



Trusted RUBIX™

Version 6

Information Schema Guide

Revision 6

RELATIONAL DATABASE MANAGEMENT SYSTEM

Infosystems Technology, Inc.

4 Professional Dr - Suite 118

Gaithersburg, MD 20879

TEL +1-202-412-0152

© 1981, 2014 Infosystems Technology, Inc. (ITI). All rights reserved. Unpublished work. Commercial computer software and software documentation: Government users are subject to ITI's standard license agreement per DFARS 227.7203-3 or, in non-DoD agencies where such protection is unavailable, to "restricted rights" under applicable FAR System clauses.

Infosystems Technology, Inc.
4 Professional Dr - Suite 118
Gaithersburg, MD 20879

THIS DOCUMENTATION CONTAINS CONFIDENTIAL INFORMATION AND TRADE SECRETS OF INFOSYSTEMS TECHNOLOGY, INC. USE, DISCLOSURE, OR REPRODUCTION IS PROHIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF INFOSYSTEMS TECHNOLOGY, INC. FOR FULL DETAILS OF THE TERMS AND CONDITIONS FOR USING THE SOFTWARE, PLEASE REFER TO THE ITI-TRUSTED RUBIX USER LICENSE AGREEMENT.

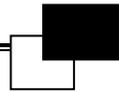
The information in this document is subject to change without notice and should not be construed as a commitment by ITI.

Infosystems Technology, Inc. assumes no responsibility for any errors that may appear in this document.

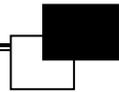
RUBIX® is a trademark of Infosystems Technology, Inc.

UNIX® is a trademark of The Open Group.

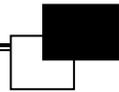
Printed in U.S.A.



INTRODUCTION	1
CATALOG_PRIVILEGES VIEW	2
↓ FUNCTION	2
↓ DESCRIPTION	2
↓ SECURITY	3
CATALOGS VIEW	3
↓ FUNCTION	3
↓ DESCRIPTION	3
↓ SECURITY	4
CHECK_CONSTRAINTS VIEW	4
↓ FUNCTION	4
↓ DESCRIPTION	4
↓ SECURITY	4
COLUMN_KEYS VIEW	5
↓ FUNCTION	5
↓ DESCRIPTION	5
↓ SECURITY	5
COLUMN_PRIVILEGES VIEW	6
↓ FUNCTION	6
↓ DESCRIPTION	6
↓ SECURITY	7
COLUMNS VIEW	7
↓ FUNCTION	7
↓ DESCRIPTION	7
↓ SECURITY	9
DATABASE_PRIVILEGES VIEW	9
↓ FUNCTION	9
↓ DESCRIPTION	9
↓ SECURITY	10
DEFAULT_CONSTRAINTS VIEW	10
↓ FUNCTION	10
↓ DESCRIPTION	10
↓ SECURITY	11
INDEXES VIEW	11
↓ FUNCTION	11
↓ DESCRIPTION	11
↓ SECURITY	12
NOTNULL_CONSTRAINTS VIEW	12
↓ FUNCTION	12
↓ DESCRIPTION	12
↓ SECURITY	13



REFERENTIAL_CONSTRAINTS VIEW	13
↓ FUNCTION	13
↓ DESCRIPTION	13
↓ SECURITY	14
RX_COLUMN_PRIVILEGES VIEW	14
↓ FUNCTION	14
↓ DESCRIPTION	14
↓ SECURITY	15
RX_TABLE_PRIVILEGES VIEW	15
↓ FUNCTION	15
↓ DESCRIPTION	15
↓ SECURITY	16
SCHEMATA VIEW	17
↓ FUNCTION	17
↓ DESCRIPTION	17
↓ SECURITY	17
SCHEMA_PRIVILEGES VIEW	17
↓ FUNCTION	17
↓ DESCRIPTION	18
↓ SECURITY	18
SERVER_INFO VIEW	18
↓ FUNCTION	18
↓ DESCRIPTION	18
↓ SECURITY	19
SQL_LANGUAGES VIEW	20
↓ FUNCTION	20
↓ DESCRIPTION	20
↓ SECURITY	20
TABLE_PRIVILEGES VIEW	21
↓ FUNCTION	21
↓ DESCRIPTION	21
↓ SECURITY	21
TABLES VIEW	22
↓ FUNCTION	22
↓ DESCRIPTION	22
↓ SECURITY	22
UNIQUE_CONSTRAINTS VIEW	23
↓ FUNCTION	23
↓ DESCRIPTION	23
↓ SECURITY	23
USAGE_PRIVILEGES VIEW	23



↓ FUNCTION	23
↓ DESCRIPTION	24
↓ SECURITY	24
VIEWS VIEW.....	25
↓ FUNCTION	25
↓ DESCRIPTION	25
↓ SECURITY	25
APPLICATIONS VIEW	26
↓ FUNCTION	26
↓ DESCRIPTION	26
↓ SECURITY	26
APPLICATION_ADMINS VIEW.....	26
↓ FUNCTION	26
↓ DESCRIPTION	26
↓ SECURITY	27
APPLICATION_USERS VIEW	27
↓ FUNCTION	27
↓ DESCRIPTION	27
↓ SECURITY	27

Introduction

System views are read-only views from which users can retrieve information about any objects to which they have access. This information also describes the system views themselves. **TRUSTED RUBIX** adheres to the Open Group 1992 standard.

