156

DBMS_PROPAGATION_ADM

The DBMS_PROPAGATION_ADM package, one of a set of Oracle Replication packages, provides administrative interfaces for configuring a propagation from a source queue to a destination queue.

This chapter contains the following topics:

- Overview
- Security Model
- Summary of DBMS_PROPAGATION_ADM Subprograms

DBMS_PROPAGATION_ADM Overview

This package provides interfaces to start, stop, and configure a propagation.

DBMS_PROPAGATION_ADM Security Model

Security on this package can be controlled by granting EXECUTE on this package to selected users or roles, or by granting EXECUTE CATALOG ROLE to selected users or roles.

If subprograms in the package are run from within a stored procedure, then the user who runs the subprograms must be granted EXECUTE privilege on the package directly. It cannot be granted through a role.

When the DBMS_PROPAGATION_ADM package is used to manage an Oracle Replication configuration, it requires that the user is granted the privileges of an Oracle Replication administrator.

Summary of DBMS_PROPAGATION_ADM Subprograms

This table lists the DBMS PROPAGATION ADM subprograms and briefly describes them.

Table 156-1 DBMS_PROPAGATION_ADM Package Subprograms

Subprogram	Description
ALTER_PROPAGATION Procedure	Adds, alters, or removes a rule set for a propagation
CREATE_PROPAGATION Procedure	Creates a propagation and specifies the source queue, destination queue, and rule set for the propagation
DROP_PROPAGATION Procedure	Drops a propagation
START_PROPAGATION Procedure	Starts a propagation
STOP_PROPAGATION Procedure	Stops a propagation



All subprograms commit unless specified otherwise.

ALTER_PROPAGATION Procedure

This procedure adds, alters, or removes a rule set for a propagation.

Syntax

```
DBMS_PROPAGATION_ADM.ALTER_PROPAGATION(
propagation_name IN VARCHAR2,
rule_set_name IN VARCHAR2 DEFAULT NULL,
remove_rule_set IN BOOLEAN DEFAULT FALSE,
negative_rule_set_name IN VARCHAR2 DEFAULT NULL,
remove negative rule set IN BOOLEAN DEFAULT FALSE);
```

Parameters

Table 156-2 ALTER_PROPAGATION Procedure Parameters

Parameter	Description
propagation_name	The name of the propagation you are altering. You must specify an existing propagation name. Do not specify an owner.
rule_set_name	The name of the positive rule set for the propagation. The positive rule set contains the rules that instruct the propagation to propagate messages.
	If you want to use a positive rule set for the propagation, then you must specify an existing rule set in the form [schema_name.]rule_set_name. For example, to specify a positive rule set in the hr schema named prop_rules, enter hr.prop_rules. If the schema is not specified, then the current user is the default.
	An error is returned if the specified rule set does not exist. You can create a rule set and add rules to it using the DBMS_RULE_ADM package.
	If you specify NULL and the remove_rule_set parameter is set to FALSE, then the procedure retains any existing positive rule set If you specify NULL and the remove_rule_set parameter is set to TRUE, then the procedure removes any existing positive rule set.
remove_rule_set	If TRUE, then the procedure removes the positive rule set for the specified propagation. If you remove a positive rule set for a propagation, and the propagation does not have a negative rule set, then the propagation propagates all messages.
	If you remove a positive rule set for a propagation, and a negative rule set exists for the propagation, then the propagation propagates all messages in its queue that are not discarded by the negative rule set.
	If FALSE, then the procedure retains the positive rule set for the specified propagation.
	If the rule_set_name parameter is non-NULL, then this parameter should be set to FALSE.

