

Rule TYPEs


This chapter describes the types used with rules, rule sets, and evaluation contexts.

This chapter contains the following topics:

- [Overview](#)
- [Security Model](#)
- [Summary of Rule Types](#)

Rule TYPEs Overview


These types are used in rules and enable clients to evaluate rules with the rules engine.

 **See Also:**

- [DBMS_RULE](#)
- [DBMS_RULE_ADM](#)

Rule TYPEs Security Model

PUBLIC is granted EXECUTE privilege on these RULE types.

 **See Also:**

Oracle Database Security Guide for more information about user group PUBLIC

Summary of Rule Types

This table lists the Rule Types and briefly describes them.

Table 316-1 Rule Types

Type	Description
RE\$ATTRIBUTE_VALUE Type	Specifies the value of a variable attribute
RE\$ATTRIBUTE_VALUE_LIST Type	Identifies a list of attribute values
RE\$COLUMN_VALUE Type	Specifies the value of a table column
RE\$COLUMN_VALUE_LIST Type	Identifies a list of column values
RE\$NAME_ARRAY Type	Identifies a list of names

Table 316-1 (Cont.) Rule Types

Type	Description
RE\$NV_ARRAY Type	Identifies a list of name-value pairs
RE\$NV_LIST Type	Identifies an object containing a list of name-value pairs and methods that operate on this list. This object type is used to represent the event context and the action context for a rule
RE\$NV_NODE Type	Identifies a name-value pair
RE\$RULE_HIT Type	Specifies a rule found because of evaluation
RE\$RULE_HIT_LIST Type	Identifies a list of rules found because of evaluation
RE\$RULE_NAME Type	Allows users to pass a list of rule as argument.
RE\$RULE_NAME_LIST Type	Identifies a list of Rule Name values
RE\$TABLE_ALIAS Type	Provides the table corresponding to an alias used in a rule evaluation context
RE\$TABLE_ALIAS_LIST Type	Identifies a list of table aliases used in a rule evaluation context
RE\$TABLE_VALUE Type	Specifies the value of a table row using a <code>ROWID</code>
RE\$TABLE_VALUE_LIST Type	Identifies a list of table values
RE\$VALUE Type	Represents one datapoint value element
RE\$VALUE_LIST Type	Defines a list of <code>RE\$VALUE</code> type
RE\$VARIABLE_TYPE Type	Provides the type of a variable used in a rule evaluation context
RE\$VARIABLE_TYPE_LIST Type	Identifies a list of variables and their types used in a rule evaluation context
RE\$VARIABLE_VALUE Type	Specifies the value of a variable
RE\$VARIABLE_VALUE_LIST Type	Identifies a list of variable values

Rule types are used with the following Oracle-supplied PL/SQL packages:

- `DBMS_RULE`
- `DBMS_RULE_ADM`

You can use the `DBMS_RULE_ADM` package to create and administer rules, rule sets, and evaluation contexts, and you can use the `DBMS_RULE` package to evaluate rules.

When you use Replication (Oracle GoldenGate and XStream), rules determine which changes are captured by a capture process, which messages are propagated by a propagation, which messages are applied by an apply process, and which messages are dequeued by a messaging client. The following Replication (Oracle GoldenGate and XStream) packages use rules:

- `DBMS_APPLY_ADM`
- `DBMS_CAPTURE_ADM`
- `DBMS_PROPAGATION_ADM`

RE\$ATTRIBUTE_VALUE Type

RE\$ATTRIBUTE_VALUE type specifies the value of a variable attribute.



Note:

Enclose the variable name and attribute name in double quotation marks (") if the name contains special characters.

Syntax

```
TYPE SYS.RE$ATTRIBUTE_VALUE (  
    variable_name    VARCHAR2(32),  
    attribute_name   VARCHAR2(4000),  
    attribute_value  ANYDATA);
```

Attributes

Table 316-2 RE\$ATTRIBUTE_VALUE Attributes

Attribute	Description
variable_name	Specifies the variable used in a rule
attribute_name	Specifies the attribute name. The attribute name can be a multi-component name, such as a1.b2.c3.
attribute_value	Specifies the attribute value

RE\$ATTRIBUTE_VALUE_LIST Type

The RE\$ATTRIBUTE_VALUE_LIST type identifies a list of attribute values.

Syntax

```
TYPE SYS.RE$ATTRIBUTE_VALUE_LIST AS VARRAY(1024) OF SYS.RE$ATTRIBUTE_VALUE;
```

RE\$COLUMN_VALUE Type

RE\$COLUMN_VALUE type specifies the value of a table column.



Note:

Enclose the column name in double quotation marks (") if the name contains special characters.