NOTE

The OPEN GROUP SQL Specification standard concerning the information schema uses a naming convention for view, table, and column names that is case insensitive. **TR** uses a naming convention that is case sensitive. The following documentation on the information schema uses the convention of uppercase letters for describing Open Group specification defined named objects. This is consistent with the convention used throughout the rest of the **TR** documentation. However, the catalog, schema, table, view, and column names should be entered in lowercase when accessing objects in the information schema.

All system views are part of the schema `INFO_SCHEM` which resides in the `SYSTEM_CATALOG`. By default the `SELECT` privilege is set to `PUBLIC` on each information schema view. By default `READ` and `EXEC` privilege is set to `PUBLIC` on `INFO_SCHEM`. By default `READ`, `EXEC` is set to `PUBLIC` for the `SYSTEM_CATALOG`. Since `INFO_SCHEM` is read-only, users can only issue queries, i.e., `SELECT` statements, against the views. However, in some cases, certain users are not entitled to access the complete system view. These cases are described for each system view in the `SECURITY` section.

Each of the views in the `INFO_SCHEM` is built upon tables contained in the `DEFINITION_SCHEMA` schema, which itself is contained in the `SYSTEM_CATALOG`. All schemas in the `SYSTEM_CATALOG` have fixed DAC privileges set to `PUBLIC`, `EXEC` and `READ`. Each table in the `SYSTEM_CATALOG` has fixed DAC privileges set to `PUBLIC CRVIEW` and `REFVIEW`. No user may change any of these DAC privileges.

Normally a view is based on another (underlying) table or view. However, Open Group does not define the basis of system views; indeed, Open Group expects the underlying data structures to vary among implementations. The information schema `INFO_SCHEM` may contain additional system views beyond those which are defined in the Open Group standard. An application obtains information on the existence and structure of any additional system views by examining the `COLUMNS`, `TABLES` and `VIEWS` system views. The maximum length of user-defined names in Open Group SQL is 18 characters.

Each row in a privilege table corresponds to a RUBIX DAC privilege of a particular user or group of users. The columns in a privilege table differ depending on the object type, but all privilege tables contain a `GRANTOR`, `GRANTEE`, `PRIVILEGE_TYPE`, and `IS_GRANTABLE` field.

The `GRANTOR` is the entity doing the granting. `GRANTOR` is always the user name of the user who granted the privilege in **TR**. The `GRANTEE` indicates to whom the privilege is being granted and typically contains a *user-name*. It can also contain "all" indicating all users.

TR provides an additional column in privilege tables called "*grantee_group*". This indicates the group to whom the privilege is granted or "all". As a result, in **TR**, there are three valid combinations of the "*grantee*" and "*grantee_group*" fields. The first form is (*user-name*, all) which means that user is has been granted the corresponding DAC privilege regardless of their *group-name*. The second form is (all, *group-name*) which means that all members of the group are granted the corresponding DAC privilege regardless of their *user-name*. The third form is (all, all) which corresponds to "public" access.

The `PRIVILEGE_TYPE` is the privilege that is granted and `IS_GRANTABLE` is "yes" if the user can grant the

privilege.

TRUSTED RUBIX supports the following system views:

CATALOG_PRIVILEGES view	RX_COLUMN_PRIVILEGES view
CATALOG view	RX_TABLE_PRIVILEGES view
CHECK_CONSTRAINTS view	SCHEMATA view
COLUMN_KEYS view	SCHEMA_PRIVILEGES view
COLUMN_PRIVILEGES view	SERVER_INFO view
COLUMNS view	SQL_LANGUAGES view
DATABASE_PRIVILEGES view	TABLE_PRIVILEGES view
DEFAULT_CONSTRAINTS view	TABLES view
INDEXES view	UNIQUE_CONSTRAINTS view
NOTNULL_CONSTRAINTS view	USAGE_PRIVILEGES view
REFERENTIAL_CONSTRAINTS view	VIEWS view
APPLICATIONS view	APPLICATION_USERS view
APPLICATION_ADMINS view	

NOTE

In addition to the Multilevel Security (MLS) Mandatory Access Control (MAC) policy, **TR** supports the Type Enforcement (TE) MAC policy of SELinux and a proprietary Attribute Based Access Control (ABAC) MAC policy of the Security Policy Manager (SPM). In general, all configured MAC policies must permit an operation for it to succeed. For more information on TE and SELinux please see the **Trusted RUBIX SELinux Guide** and for more information on ABAC and the SPM please see the **Trusted RUBIX Security Policy Manager Reference Guide and Tutorial**.