Table 156-2 (Cont.) ALTER_PROPAGATION Procedure Parameters

Parameter	Description
negative_rule_set_name	The name of the negative rule set for the propagation. The negative rule set contains the rules that instruct the propagation to discard messages.
	If you want to use a negative rule set for the propagation, then you must specify an existing rule set in the form [schema_name.]rule_set_name. For example, to specify a negative rule set in the hr schema named neg_rules, enter hr.neg_rules. If the schema is not specified, then the current user is the default.
	An error is returned if the specified rule set does not exist. You can create a rule set and add rules to it using the DBMS_RULE_ADM package.
	If you specify NULL and the remove_negative_rule_set parameter is set to FALSE, then the procedure retains any existing negative rule set. If you specify NULL and the remove_negative_rule_set parameter is set to TRUE, then the procedure removes any existing negative rule set.
	If you specify both a positive and a negative rule set for a propagation, then the negative rule set is always evaluated first.
remove_negative_rule_set	If TRUE, then the procedure removes the negative rule set for the specified propagation. If you remove a negative rule set for a propagation, and the propagation does not have a positive rule set, then the propagation propagates all messages.
	If you remove a negative rule set for a propagation, and a positive rule set exists for the propagation, then the propagation propagates all messages in its queue that are not discarded by the positive rule set.
	If FALSE, then the procedure retains the negative rule set for the specified propagation.
	If the negative_rule_set_name parameter is non-NULL, then this parameter should be set to FALSE.

CREATE_PROPAGATION Procedure

This procedure creates a propagation and specifies the source queue, destination queue, and any rule set for the propagation. A propagation propagates messages in a local source queue to a destination queue. The destination queue might or might not be in the same database as the source queue.

Syntax



Parameters

Table 156-3 CREATE_PROPAGATION Procedure Parameters

Parameter	Description
propagation_name	The name of the propagation you are creating. A <code>NULL</code> setting is not allowed. Do not specify an owner.
	Note: The propagation_name setting cannot be altered after the propagation is created.
source_queue	The name of the source queue, specified as [schema_name.] queue_name. The current database must contain the source queue.
	For example, to specify a source queue named streams_queue in the strmadmin schema, enter strmadmin.streams_queue for this parameter. If the schema is not specified, then the current user is the default.
destination_queue	The name of the destination queue, specified as [schema_name.] queue_name.
	For example, to specify a destination queue named streams_queue in the strmadmin schema, enter strmadmin.streams_queue for this parameter. If the schema is not specified, then the current user is the default.
destination_dblink	The name of the database link that will be used by the propagation. The database link is from the database that contains the source queue to the database that contains the destination queue.
	If $\mathtt{NULL},$ then the source queue and destination queue must be in the same database.
	Note: Connection qualifiers are not allowed.
rule_set_name	The name of the positive rule set for the propagation. The positive rule set contains the rules that instruct the propagation to propagate messages.
	If you want to use a positive rule set for the propagation, then you must specify an existing rule set in the form [schema_name.]rule_set_name. For example, to specify a positive rule set in the hr schema named prop_rules, enter hr.prop_rules. If the schema is not specified, then the current user is the default.
	An error is returned if the specified rule set does not exist. You can create a rule set and add rules to it using the DBMS_RULE_ADM package.
	If you specify NULL, and no negative rule set exists for the propagation, then the propagation propagates all messages in its queue.
	If you specify \mathtt{NULL} , and a negative rule set exists for the propagation, then the propagation propagates all messages in its queue that are not discarded by the negative rule set.



Table 156-3 (Cont.) CREATE_PROPAGATION Procedure Parameters

Parameter	Description
negative_rule_set_name	The name of the negative rule set for the propagation. The negative rule set contains the rules that instruct the propagation to discard messages.
	If you want to use a negative rule set for the propagation, then you must specify an existing rule set in the form [schema_name.]rule_set_name. For example, to specify a negative rule set in the hr schema named neg_rules, enter hr.neg_rules. If the schema is not specified, then the current user is the default.
	An error is returned if the specified rule set does not exist. You can create a rule set and add rules to it using the DBMS_RULE_ADM package.
	If you specify NULL, and no positive rule set exists for the propagation, then the propagation propagates all messages in its queue.
	If you specify \mathtt{NULL} , and a positive rule set exists for the propagation, then the propagation propagates all messages in its queue that are not discarded by the positive rule set.
	If you specify both a positive and a negative rule set for a propagation, then the negative rule set is always evaluated first.
queue_to_queue	If TRUE or NULL, then the propagation is a queue to queue propagation. A queue-to-queue propagation always has its own propagation job and uses a service for automatic failover when the destination queue is a buffered queue in an Oracle Real Application Clusters (Oracle RAC) database.
	If FALSE, then the propagation is a queue-to-dblink propagation. A queue-to-dblink propagation can share a propagation job with other propagations that use the same database link and does not support automatic failover in an Oracle RAC environment.
original_propagation_name	Specify the original propagation name if the propagation being created is part of a split and merge operation. The split operation clones the original propagation under a new name. The name of the original propagation is important when the cloned propagation is copied back to the original stream.
	Specify \mathtt{NULL} if the propagation being created is not part of a split and merge operation.
auto_merge_theshold	Specify a positive number if both of the following conditions are met:
	 The propagation being created is part of a split and merge operation.
	The stream will be merged back to the original stream automatically.
	Specify NULL if either of the following conditions are met:
	 The propagation being created is not part of a split and merge operation.
	 The propagation being created is part of a split and merge operation, but the stream being split off will not be merged back to the original stream automatically.