Syntax

```
TYPE SYS.RE$COLUMN_VALUE (  
    table_alias    VARCHAR2(32),
```

```
column_name  VARCHAR2(4000),  
column_value ANYDATA);
```

Attributes

Table 316-3 RE\$COLUMN_VALUE Attributes

Attribute	Description
table_alias	Specifies the alias used for the table in a rule
column_name	Specifies the column name
column_value	Specifies the column value

RE\$COLUMN_VALUE_LIST Type

RE\$COLUMN_VALUE_LIST type identifies a list of column values.

Syntax

```
TYPE SYS.RE$COLUMN_VALUE_LIST AS VARRAY(1024) OF SYS.RE$COLUMN_VALUE;
```

RE\$NAME_ARRAY Type

RE\$NAME_ARRAY type identifies a list of names.

Syntax

```
TYPE SYS.RE$NAME_ARRAY AS VARRAY(1024) OF VARCHAR2(30);
```

RE\$NV_ARRAY Type

RE\$NV_ARRAY type identifies a list of name-value pairs.

Syntax

```
TYPE SYS.RE$NV_ARRAY AS VARRAY(1024) OF SYS.RE$NV_NODE;
```

RE\$NV_LIST Type

identifies an object containing a list of name-value pairs and methods that operate on this list.

This object type is used to represent the event context for rule set evaluation and the action context for a rule.

Syntax

```
TYPE SYS.RE$NV_LIST AS OBJECT(  
    actx_list SYS.RE$NV_ARRAY);
```

Attributes

Table 316-4 RE\$NV_LIST Attributes

Attribute	Description
actx_list	The list of name-value pairs

RE\$NV_LIST Subprograms

This section describes the following member procedures and member functions of the SYS.RE\$NV_LIST type:

- [ADD_PAIR Member Procedure](#)
- [GET_ALL_NAMES Member Function](#)
- [GET_VALUE Member Function](#)
- [REMOVE_PAIR Member Procedure](#)

ADD_PAIR Member Procedure

Adds a name-value pair to the list of name-value pairs.



Note:

Enclose the name in double quotation marks (") if the name contains special characters.

Syntax

```
MEMBER PROCEDURE ADD_PAIR(  
    name    IN VARCHAR2,  
    value   IN ANYDATA);
```

Parameters

Table 316-5 ADD_PAIR Procedure Parameters

Parameter	Description
name	The name in the name-value pair being added to the list. If the name exists in the list, then this procedure raises an error.
value	The value in the name-value pair being added to the list

GET_ALL_NAMES Member Function

Returns a list of all the names in the name-value pair list.

Syntax

```
MEMBER FUNCTION GET_ALL_NAMES()  
RETURN SYS.RE$NAME_ARRAY;
```

GET_VALUE Member Function

Returns the value for the specified name in a name-value pair list.



Note:

Enclose the name in double quotation marks (") if the name contains special characters.

Syntax

```
MEMBER FUNCTION GET_VALUE(  
    name IN VARCHAR2)  
RETURN ANYDATA;
```

Parameters

Table 316-6 GET_VALUE Function Parameters

Parameter	Description
name	The name whose value to return

REMOVE_PAIR Member Procedure

Removes the name-value pair with the specified name from the name-value pair list.



Note:

Enclose the name in double quotation marks (") if the name contains special characters.

Syntax

```
MEMBER PROCEDURE REMOVE_PAIR(  
    name IN VARCHAR2);
```

Parameters

Table 316-7 REMOVE_PAIR Procedure Parameters

Parameter	Description
name	The name of the pair to remove

RE\$NV_NODE Type

RE\$COLUMN_VALUE type identifies a name-value pair.



Note:

Enclose the name in double quotation marks (") if the name contains special characters.

Syntax

```
TYPE SYS.RE$NV_NODE (  
    nvn_name    VARCHAR2(30),  
    nvn_value   ANYDATA);
```

Attributes

Table 316-8 RE\$NV_NODE Attributes

Attribute	Description
nvn_name	Specifies the name in the name-value pair
nvn_value	Specifies the value in the name-value pair

RE\$RULE_HIT Type

The RE\$RULE_HIT type specifies a rule found because of an evaluation.



See Also:

- [CREATE_RULE Procedure](#)
- [ALTER_RULE Procedure](#)

Syntax

```
TYPE SYS.RE$RULE_HIT (  
    rule_name    VARCHAR2(65),  
    rule_action_context RE$NV_LIST);
```

Attributes

Table 316-9 RE\$RULE_HIT Attributes

Attribute	Description
rule_name	The rule name in the form <i>schema_name.rule_name</i> . For example, a rule named <i>employee_rule</i> in the <i>hr</i> schema is returned in the form "hr"."employee_rule".