CATALOG_PRIVILEGES view

↓ FUNCTION

The CATALOG_PRIVILEGES system view contains exactly one row for each catalog. Its purpose is to identify all privileges on all catalogs.

↓ DESCRIPTION

Each row has the following columns:

GRANTOR: The entity that granted the privilege. GRANTOR is always the user name of the user who granted the privilege in **TR**.

GRANTEE: The user to whom the granted privilege is granted or “ALL”.

GRANTEE_GROUP: The group to whom the privilege is granted or “ALL”.

TABLE_CAT: The catalog on which the privilege is granted.

PRIVILEGE_TYPE: The privilege type granted (e.g., SELECT).

IS_GRANTABLE: This column indicates whether the grantee or grantee group can grant this privilege to other users/groups. The possible values for this column are “yes” or “no.”

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table those rows do not appear in the result of a SELECT upon the view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC, READ on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

CATALOGS View

↓ FUNCTION

The CATALOGS VIEW system view contains a row for each catalog in the database.

↓ DESCRIPTION

Each row contains these columns:

CAT_NAME: The name of the catalog.

CAT_OWNER: As **TRUSTED RUBIX** has no object owner concept this column is always set to RUBIX.

DEF_CHAR_SET_CAT: Unused

DEF_CHAR_SET_NAME: Unused

REMARKS: May contain descriptive information about the catalog.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the catalog it refers to. Thus, if you do not have MAC dominance over a catalog those rows do not appear in the result of a SELECT upon the view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC, READ on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

CHECK_CONSTRAINTS view

↓ FUNCTION

The CHECK_CONSTRAINTS system view contains a row for each constraint specifying a column (or columns) value check that has been applied to a table.

↓ DESCRIPTION

Each row contains these columns:

CHECK_CATALOG: The catalog holding the target table.

CHECK_SCHEMA: The schema holding the target table.

CHECK_TABLE: The target table.

CONSTRAINT_NAME: The name of the constraint.

CHECK_CLAUSE: The parse tree corresponding to the check operation

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)

- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

COLUMN_KEYS View

↓ FUNCTION

The COLUMN_KEYS system view contains a row for each column accessible to the current user.

↓ DESCRIPTION

Each row has the following columns:

TABLE_CAT: The name of the catalog that is the container for the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column.

TABLE_NAME: The name of the table or view.

COLUMN_NAME: The name of the column of the specified table or view.

IS_KEY: This column may contain one of two values—"yes" or "no". If "yes", then this column is a key for the specified column name.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

COLUMN_PRIVILEGES View

↓ FUNCTION

The COLUMN_PRIVILEGES system view describes each case where a column privilege was granted.

↓ DESCRIPTION

The view contains exactly one row for each column of each table to which the current user has access (see SECURITY below). Within each row, there are the following columns:

GRANTOR: The user name that granted the privilege in question. GRANTOR is always the user name of the user who granted the privilege.

GRANTEE: The user name of the user to whom the privilege in question was granted or “all”. Granting a privilege to “public” results in only one row in the COLUMN_PRIVILEGES view (per privilege granted) and the GRANTEE column contains the value “all”.

TABLE_CAT: The name of the catalog that that is the container of the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column.

TABLE_NAME: The name of the table in question.

COLUMN_NAME: The name of the column in question.

PRIVILEGE_TYPE: The type of column privilege that was granted. Its value is “insert”, “references”, “select” or “update”.

IS_GRANTABLE: This column contains the value “yes” if the person invoking the SELECT on COLUMN_PRIVILEGES has the ability to further grant this privilege.

REMARKS: May contain descriptive information about the table. It is always NULL.

COLUMN_PRIVILEGES includes one row for each privilege granted to each user or to “public” on each column (whether or not the specified privilege can be granted or revoked on individual columns).

The DELETE privilege is not applied on columns.

This privilege is found in the TABLE_PRIVILEGES table.

An exception is that COLUMN_PRIVILEGES does not contain information on the DELETE privilege, which in no case applies to individual columns.

Each user only sees those rows in which they have the READ privilege on the schema in which the table is contained.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

COLUMNS View

↓ FUNCTION

The COLUMNS system view contains a row for each column accessible to the current user.

↓ DESCRIPTION

Each row contains these columns:

TABLE_CAT: The name of the catalog that is the container for the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column.

TABLE_NAME: The name of the table or view.

