Windows Azure® SQL Database Permissions

Permission Syntax

Most permission statements have the format :

AUTHORIZATION PERMISSION ON SECURABLE::NAME TO PRINCIPAL;

- AUTHORIZATION must be GRANT, REVOKE or DENY.
- PERMISSION is listed in the charts below.
- ON SECURABLE::NAME is the database, or database object and its name. Some permissions do not require ON SECURABLE::NAME.
- PRINCIPAL is the login, user, or role which receives or loses the permission. Grant permissions to roles whenever possible.

Sample grant statement:

GRANT UPDATE ON OBJECT::Production.Parts TO PartsTeam;

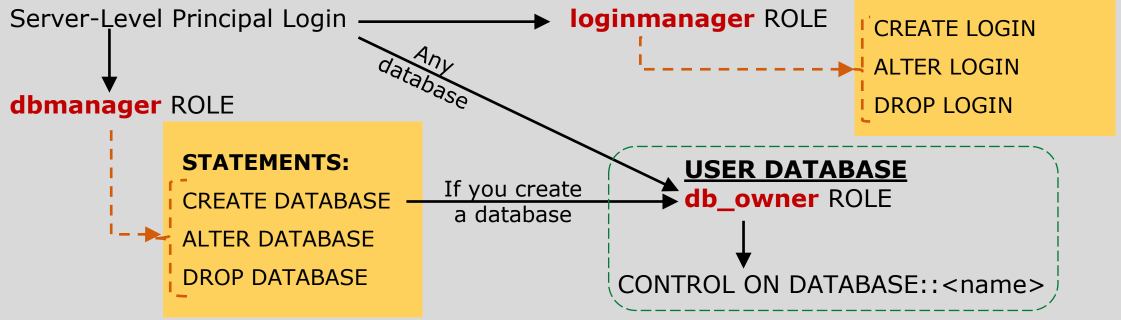
Denying a permission at any level, overrides a related grant.

To remove a previously granted permission, use REVOKE; not DENY.

How to Read this Chart

- Most of the more granular permissions are included in more than one higher level scope permission. So permissions can be inherited from more than one type of higher scope.
- Black, green, and blue arrows and boxes point to subordinate permissions that are included in the scope of higher a level permission.
- Brown arrows and boxes indicate some of the statements that can use the permission.

Permissions Outside the Database



NOTES:

- The server-level principal login is created by the provisioning process and has all permissions on the SQL Database server.
- The CONTROL DATABASE permission and the members of the db_owner role have all permissions on the database.
- Permissions do not imply role memberships and role memberships do not grant permissions. (E.g. The CONTROL DATABASE permission does not imply membership in the db_owner fixed database role. Membership in the db_owner role does not grant the CONTROL DATABASE permission.) However, it is sometimes possible to impersonate between roles and equivalent permissions.
- Granting any permission on a securable allows VIEW DEFINITION on that securable. It is an implied permissions and it cannot be revoked, but it can be explicitly denied by using the DENY VIEW DEFINITION statement.