Table 316-9 (Cont.) RE\$RULE_HIT Attributes

Attribute	Description
rule_action_context	The rule action context as specified in the CREATE_RULE or ALTER_RULE procedure of the DBMS_RULE_ADM package

RE\$RULE_HIT_LIST Type

The RE\$RULE_HIT_LIST type identifies a list of rules found because of an evaluation.

Syntax

```
TYPE SYS.RE$RULE_HIT_LIST AS VARRAY(1024) OF SYS.RE$RULE_HIT;
```

RE\$RULE_NAME Type

The RE\$RULE_HIT type is create to allow use to pass a list of rule as argument.

Syntax

```
CREATE OR REPLACE TYPE SYS.RE$RULE_NAME AS OBJECT(  
    rule_name          IN          varchar2,  
    evaluation_context IN          varchar2);
```

Attributes

Table 316-10 RE\$RULE_NAME Attributes

Attribute	Description
rule_name	The rule name in the form <i>SCHEMA_NAME.RULE_NAME</i> . For example, a rule named employee_rule in the hr schema is returned in the form "hr"."employee_rule".
Evaluation_context	The evaluation context as specified in the CREATE_RULE or ALTER_RULE procedure of the DBMS_RULE_ADM package

RE\$RULE_NAME_LIST Type

RE\$RULE_HIT_LIST type Identifies a list of Rule Name values.

Syntax

```
CREATE OR REPLACE TYPE SYS.RE$RULE_NAME_LIST AS VARRAY(1024) OF SYS.RE$RULE_NAME;
```

RE\$TABLE_ALIAS Type

RE\$TABLE_ALIAS type provides the table corresponding to an alias used in a rule evaluation context.

A specified table name must satisfy the schema object naming rules.



Note:

Enclose the table name in double quotation marks (") if the name contains special characters.



See Also:

Oracle Database SQL Language Reference for information about schema object naming rules

Syntax

```
TYPE SYS.RE$TABLE_ALIAS IS OBJECT(  
    table_alias  VARCHAR2(32),  
    table_name   VARCHAR2(194));
```

Attributes

Table 316-11 RE\$TABLE_ALIAS Attributes

Attribute	Description
table_alias	The alias used for the table in a rule
table_name	<p>The table name referred to by the alias. A synonym can be specified. The table name is resolved in the evaluation context schema.</p> <p>The format is one of the following:</p> <p><i>schema_name.table_name</i></p> <p><i>table_name</i></p> <p>For example, if the <i>schema_name</i> is <i>hr</i> and the <i>table_name</i> is <i>employees</i>, then enter the following:</p> <p><i>hr.employees</i></p>

RE\$TABLE_ALIAS_LIST Type

The `RE$TABLE_ALIAS_LIST` type identifies a list of table aliases used in a rule evaluation context.

Syntax

```
TYPE SYS.RE$TABLE_ALIAS_LIST AS VARRAY(1024) OF SYS.RE$TABLE_ALIAS;
```

RE\$TABLE_VALUE Type

The `RE$TABLE_VALUE` type specifies the value of a table row using a `ROWID`.

Syntax

```
TYPE SYS.RE$TABLE_VALUE (  
    table_alias  VARCHAR2(32),  
    table_rowid  VARCHAR2(18));
```

Attributes

Table 316-12 RE\$TABLE_VALUE Attributes

Attribute	Description
table_alias	Specifies the alias used for the table in a rule
table_rowid	Specifies the rowid for the table row

RE\$TABLE_VALUE_LIST Type

The `RE$TABLE_VALUE_LIST` type identifies a list of table values.



Note:

Each table alias in the list in the list must be unique.

Syntax

```
TYPE SYS.RE$TABLE_VALUE_LIST AS VARRAY(1024) OF SYS.RE$TABLE_VALUE;
```

RE\$VALUE Type

The `RE$VALUE` type represents one datapoint value element.

Syntax

```
TYPE SYS.RE$VALUE (  
    table_values      IN      SYS.RE$TABLE_VALUE_LIST:= NULL,  
    column_values     IN      SYS.RE$COLUMN_VALUE_LIST:=NULL,  
    variable_values   IN      SYS.RE$VARIABLE_VALUE_LIST:=NULL,  
    attribute_values  IN      SYS.RE$ATTRIBUTE_VALUE_LIST:=NULL)
```

Attributes

Table 316-13 RE\$VALUE Attributes

Attribute	Description
table_values	ROWID of table row for expression evaluation
column_values	Values of columns referred in the expression

Table 316-13 (Cont.) RE\$VALUE Attributes

Attribute	Description
variable_values	Values of variables referred in expressio
attribute_values	Values of attributes referred in expression

RE\$VALUE_LIST Type

RE\$TABLE_VALUE_LIST type is a varray for defining a list of RE\$VALUE type. It represents a list of datapoint element.