COLUMN_NAME: The name of the column of the specified table or view.

ORDINAL_POSITION: The ordinal position of the column in the table. The first column in the table is number 1.

COLUMN_DEF: The column's default value, using legal syntax for *default-value* in the *column-definition* of the CREATE TABLE or ALTER TABLE statement. If the default value is a character string, then this column is that string enclosed in single quotes. If the default value is a numeric literal,

then this column contains the original character representation with no enclosing single quotes. If the default value is a date-time literal, then this column contains the appropriate keyword (DATE, TIME or TIMESTAMP) followed by the *date-value* and/or *time-value* enclosed in single quotes. If the default is a *pseudo-literal*, then this column contains the keyword, such as CURRENT_DATE, with no enclosing single quotes. If NULL was specified as the default value, then this column is the word NULL, not enclosed in quotes. If the default value cannot be represented without truncation, then this column contains TRUNCATED with no enclosing single quotes. If no default value was specified, then this column is null. The value of COLUMN_DEF is suitable for use in generating a new *column-definition*, except when it contains the value TRUNCATED.

IS_NULLABLE: Contains the value “no” if the column is known to be not nullable, according to the rules in the International Standard; and “yes” otherwise.

DATA_TYPE: Identifies the type of the column and can contain one of the following values: CHARACTER, CHARACTER VARYING, DATETIME, DECIMAL, DOUBLE PRECISION, FLOAT, INTEGER, INTERVAL, NUMERIC, REAL, SMALLINT, TIMESTAMP.

CHAR_MAX_LENGTH: Contains the maximum length in characters for a character data type column. For all other data types it is null.

CHAR_OCTET_LENGTH: Contains the maximum length in octets for a character data type column. For all other data types it is null. (For single-octet character sets, this is the same as CHAR_MAX_LENGTH.)

NUM_PREC: If DATA_TYPE is an approximate numeric data type, this column contains the number of bits of mantissa precision of the column. For exact numeric data types, this column contains the total number of decimal digits allowed in the column. For the date/time subtypes TIME and TIMESTAMP and for INTERVAL types that have a SECOND field, this column contains the number of digits of precision of the fractional seconds component. Otherwise, this column is null. NUM_PREC_RADIX (see below) indicates the units of measurement.

NUM_PREC_RADIX: If DATA_TYPE is an approximate numeric data type, this column contains the value 2 because NUM_PREC specifies a number of bits. For exact numeric data types, this column contains the value 10 because NUM_PREC specifies a number of decimal digits. Otherwise, this column is null. By combining the precision with the radix, an application can calculate the maximum number that the column can hold.

NUM_SCALE: Defines the total number of significant digits to the right of the decimal point. For the INTEGER and SMALLINT data types, it is 0. For the CHARACTER, CHARACTER VARYING, DATETIME, FLOAT, INTERVAL, REAL and DOUBLE PRECISION data types, it is null.

DATETIME_CODE: For date/time and interval data types, this column is the subtype code, using the same values as the DATETIME_INTERVAL_CODE field of the SQL descriptor of dynamic SQL. For other data types, this column is null.

INTERVAL_CODE: For interval data types, this column contains a character string with text of an *interval-qualifier* specifying the interval precision (for example, 'HOUR TO MINUTE'). For other data types, its value is null.

INTERVAL_PREC: This column contains the number of significant digits for the leading precision of interval data types. For all other data types, this column is null.

CHAR_SET_CAT: [To be defined as per Open Group specification.]

CHAR_SET_SCHEM: [To be defined as per Open Group specification.]

CHAR_SET_NAME: [To be defined as per Open Group specification.]

COLLATION_CAT: [To be defined as per Open Group specification.]

COLLATION_SCHEM: [To be defined as per Open Group specification.]

COLLATION_NAME: [To be defined as per Open Group specification.]

REMARKS: May contain descriptive information about the column.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

DATABASE_PRIVILEGES View

↓ FUNCTION

The DATABASE_PRIVILEGES system view contains exactly one row for each privilege granted on the database.

↓ DESCRIPTION

The DATABASE_PRIVILEGES system view has the following columns:

GRANTOR: The entity that granted the privilege. GRANTOR is always the user name of the user who granted the privilege in **TRUSTED RUBIX**.

GRANTEE: The user to whom the privilege is granted or "all".

GRANTEE_GROUP: The group to whom the privilege is granted or "all".

PRIVILEGE_TYPE: The privilege type (e.g., READ).

IS_GRANTABLE: This column contains the value "yes" if the person invoking the READ on DATABASE_PRIVILEGES has the ability to further grant this privilege

↓ SECURITY

MAC The current user's sensitivity label must dominate the sensitivity label of the database.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC, READ on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

DEFAULT_CONSTRAINTS view

↓ FUNCTION

The DEFAULT_CONSTRAINTS system view contains a row for each constraint specifying a default column value that has been applied to a table.

