATE_LINK Procedure	Creates a database link between two nodes in the RMF topology
CREATE_TOPOLOGY Procedure	Creates the RMF topology
DROP_LINK Procedure	Removes a database link between two nodes in the RMF topology
DROP_TOPOLOGY Procedure	Deletes the RMF topology
ENABLE_SERVICE Procedure	Enables a service on a node in the RMF topology
GET_NODE_ID_LOCAL Function	Returns the node ID of a node in the RMF topology
GET_NODE_NAME_LOCAL Function	Returns the node name of a node in the RMF topology
GET_TARGET_ID Function	Returns the destination ID in the RMF topology
GET_TOPOLOGY_NAME_LOCAL Function	Returns the RMF topology name of the local node
QUERY_LINK_INFO Procedure	Returns the information about a database link in the RMF topology
QUERY_NODE_INFO Procedures	Returns the information about a node in the RMF topology
REGISTER_NODE Function and Procedure	Registers a node in the RMF topology
SWITCH_DESTINATION Procedure	Designates a source node as a destination node in the RMF topology

Table 211-1 (Cont.) DBMS_UMF Package Subprograms

Subprogram	Description
UNCONFIGURE_NODE Procedure	Resets the configuration of a node in the RMF topology
UNREGISTER_NODE Procedure	Removes the registration of a node in the RMF topology

CONFIGURE_NODE Procedure

This procedure configures a node that needs to be registered with the RMF topology. This procedure must be executed on the node that needs to be configured.

Syntax

```
DBMS_UMF.CONFIGURE_NODE(
   node_name          IN VARCHAR2 DEFAULT NULL,
   dblink_to_target   IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 211-2 CONFIGURE_NODE Procedure Parameters

Parameter	Description
node_name	Name of the node. Each node in the RMF topology must be assigned a unique name. If a name is not provided for a node in this procedure, then the value of the initialization parameter DB_UNIQUE_NAME is assigned as the name for the node by default.
dblink_to_target	Database link from this node to the destination node.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

CREATE_LINK Procedure

This procedure creates database links between two nodes in the RMF topology. This procedure must be executed only on the destination node.

Syntax

```
DBMS_UMF.CREATE_LINK(
   topology_name      IN VARCHAR2,
   node_a_name        IN VARCHAR2,
   node_b_name        IN VARCHAR2,
   dblink_a_to_b      IN VARCHAR2,
   dblink_b_to_a      IN VARCHAR2);
```

Parameters

Table 211-3 CREATE_LINK Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology.
node_a_name	Name of the first node.
node_b_name	Name of the second node.
dblink_a_to_b	Database link from the first node to the second node.
dblink_b_to_b	Database link from the second node to the first node.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

CREATE_TOPOLOGY Procedure

This procedure creates the RMF topology and designates the node on which it is executed as the destination node for that topology.

Syntax

```
DBMS_UMF.CREATE_TOPOLOGY(
   topology_name IN VARCHAR2);
```

Parameters

Table 211-4 CREATE_TOPOLOGY Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

DROP_LINK Procedure

This procedure removes the database links between two nodes in the RMF topology. This procedure must be executed only on the destination node.

Syntax

```
DBMS_UMF.DROP_LINK(
   topology_name  IN VARCHAR2,
   node_a_name    IN VARCHAR2,
   node_b_name    IN VARCHAR2);
```

Parameters

Table 211-5 DROP_LINK Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology.
node_a_name	Name of the first node.
node_b_name	Name of the second node.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

DROP_TOPOLOGY Procedure

This procedure deletes the RMF topology. This procedure must be executed only on the destination node.

Syntax

```
DBMS_UMF.DROP_TOPOLOGY(
   topology_name IN VARCHAR2);
```

Parameters

Table 211-6 DROP_TOPOLOGY Procedure Parameters

Parameter	Description
topology_name	Name of the topology to delete.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

ENABLE_SERVICE Procedure

This procedure enables a service, such as the AWR service, on a node in the RMF topology. This procedure must be executed only on the destination node.

Syntax