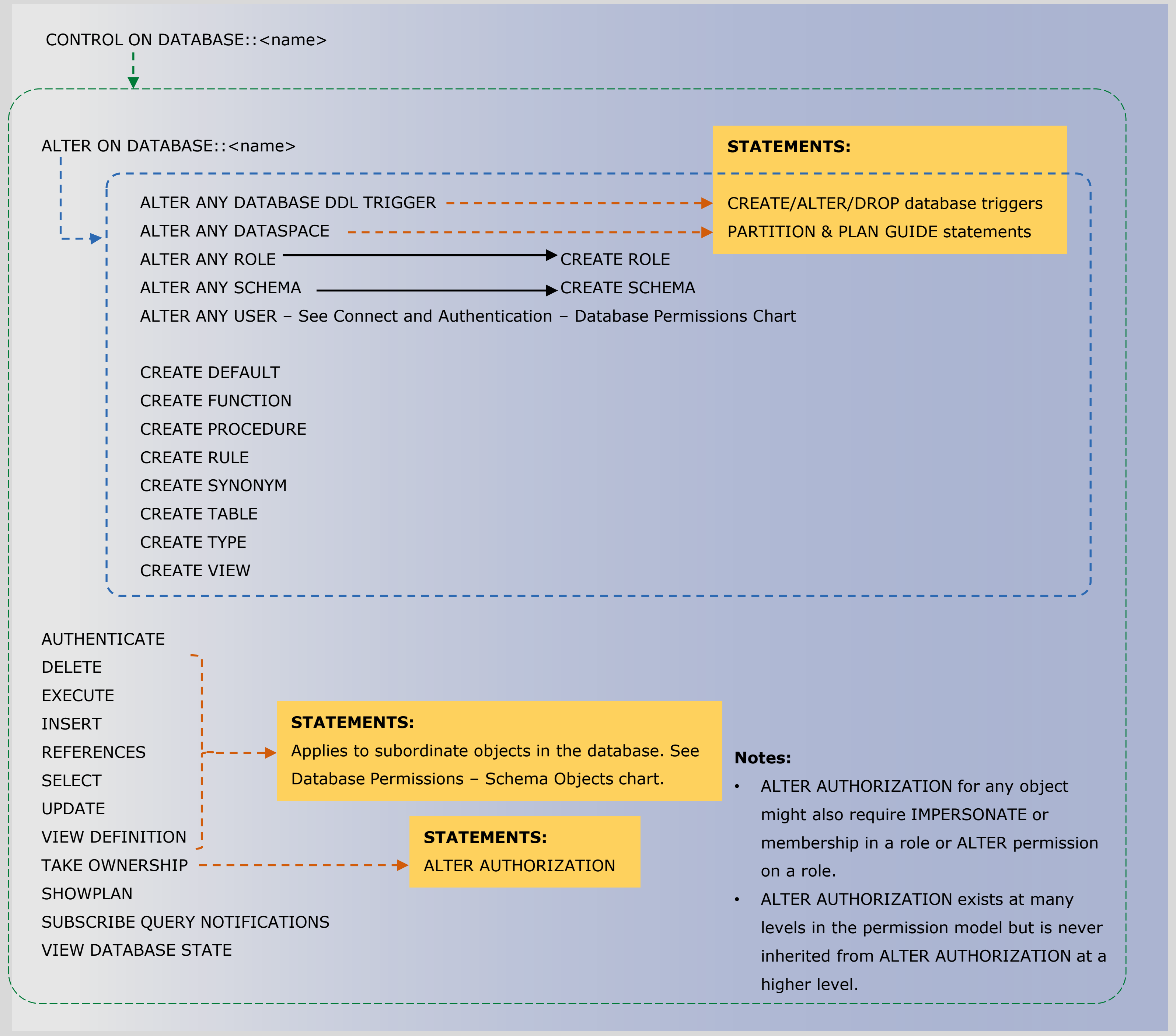
Questions and comments to **Rick.Byham@Microsoft.com**