↓ DESCRIPTION

Each row contains these columns:

SYSTEM_CATALOG: The catalog holding the target table.

DEFAULT_SCHEMA: The schema holding the target table.

DEFAULT_TABLE: The target table holding the target column.

DEFAULT_COLUMN: The target column to have a default value.

DEFAULT_VALUE: The parse tree corresponding to the default value.

CONSTRAINT_NAME: The name of the constraint

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

INDEXES View

↓ FUNCTION

The INDEXES system view describes each index accessible to the current user.

↓ DESCRIPTION

The INDEXES system view contains one row for each index column. The row contains at least these columns:

TABLE_CAT: The name of the catalog that is the container for the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column. .

TABLE_NAME: The name of the base table.

COLUMN_NAME: The name of the column of the specified base table and index.

INDEX_NAME: The unique name of the index.

ORDINAL_POSITION: The ordinal number of the column in the index. The ordering is determined by the order of the columns in the CREATE INDEX statement. The numbering of the columns of the index starts from 1 and increases contiguously.

NON_UNIQUE: Contains the value “no” if at most one row is allowed in TABLE_NAME for each combination of values in the specified columns for this index; and “yes” otherwise.

ASC_OR_DESC: Contains the value 'A' if the order of the referenced column is ascending and 'D' if the order of the referenced column is descending.

REMARKS: May contain descriptive information about the index.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

NOTNULL_CONSTRAINTS view

↓ FUNCTION

The NOTNULL_CONSTRAINTS system view contains a row for each *not-null* constraint specified for a column that has been applied to a table.

↓ DESCRIPTION

Each row contains these columns:

NOTNULL_CATALOG: The catalog holding the target table.

NOTNULL_SCHEMA: The schema holding the target table.

NOTNULL_TABLE: The target table holding the target column.

NOTNULL_COLUMN: The target column to be *not-null*.

CONSTRAINT_NAME: the name of the constraint.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

REFERENTIAL_CONSTRAINTS view

↓ FUNCTION

The REFERENTIAL_CONSTRAINTS system view contains a row for each referential integrity (foreign key) constraint on a column or columns of a table.

↓ DESCRIPTION

Each row contains these columns:

REFER_CATALOG: The catalog holding the target column.

REFER_SCHEMA: The schema holding the target column.

REFER_TABLE: The target column.

REFER_INDEX: The index on the target columns.

REFER_OPERATION: The operation to perform an action upon (update or delete).

REFER_PARENT_CATALOG: The catalog holding the target column.

REFER_PARENT_SCHEMA: The schema holding the target column.

REFER_PARENT_TABLE: The target column.

REFER_PARENT_COLBITS: The bitmap of the parent columns.

REFER_ACTION: The action to take on update or delete (cascade, set default, set null, no action).

REFER_FLAGS: Flags used internally for the referential integrity.

REFER_COLBITS: The bitmap of the target columns.

CONSTRAINT_NAME: The name of the constraint.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEMA schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEMA column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

RX_COLUMN_PRIVILEGES View

↓ FUNCTION

The RX_COLUMN_PRIVILEGES system view describes the complete **TRUSTED RUBIX** privileges for any given column.

↓ DESCRIPTION

The RX_COLUMN_PRIVILEGES system view is an exact copy of the COLUMN_PRIVILEGES system view except that a higher level of granularity of protection is provided by the addition of another column, namely “*grantee_group*”. This set of system views is required to match the Open Group specification for the COLUMN_PRIVILEGES system view and to increase the level of security for **TRUSTED RUBIX**. Within each row, there are the following columns:

GRANTOR: The user name that granted the privilege in question. GRANTOR is always the user name of the user who granted the privilege.

GRANTEE: The user name of the user to whom the privilege in question was granted or “all”. Granting a privilege to “public” results in only one row in the RX_COLUMN_PRIVILEGES view (per privilege granted) and the grantee column contains the value “all”.

GRANTEE_GROUP: The group to whom the privilege is granted or “all”.

TABLE_CAT: The name of the catalog that contains the table in question.

TABLE_SCHEM: The name of the schema that contains the table in question.

TABLE_NAME: The name of the table in question

COLUMN_NAME: The name of the column in question.

PRIVILEGE_TYPE: The type of column privilege that was granted. Its value is “insert”, “references”, “select” or “update”.

IS_GRANTABLE: This column contains the value “yes” if the person invoking the SELECT on COLUMN_PRIVILEGES has the ability to further grant this privilege.

REMARKS: May contain descriptive information about the table. It is always NULL.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

RX_TABLE_PRIVILEGES View

↓ FUNCTION

The RX_TABLE_PRIVILEGES system view describes the complete **TRUSTED RUBIX** privileges on any given table or view.