```
DBMS_UMF.ENABLE_SERVICE(
   topology_name      IN VARCHAR2,
   node_name          IN VARCHAR2,
   service_type       IN NUMBER);
```

Parameters

Table 211-7 ENABLE_SERVICE Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology.
node_name	Name of the node on which a specific service needs to be enabled.
service_type	Numeric constant identifying the service. The only allowed value for this parameter is UMF_SERVICE_TYPE_AWR, which is Automatic Workload Repository (AWR) service.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

GET_NODE_ID_LOCAL Function

This function returns the node ID of the node in the RMF topology on which this function is executed.

Syntax

```
DBMS_UMF.GET_NODE_ID_LOCAL(
   topology_name  IN  VARCHAR2 DEFAULT NULL)
 RETURN NUMBER;
```

Parameters

Table 211-8 GET_NODE_ID_LOCAL Function Parameters

Parameter	Description
topology_name	Name of the RMF topology with which the node is registered.

Return Value

Returns the node ID of the node in the RMF topology on which this function is executed.

Security Model

Only the database users SYS and SYS$UMF can execute this function.

GET_NODE_NAME_LOCAL Function

This function returns the name of the node in the RMF topology on which this function is executed.

Syntax

```
DBMS_UMF.GET_NODE_NAME_LOCAL RETURN VARCHAR2;
```

Return Value

Returns the name of the node in the RMF topology on which this function is executed.

Security Model

Only the database users `SYS` and `SYS$UMF` can execute this procedure.

GET_TARGET_ID Function

This function returns the ID of the destination node in the RMF topology. This function can be executed on any node in the RMF topology.

Syntax

```
DBMS_UMF.GET_TARGET_ID(
   topology_name  IN  VARCHAR2)
 RETURN NUMBER;
```

Parameters

Table 211-9 GET_TARGET_ID Function Parameters

Parameter	Description
`topology_name`	Name of the RMF topology.

Return Value

Returns the ID of the destination node in the RMF topology.

Security Model

Only the database users `SYS` and `SYS$UMF` can execute this procedure.

GET_TOPOLOGY_NAME_LOCAL Function

This function returns the name of the active RMF topology of the node on which this function is executed.

Syntax

```
DBMS_UMF.GET_TOPOLOGY_NAME_LOCAL RETURN VARCHAR2;
```

Return Value

Returns the name of the active RMF topology of the node on which this function is executed.

Security Model

Only the database users `SYS` and `SYS$UMF` can execute this procedure.

QUERY_LINK_INFO Procedure

This procedure returns the name of the database link between two nodes in the RMF topology. This procedure can be executed on any node in the RMF topology.

Syntax

```
DBMS_UMF.QUERY_LINK_INFO(
   topology_name      IN   VARCHAR2,
   from_node_id       IN   NUMBER,
   to_node_id         IN   NUMBER,
   link_name          OUT  VARCHAR2);
```

Parameters

Table 211-10 QUERY_LINK_INFO Procedure Parameters

Parameter	Description
`topology_name`	Name of the RMF topology.
`from_node_id`	Node ID of the first node.
`to_node_id`	Node ID of the second node.
`link_name`	Name of the database link from the first node to the second node returned by the procedure.

Security Model

Only the database users `SYS` and `SYS$UMF` can execute this procedure.

QUERY_NODE_INFO Procedures

This procedure returns information about a node in the RMF topology. This procedure can be executed on any node in the RMF topology.

Syntax

```
DBMS_UMF.QUERY_NODE_INFO(
   topology_name      IN    VARCHAR2  DEFAULT NULL,
   node_name          IN    VARCHAR2,
   node_id            OUT   NUMBER);

DBMS_UMF.QUERY_NODE_INFO(
   node_id            IN    NUMBER,
```

```
   topology_name       OUT   VARCHAR2,
   node_name           OUT   VARCHAR2);
```

Parameters

Table 211-11 QUERY_NODE_INFO Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology with which the node is registered.
node_name	Name of the node.
node_id	Identifier of the node.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

REGISTER_NODE Function and Procedure

This function and procedure registers a node with the RMF topology. This procedure and function must be executed only on the destination node in the RMF topology.

Syntax