Syntax

```
CREATE OR REPLACE TYPE SYS.RE$VALUE _LIST AS VARRAY(1024) OF SYS.RE$VALUE;
```

RE\$VARIABLE_TYPE Type

The RE\$VARIABLE_TYPE type provides the type of a variable used in a rule evaluation context. A specified variable name must satisfy the schema object naming rules.



Note:

Enclose the variable name in double quotation marks (") if the name contains special characters.



See Also:

Oracle Database SQL Language Reference for information about schema object naming rules

Syntax

```
TYPE SYS.RE$VARIABLE_TYPE (  
    variable_name          VARCHAR2(32),  
    variable_type          VARCHAR2(4000),  
    variable_value_function VARCHAR2(228),  
    variable_method_function VARCHAR2(228));
```

Attributes

Table 316-14 RE\$VARIABLE_TYPE Attributes

Attribute	Description
variable_name	The variable name used in a rule

Table 316-14 (Cont.) RE\$VARIABLE_TYPE Attributes

Attribute	Description
variable_type	The type that is resolved in the evaluation context schema. Any valid Oracle built-in datatype, user-defined type, or Oracle-supplied type can be specified. See the <i>Oracle Database SQL Language Reference</i> for more information about these types.
variable_value_function	A value function that can be specified for implicit variables. A synonym can be specified. The function name is resolved in the evaluation context schema. It is executed on behalf of the owner of a rule set using the evaluation context or containing a rule that uses the evaluation context. See the " Usage Notes " for more information.
variable_method_function	Specifies a value function, which can return the result of a method invocation. Specifying such a function can speed up evaluation, if there are many simple rules that invoke the method on the variable. The function can be a synonym or a remote function. The function name is resolved in the evaluation context schema. It is executed on behalf of the owner of a rule set using the evaluation context or containing a rule that uses the evaluation context. See the " Usage Notes " for more information.

Usage Notes

The functions for both the `variable_value_function` parameter and `variable_method_function` parameter have the following format:

```
schema_name.package_name.function_name@dblink
```

Any of the following parts of the format can be omitted: `schema_name`, `package_name`, and `@dblink`.

For example, if the `schema_name` is `hr`, the `package_name` is `var_pac`, the `function_name` is `func_value`, and the `dblink` is `dbs1.net`, then enter the following:

```
hr.var_pac.func_value@dbs1.net
```

The following sections describe the signature of the functions.

Signature for variable_value_function

The function must have the following signature:

```
FUNCTION variable_value_function_name(
    evaluation_context_schema IN VARCHAR2,
    evaluation_context_name   IN VARCHAR2,
    variable_name              IN VARCHAR2,
    event_context              IN SYS.RE$NV_LIST )
RETURN SYS.RE$VARIABLE_VALUE;
```

Signature for variable_method_function

This function must have the following signature:

```
FUNCTION variable_method_function_name(
    evaluation_context_schema IN VARCHAR2,
```

```
evaluation_context_name    IN VARCHAR2,  
variable_value             IN SYS.RE$VARIABLE_VALUE,  
method_name               IN VARCHAR2,  
event_context             IN SYS.RE$NV_LIST)  
RETURN SYS.RE$ATTRIBUTE_VALUE;
```

RE\$VARIABLE_TYPE_LIST Type

The `RE$VARIABLE_TYPE_LIST` type identifies a list of variables and their types used in a rule evaluation context.

Syntax

```
TYPE SYS.RE$VARIABLE_TYPE_LIST AS VARRAY(1024) OF SYS.RE$VARIABLE_TYPE;
```

RE\$VARIABLE_VALUE Type

The `RE$VARIABLE_VALUE` type specifies the value of a variable.



Note:

Enclose the variable name in double quotation marks (") if the name contains special characters.

Syntax

```
TYPE SYS.RE$VARIABLE_VALUE (  
    variable_name  VARCHAR2(32),  
    variable_data  ANYDATA);
```

Attributes

Table 316-15 RE\$VARIABLE_VALUE Attributes

Attribute	Description
variable_name	Specifies the variable name used in a rule
variable_data	Specifies the data for the variable value

RE\$VARIABLE_VALUE_LIST Type

The `RE$VARIABLE_VALUE_LIST` type identifies a list of variable values.

Syntax

```
TYPE SYS.RE$VARIABLE_VALUE_LIST AS VARRAY(1024) OF SYS.RE$VARIABLE_VALUE;
```