↓ DESCRIPTION

The RX_TABLE_PRIVILEGES system view is an exact copy of the TABLE_PRIVILEGES system view except that a higher level of granularity of protection is provided for by the addition of another column, namely “*grantee_group*”. This set of system views is required to follow the Open Group specification for the

TABLE_PRIVILEGES system view and to increase the level of security for **TR**.

GRANTOR: The user name of the user that granted the privilege in question. GRANTOR is always the user name of the user who granted the privilege.

GRANTEE: The user name of the user to whom the privilege in question was granted or “all”. Granting a privilege to “public” results in only one row in the RX_TABLE_PRIVILEGES view (per privilege granted) and the “grantee” column contains the value “all”.

GRANTEE_GROUP: The group to whom the privilege is granted or “all”.

TABLE_CAT: The name of the catalog that contains the table in question.

TABLE_SCHEM: The name of the schema that contains the table in question.

TABLE_NAME: The name of the table in question.

PRIVILEGE_TYPE: The type of privilege that was granted. Its value must be one of the following: DELETE, INSERT, REFERENCES, SELECT, UPDATE.

IS_GRANTABLE: This column contains the value “yes” if the person invoking the SELECT on COLUMN_PRIVILEGES has the ability to further grant this privilege.

REMARKS: May contain descriptive information about the table.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

SCHEMATA View

↓ FUNCTION

The SCHEMATA system view contains one row describing each schema of the database.

↓ DESCRIPTION

Within each row of the SCHEMATA system view there are the following columns:

CAT_NAME: The name of the catalog that contains the schema in question.

SCHEM_NAME: The name of the schema in question.

SCHEM_OWNER: As **TRUSTED RUBIX** has no object owner concept this column is always set to RUBIX).

DEF_CHAR_SET_CAT: [To be defined as per Open Group specification.]

DEF_CHAR_SET_NAME: [To be defined as per Open Group specification.]

REMARKS: May contain descriptive information about the table.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the schema it refers to. Thus, if you do not have MAC dominance over a schema, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of this view (**RA003**)
- READ on the catalog listed in the CAT_NAME column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

SCHEMA_PRIVILEGES View

↓ FUNCTION

The SCHEMA_PRIVILEGES system view contains exactly one row for each schema.

↓ DESCRIPTION

The `SCHEMA_PRIVILEGES` system view has the following columns:

GRANTOR: The entity that granted the privilege. **GRANTOR** is always the user name of the user who granted the privilege in **TRUSTED RUBIX**.

GRANTEE: The user to whom the privilege is granted or “all”.

GRANTEE_GROUP: The group to whom the privilege is granted or “all”.

TABLE_SCHEM: The schema on which the privilege is granted.

PRIVILEGE_TYPE: The privilege type (e.g., `SELECT`).

IS_GRANTABLE: This column indicates whether the grantee or *grantee_group* can grant this privilege to other users/groups. The possible values for this column are “yes” or “no”.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the schema it refers to. Thus, if you do not have MAC dominance over a schema, those rows do not appear in a `SELECT` of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on `SYSTEM_CATALOG` (**RA008**)
- EXEC and READ on `INFO_SCHEM` schema (**RA002**)
- SELECT on referenced columns of this view (**RA003**)
- READ on the catalog listed in the `TABLE_CAT` column of this view (else the row is not visible).

AUDIT See the `SELECT` command in the **SQL Reference Guide**.

SERVER_INFO View

↓ FUNCTION

The `SERVER_INFO` system view describes the server the application is currently connected to.

↓ DESCRIPTION

Each row of the `SERVER_INFO` system view provides information about one attribute. The `SERVER_ATTRIBUTE` column identifies an attribute of servers; the `ATTRIBUTE_VALUE` specifies the value of that attribute as it applies to the current server.

All Open Group-compliant database systems must provide applications with certain minimum information about any server. That is, for all the values of `SERVER_ATTRIBUTE` listed below, the `SERVER_INFO` system view must contain one row with that value, describing that attribute of the current server.

Implementations may define additional attributes and their corresponding values. For these attributes, the name of the `SERVER_ATTRIBUTE`, the range of values in `ATTRIBUTE_VALUE`, and the resulting interpretation are implementation-defined.

The attributes that `SERVER_INFO` specifies are:

`CATALOG_NAME`: This attribute is “yes” if the server supports catalog names, and “no otherwise.

`COLLATION_SEQ`: This attribute is the assumed ordering of the character set for this server.

`IDENTIFIER_LENGTH`: This attribute is the maximum number of characters for a user-defined name. The value is the character string representation of the decimal value.

`INTERVAL_FRACTIONAL_PRECISION`: This attribute is the default fractional precision for all objects of data type `INTERVAL SECOND`, `INTERVAL MINUTE TO SECOND`, `INTERVAL HOUR TO SECOND`, or `INTERVAL DAY TO SECOND`.