```
DBMS_UMF.REGISTER_NODE(
   topology_name           IN  VARCHAR2,
   node_name               IN  VARCHAR2,
   dblink_to_node          IN  VARCHAR2 DEFAULT NULL,
   dblink_from_node        IN  VARCHAR2 DEFAULT NULL,
   as_source               IN  VARCHAR2 DEFAULT 'TRUE',
   as_candidate_target     IN  VARCHAR2 DEFAULT 'FALSE');

DBMS_UMF.REGISTER_NODE(
   topology_name           IN  VARCHAR2,
   node_name               IN  VARCHAR2,
   dblink_to_node          IN  VARCHAR2 DEFAULT NULL,
   dblink_from_node        IN  VARCHAR2 DEFAULT NULL,
   as_source               IN  VARCHAR2 DEFAULT 'TRUE',
   as_candidate_target     IN  VARCHAR2 DEFAULT 'FALSE');
   node_id                OUT VARCHAR2);

DBMS_UMF.REGISTER_NODE(
   topology_name           IN  VARCHAR2,
   node_name               IN  VARCHAR2,
   dblink_to_node          IN  VARCHAR2 DEFAULT NULL,
   dblink_from_node        IN  VARCHAR2 DEFAULT NULL,
   as_source               IN  VARCHAR2 DEFAULT 'TRUE',
   as_candidate_target     IN  VARCHAR2 DEFAULT 'FALSE')
 RETURN NUMBER;
```

Parameters

Table 211-12 REGISTER_NODE Function and Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology.
node_name	Name of the node to register.
dblink_to_node	Name for the database link from the destination to the node.
dblink_from_node	Name for the database link from the node to the destination.
as_source	Set to TRUE, if the node is a source, else set to FALSE.
as_candidate_target	Set to TRUE, if the node is a candidate destination, else set to FALSE.
node_id	Node ID returned by the procedure.

Return Value

Returns the node ID of the registered node.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

SWITCH_DESTINATION Procedure

This procedure makes the candidate destination as the new destination in the RMF topology. This procedure must be executed only on the candidate destination node.

Syntax

```
DBMS_UMF.SWITCH_DESTINATION(
   topology_name      IN VARCHAR2,
   force_switch       IN BOOLEAN DEFAULT TRUE);
```

Parameters

Table 211-13 SWITCH_DESTINATION Procedure Parameters

Parameter	Description
topology_name	Name of the RMF topology.

Table 211-13 (Cont.) SWITCH_DESTINATION Procedure Parameters

Parameter	Description
force_switch	If set to FALSE, the execution of this procedure fails in the following situations: • Candidate destination is a read-only database, and hence it cannot become the new destination. • Candidate destination does not have database links to one or more sources in the topology. • Candidate destination is unable to get the latest AWR data from the old destination. If set to TRUE, the execution of this procedure fails in the following situation: • Candidate destination is a read-only database, and hence it cannot become the new destination.

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

UNCONFIGURE_NODE Procedure

This procedure removes the configuration details of the node on which this procedure is executed.

Syntax

```
DBMS_UMF.UNCONFIGURE_NODE;
```

Security Model

Only the database users SYS and SYS$UMF can execute this procedure.

UNREGISTER_NODE Procedure

This procedure removes a node for the RMF topology. This procedure must be executed only on the destination node.

Syntax

```
DBMS_UMF.UNREGISTER_NODE(
   topology_name    IN VARCHAR2,
   node_name        IN VARCHAR2);
```

Parameters

Table 211-14 UNREGISTER_NODE Procedure Parameters

Parameter	Description
`topology_name`	Name of the RMF topology.
`node_name`	Name of the node which needs to be removed from the topology.

Security Model

Only the database users `SYS` and `SYS$UMF` can execute this procedure.