Microsoft®

June 18, 2012

Database Level Permissions

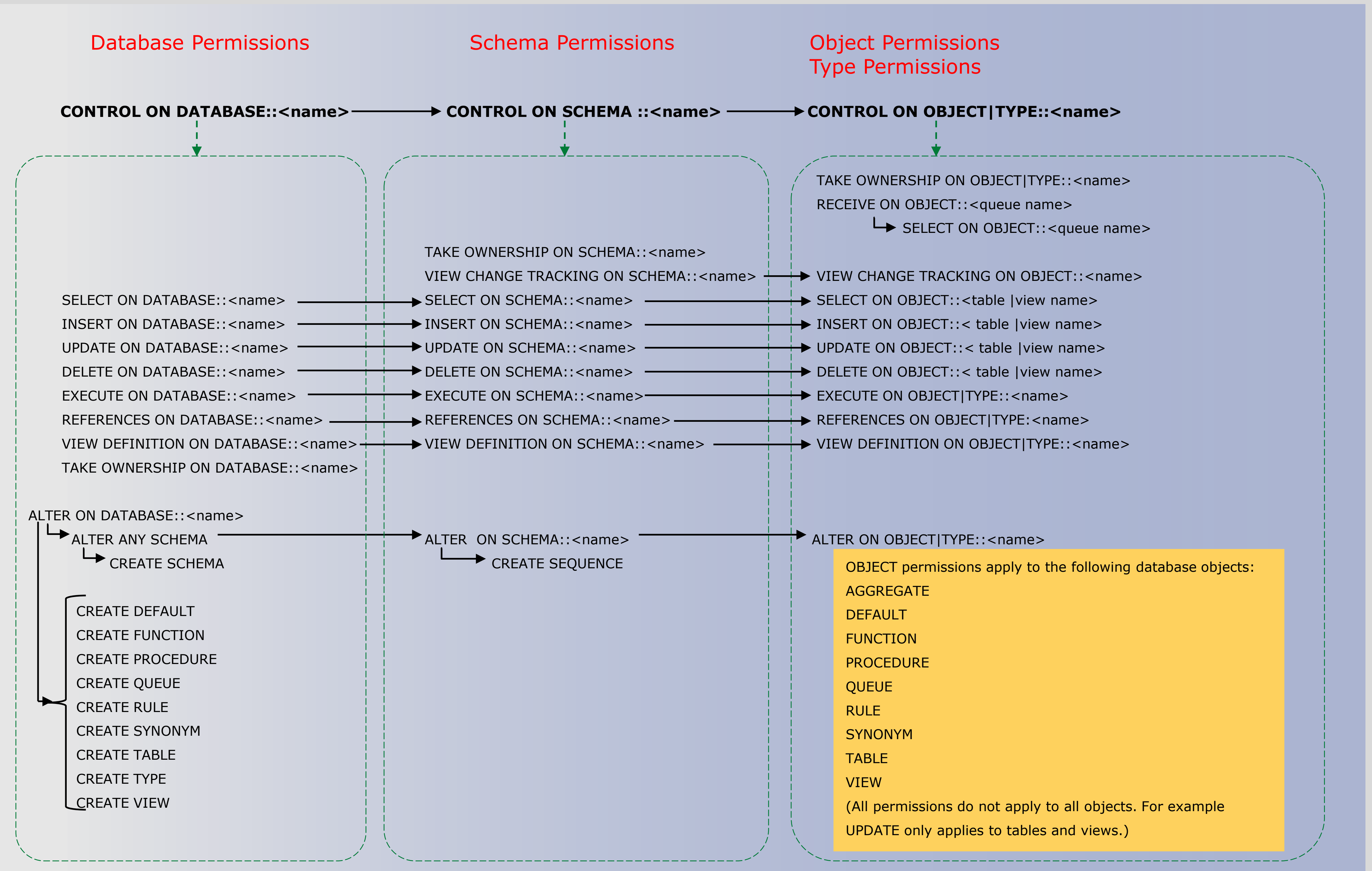
Top Level Database Permissions


CONTROL ON DATABASE::<name>
ALTER ON DATABASE::<name>
STATEMENTS:
ALTER ANY DATABASE DDL TRIGGER
CREATE/ALTER/DROP database triggers
ALTER ANY DATASPACE
PARTITION & PLAN GUIDE statements
ALTER ANY ROLE
CREATE ROLE
ALTER ANY SCHEMA
CREATE SCHEMA
ALTER ANY USER – See Connect and Authentication – Database Permissions Chart
CREATE DEFAULT
CREATE FUNCTION
CREATE PROCEDURE
CREATE RULE
CREATE SYNONYM
CREATE TABLE
CREATE TYPE
CREATE VIEW
AUTHENTICATE
DELETE
EXECUTE
INSERT
REFERENCES
SELECT
UPDATE
VIEW DEFINITION
TAKE OWNERSHIP
SHOWPLAN
SUBSCRIBE QUERY NOTIFICATIONS
VIEW DATABASE STATE
STATEMENTS:
Applies to subordinate objects in the database. See Database Permissions – Schema Objects chart.
STATEMENTS:
ALTER AUTHORIZATION