`INTERVAL_LEADING_PRECISION`: This attribute is the default leading precision for all intervals.

`ROW_LENGTH`: This attribute is the maximum size of a row. The value is the character string representation of the decimal value.

`TIME_PRECISION`: This attribute is the default fractional precision for all objects whose data type is date/time and whose subtype is `TIME`.

`TIMESTAMP_PRECISION`: This attribute is the default fractional precision for all objects whose data type is date/time and whose subtype is `TIMESTAMP`.

`TXN_ISOLATION`: This attribute is the initial transaction isolation level the server assumes. The value must be one of the following, corresponding to isolation levels defined in the International Standard: `READ UNCOMMITTED`, `READ COMMITTED`, `REPEATABLE READ`, `SERIALIZABLE`.

`USERID_LENGTH`: This attribute is the maximum number of characters of a user name (or “authorization identifier”). The value is the character string representation of the decimal value.

↓ SECURITY

MAC The user must have MAC dominance over the database.

DAC A user can see a particular row in this view if the user has the `rubix.dac.select.dbname` authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on `SYSTEM_CATALOG` (**RA008**)
- EXEC on `INFO_SCHEM` schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

SQL_LANGUAGES View

↓ FUNCTION

List the SQL standards and SQL dialectics that are supported.

↓ DESCRIPTION

The SQL_LANGUAGES system view contains a row for every conformance claim the SQL product makes (including subsets defined for ISO and vendor-specific versions). Rows defining ISO standard and vendor-specific languages may exist in the same table.

Each row has at least these columns and, if it makes an Open Group SQL conformance claim, the columns contain these values:

Column Name	Meaning	Open Group Values
SOURCE	The organization that defined this SQL version.	Open Group SQL
SOURCE_YEAR	The year the relevant source document was approved.	1992
CONFORMANCE	The conformance level to the relevant document that the implementation claims.	NULL
INTEGRITY	An indication of whether the implementation supports the Integrity Enhancement Feature (IEF).	YES
IMPLEMENTATION	A character string, defined by the vendor that uniquely identifies the vendor's SQL product.	TRUSTED RUBIX Release 6.0
BINDING_STYLE	A column included to envisage future adoption of direct, module or other binding styles.	ODBC
PROGRAMMING_LANG	The host language for with the binding style is supported.	C

↓ SECURITY

MAC The user must have MAC dominance over the database.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

TABLE_PRIVILEGES View

↓ FUNCTION

The TABLE_PRIVILEGES system view contains one row describing each case where a privilege was granted on an entire table.

↓ DESCRIPTION

TABLE_PRIVILEGES includes rows that represent the implicit privileges that a user has on the table. Within each row, there are the following columns:

GRANTOR: The user name of the user that granted the privilege in question. GRANTOR is always the user name of the user who granted the privilege.

GRANTEE: The user name of the user to whom the privilege in question was granted or “all”. Granting a privilege to “public” results in only one row in the TABLE_PRIVILEGES view (per privilege granted) and the GRANTEE column contains the value “all”.

TABLE_CAT: The name of the catalog that is the container for the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column.

TABLE_NAME: The name of the table in question.

PRIVILEGE_TYPE: The type of privilege that was granted. Its value must be one of the following: DELETE, INSERT, REFERENCES, SELECT, UPDATE.

IS_GRANTABLE: This column contains the value “yes” if the person invoking the SELECT on COLUMN_PRIVILEGES has the ability to further grant this privilege.

REMARKS: May contain descriptive information about the table.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)

- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

TABLES View

↓ FUNCTION

The TABLES system view contains exactly one row for each table to which the current user has access.

↓ DESCRIPTION

Each row of the TABLES system view contains at least these columns:

TABLE_CAT: The name of the catalog that is the container for the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column.

TABLE_NAME: The name of the table or view.

TABLE_TYPE: Identifies the type of the table or view. It can have two possible values: 'TABLE' or 'VIEW'.

REMARKS: May contain descriptive information about the table.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

UNIQUE_CONSTRAINTS view

↓ FUNCTION

The UNIQUE_CONSTRAINTS system view contains a row for each column whose value must be unique.

↓ DESCRIPTION

Each row contains these columns:

UNIQUE_CATALOG: The catalog holding the target table.

UNIQUE_SCHEMA: The schema holding the target table.

UNIQUE_TABLE: The target table holding the target column.

UNIQUE_COLUMN: The single target column to be unique.

UNIQUE_INDEXNAME: The name of the index providing uniqueness.

CONSTRAINT_NAME: the name of the constraint.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the view it refers to. Thus, if you do not have MAC dominance over a view, those rows do not show up in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

USAGE_PRIVILEGES View

↓ FUNCTION

The `USAGE_PRIVILEGES` system view contains one row describing each case where a privilege was granted on a character set or a collation.