Usage Notes

This procedure starts propagation and might create a propagation job. If this procedure creates a propagation job, then it establishes a default schedule for the propagation job. Each propagation job is an Oracle Scheduler job. You can adjust the schedule of a propagation job using Oracle Scheduler.

The user who owns the source queue is the user who propagates messages. This user must have the necessary privileges to propagate messages.

```
See Also:

DBMS_RULE_ADM
```

DROP_PROPAGATION Procedure

This procedure drops a propagation and deletes all messages for the destination queue in the source queue. This procedure also removes the schedule for propagation from the source queue to the destination queue.

Syntax

Parameters

Table 156-4 DROP_PROPAGATION Procedure Parameters

Parameter	Description
propagation_name	The name of the propagation you are dropping. You must specify an existing propagation name. Do not specify an owner.
drop_unused_rule_sets	If TRUE, then the procedure drops any rule sets, positive and negative, used by the specified propagation if these rule sets are not used by any other Oracle Replication client, which includes capture processes, propagations, apply processes, and messaging clients. If this procedure drops a rule set, then this procedure also drops any rules in the rule set that are not in another rule set.
	If FALSE, then the procedure does not drop the rule sets used by the specified propagation, and the rule sets retain their rules.

Usage Notes

When you use this procedure to drop a propagation, information about rules created for the propagation is removed from the data dictionary views for Oracle Replication rules. Information about such a rule is removed even if the rule is not in either rule set for the propagation.

The following are the data dictionary views for Oracle Replication rules:

- ALL STREAMS GLOBAL RULES
- DBA_STREAMS_GLOBAL_RULES



- ALL_STREAMS_SCHEMA_RULES
- DBA_STREAMS_SCHEMA_RULES
- ALL_STREAMS_TABLE_RULES
- DBA_STREAMS_TABLE_RULES



When you drop a propagation, the propagation job used by the propagation is dropped automatically, if no other propagations are using the propagation job.

START_PROPAGATION Procedure

This procedure starts a propagation.

Syntax

```
DBMS_PROPAGATION_ADM.START_PROPAGATION(
    propagation name IN VARCHAR2);
```

Parameter

Table 156-5 START PROPAGATION Procedure Parameter

Parameter	Description
propagation_name	The name of the propagation you are starting. You must specify an existing propagation name. Do not specify an owner.

Usage Notes

The propagation status is persistently recorded. Hence, if the status is ENABLED, then the propagation is started upon database instance startup.

STOP_PROPAGATION Procedure

This procedure stops a propagation.

Syntax

Parameter

Table 156-6 STOP_PROPAGATION Procedure Parameter

Parameter	Description
propagation_name	The name of the propagation you are stopping. You must specify an existing propagation name. Do not specify an owner.



Table 156-6 (Cont.) STOP_PROPAGATION Procedure Parameter

Parameter	Description
force	If TRUE, then the procedure stops the propagation and clears the statistics for the propagation.
	If ${\tt FALSE},$ then the procedure stops the propagation without clearing the statistics for the propagation.

Usage Notes

The propagation status is persistently recorded. Hence, if the status is <code>DISABLED</code> or <code>ABORTED</code>, then the propagation is not started upon database instance startup.