Notes:

- ALTER AUTHORIZATION for any object might also require IMPERSONATE or membership in a role or ALTER permission on a role.
- ALTER AUTHORIZATION exists at many levels in the permission model but is never inherited from ALTER AUTHORIZATION at a higher level.

Database Permissions – Schema Objects

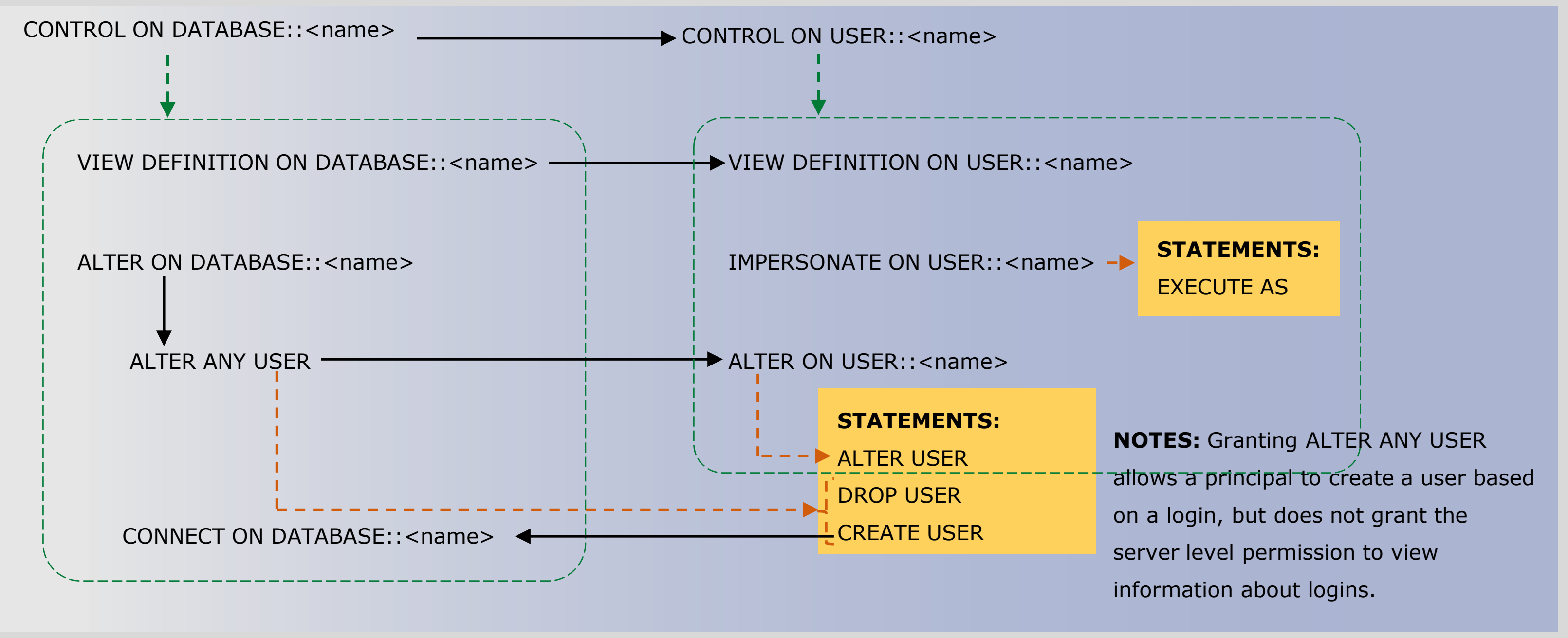

Database Permissions
Schema Permissions
Object Permissions
Type Permissions
CONTROL ON DATABASE::<name>
CONTROL ON SCHEMA ::<name>
CONTROL ON OBJECT|TYPE::<name>
TAKE OWNERSHIP ON OBJECT|TYPE::<name>
RECEIVE ON OBJECT::<queue name>
SELECT ON OBJECT::<queue name>
TAKE OWNERSHIP ON SCHEMA::<name>
VIEW CHANGE TRACKING ON SCHEMA::<name>
VIEW CHANGE TRACKING ON OBJECT::<name>
SELECT ON DATABASE::<name>
SELECT ON SCHEMA::<name>
SELECT ON OBJECT::<table |view name>
INSERT ON DATABASE::<name>
INSERT ON SCHEMA::<name>
INSERT ON OBJECT::< table |view name>
UPDATE ON DATABASE::<name>
UPDATE ON SCHEMA::<name>