↓ DESCRIPTION

Within each row of the `USAGE_PRIVILEGES` system view there are the following columns:

GRANTOR: The user name of the user that granted the privilege in question. `GRANTOR` is always the user name of the user who granted the privilege.

GRANTEE: The user name of the user to whom the privilege in question was granted. Granting a privilege to “public” results in only one row in the `USAGE_PRIVILEGES` view (per privilege granted) and the `GRANTEE` column contains the value “public”.

OBJECT_CAT: The name of the catalog that contains the object in question.

OBJECT_SCHEM: The name of the schema that contains the object in question.

OBJECT_NAME: The name of the object in question.

OBJECT_TYPE: The type of the object. Its value is ‘CHARACTER SET’ or ‘COLLATION’.

PRIVILEGE_TYPE: The type of privilege that was granted. Its value must be ‘USAGE’.

IS_GRANTABLE: This column contains the value “yes” if the person invoking the `SELECT` on `COLUMN_PRIVILEGES` has the ability to further grant this privilege.

REMARKS: May contain descriptive information about the table.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the table it refers to. Thus, if you do not have MAC dominance over a table, those rows do not appear in a `SELECT` of this view.

DAC A user can see a particular row in this view if the user has the `rubix.dac.select.dbname` authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on `SYSTEM_CATALOG` (**RA008**)
- EXEC on `INFO_SCHEM` schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the `OBJECT_CAT` column of this view (else the row is not visible).
- READ on the schema listed in the `OBJECT_SCHEM` column of this view (else the row is not visible).

AUDIT See the `SELECT` command in the **SQL Reference Guide**.

VIEWS View

↓ FUNCTION

The VIEWS system view contains exactly one row for each viewed table to which the current user has access.

↓ DESCRIPTION

Each row of the VIEWS system view contains at least these columns:

TABLE_CAT: The name of the catalog that is the container for the table named in the TABLE_NAME column.

TABLE_SCHEM: The name of the schema that is the container for the table named in the TABLE_NAME column.

TABLE_NAME: The name of the table or view.

VIEW_DEFINITION: The definition of the view as it would appear in a CREATE VIEW statement.

CHECK_OPTION: Contains the value 'CASCADED' if WITH CHECK OPTION was specified in the CREATE VIEW statement that created the table, and the value 'NONE' otherwise.

IS_UPDATABLE: Contains the value “yes” if the table is updateable and “no” otherwise.

REMARKS: May contain descriptive information about the view.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the view it refers to. Thus, if you do not have MAC dominance over a view, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)
- EXEC on the catalog listed in the TABLE_CAT column of this view (else the row is not visible).
- READ on the schema listed in the TABLE_SCHEM column of this view (else the row is not visible).

AUDIT See the SELECT command in the **SQL Reference Guide**.

APPLICATIONS View

↓ FUNCTION

The APPLICATIONS system view contains exactly one row for each Application in the database.

↓ DESCRIPTION

Each row of the APPLICATIONS system view contains at least these columns:

APP_NAME: The name of the Application.

APP_TIMEOUT: The timeout after which the authentication for Applications Users expire.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the Application it refers to. Thus, if you do not have MAC dominance over an Application, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

APPLICATION_ADMINS View

↓ FUNCTION

The APPLICATION_ADMINS system view contains exactly one row for each Application Administrator in the database.

↓ DESCRIPTION

Each row of the APPLICATION_ADMINS system view contains at least these columns:

APP_NAME: The name of the Application.

APP_ADMIN: The name of the RDBMS user that has been designation an Application Administrator.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the Application Administrator it refers to. Thus, if you do not have MAC dominance over an Application Administrator, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.

APPLICATION_USERS View

↓ FUNCTION

The APPLICATION_USERS system view contains exactly one row for each Application User in the database.

↓ DESCRIPTION

Each row of the APPLICATION_USERS system view contains at least these columns:

APP_NAME: The name of the Application.

APP_USER_NAME: The name of the Application User.

APP_USER_ID: The numerical ID of the Application User.

PASSWORD: The SHA 256 hashed password and seed string for the Application User.

↓ SECURITY

MAC Every row in this view is at the same sensitivity label as the Application User it refers to. Thus, if you do not have MAC dominance over an Application User, those rows do not appear in a SELECT of this view.

DAC A user can see a particular row in this view if the user has the *rubix.dac.select.dbname* authorization or all of the following are true:

- EXEC on database (**RA001**)
- EXEC on SYSTEM_CATALOG (**RA008**)
- EXEC on INFO_SCHEM schema (**RA002**)
- SELECT on referenced columns of the info schema view (**RA003**)

AUDIT See the SELECT command in the **SQL Reference Guide**.