UPDATE ON OBJECT::< table |view name>
DELETE ON DATABASE::<name>
DELETE ON SCHEMA::<name>
DELETE ON OBJECT::< table |view name>
EXECUTE ON DATABASE::<name>
EXECUTE ON SCHEMA::<name>
EXECUTE ON OBJECT|TYPE::<name>
REFERENCES ON DATABASE::<name>
REFERENCES ON SCHEMA::<name>
REFERENCES ON OBJECT|TYPE:<name>
VIEW DEFINITION ON DATABASE::<name>
VIEW DEFINITION ON SCHEMA::<name>
VIEW DEFINITION ON OBJECT|TYPE::<name>
TAKE OWNERSHIP ON DATABASE::<name>
ALTER ON DATABASE::<name>
ALTER ANY SCHEMA
CREATE SCHEMA
ALTER ON SCHEMA::<name>
CREATE SEQUENCE
ALTER ON OBJECT|TYPE::<name>
CREATE DEFAULT
CREATE FUNCTION
CREATE PROCEDURE
CREATE QUEUE
CREATE RULE
CREATE SYNONYM
CREATE TABLE
CREATE TYPE
CREATE VIEW

OBJECT permissions apply to the following database objects:
AGGREGATE
DEFAULT
FUNCTION
PROCEDURE
QUEUE
RULE
SYNONYM
TABLE
VIEW
(All permissions do not apply to all objects. For example UPDATE only applies to tables and views.)

Notes:

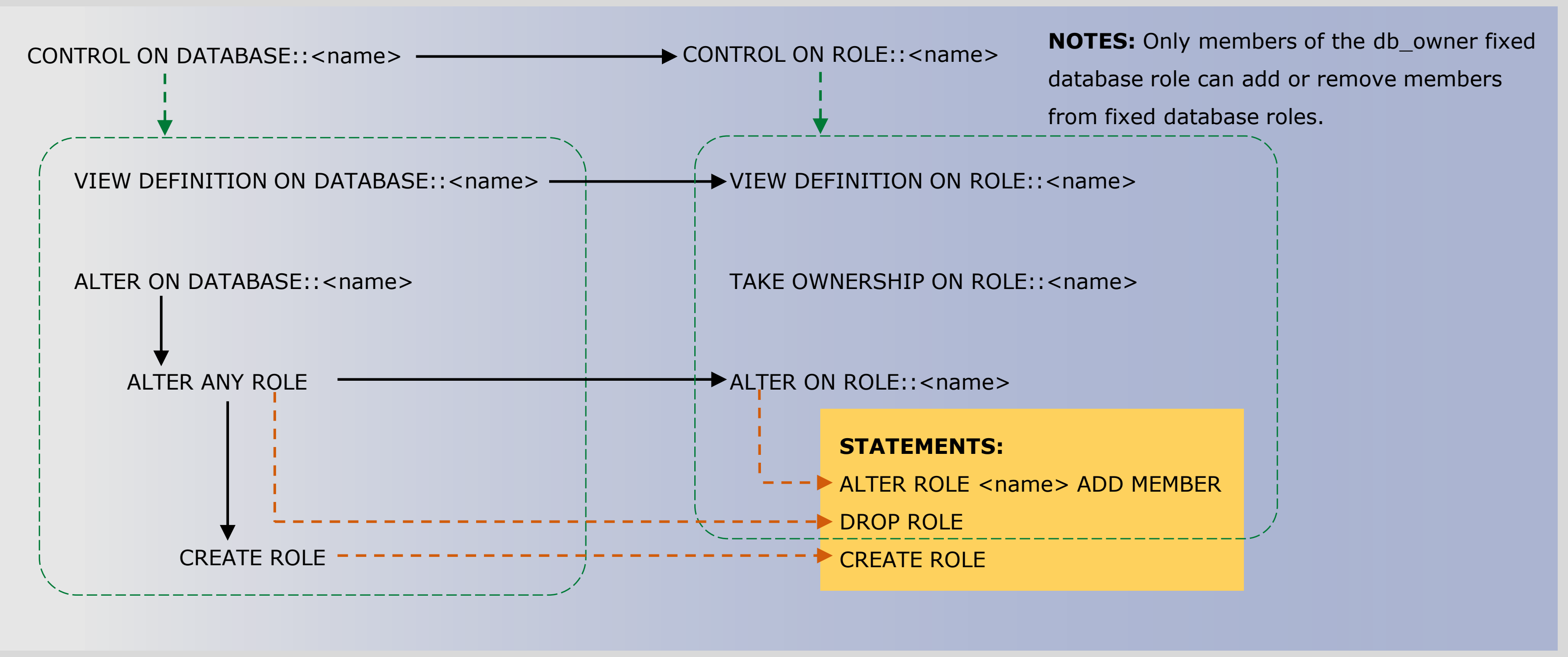
- To create a schema object (such as a table) you must have CREATE permission for that object type plus ALTER ON SCHEMA::<name> for the schema of the object.
- To alter an object (such as a table) you must have ALTER permission on the object (or schema),or CONTOL permission on the object.
- To drop an object (such as a table) you must have ALTER permission on the schema or CONTOL permission on the object.
- To create an index requires ALTER OBJECT::<name> permission on the table or view.
- To create or alter a trigger on a table or view requires ALTER OBJECT::<name> on the table or view.
- To create statistics requires ALTER OBJECT::<name> on the table or view.

Connect and Authentication – Database Permissions


CONTROL ON DATABASE::<name>
CONTROL ON USER::<name>
VIEW DEFINITION ON DATABASE::<name>
VIEW DEFINITION ON USER::<name>
ALTER ON DATABASE::<name>
IMPERSONATE ON USER::<name>
STATEMENTS:
EXECUTE AS
ALTER ANY USER
ALTER ON USER::<name>
STATEMENTS:
ALTER USER
DROP USER
CREATE USER
CONNECT ON DATABASE::<name>

NOTES: Granting ALTER ANY USER allows a principal to create a user based on a login, but does not grant the server level permission to view information about logins.

Database Role Permissions


CONTROL ON DATABASE::<name>
CONTROL ON ROLE::<name>
VIEW DEFINITION ON DATABASE::<name>
VIEW DEFINITION ON ROLE::<name>
ALTER ON DATABASE::<name>
TAKE OWNERSHIP ON ROLE::<name>
ALTER ANY ROLE
ALTER ON ROLE::<name>
STATEMENTS:
ALTER ROLE <name> ADD MEMBER
DROP ROLE
CREATE ROLE
CREATE ROLE

NOTES: Only members of the db_owner fixed database role can add or remove members from fixed database roles.