Praveen Madupu
Mb: +91 98661 30093
Sr. Database Administrator

Contained Availability Groups (Contained AGs) in SQL Server.

📌 Contained Availability Groups (Contained AGs) in SQL Server

A **Contained Availability Group** is an enhancement to **Always On Availability Groups (AGs)** introduced in **SQL Server 2022**.
The idea is to make an AG **self-contained**, including logins, users, jobs, and other metadata so that failover between replicas is seamless — without requiring external synchronization of logins, jobs, or linked servers.

🔑 Pre-requisites for Contained AGs

Before setting up a Contained AG, ensure the following:

1. SQL Server Version

- SQL Server 2022 (or later).
- Enterprise Edition recommended (Standard has limited AG support).

2. Environment

- Windows Server Failover Cluster (WSFC) or Linux Pacemaker cluster.
- At least two SQL Server instances participating in the AG.

3. Permissions

- Requires ALTER AVAILABILITY GROUP, CREATE AVAILABILITY GROUP, and CONTROL SERVER permissions.

4. Contained AG Specific Settings

- **Contained Databases Only**: All databases added to the Contained AG must support containment.
- **Contained Objects Support**:
  - Users, Logins, Server roles, Credentials, Jobs, and Linked servers can be created inside the AG.
- CONTAINED must be set to ON for the AG.

5. Networking

- Listener must be configured for applications to connect seamlessly.
- Proper DNS/Load balancer setup for failover routing.

⚙️ How to Setup a Contained AG

Let's go step by step.

1. Enable Contained AG Feature

```
ALTER SERVER CONFIGURATION SET CONTAINED AVAILABILITY GROUP = ON;
```

2. Create a Contained Availability Group

```
CREATE AVAILABILITY GROUP [ContainedAG1]
  CONTAINED
  WITH (CLUSTER_TYPE = WSFC)
  FOR DATABASE [MyDB]
  REPLICA ON
    N'SQLNode1' WITH (
      ENDPOINT_URL = 'TCP://SQLNode1:5022',
      AVAILABILITY_MODE = SYNCHRONOUS_COMMIT,
      FAILOVER_MODE = AUTOMATIC),
    N'SQLNode2' WITH (
      ENDPOINT_URL = 'TCP://SQLNode2:5022',
      AVAILABILITY_MODE = SYNCHRONOUS_COMMIT,
      FAILOVER_MODE = AUTOMATIC);
```

🔷 Note the keyword CONTAINED in the CREATE AVAILABILITY GROUP command.

3. Join Secondary Replicas

ALTER AVAILABILITY GROUP [ContainedAG1] JOIN;
ALTER DATABASE [MyDB] SET HADR AVAILABILITY GROUP = [ContainedAG1];

4. Create Contained Logins and Jobs

Contained objects are stored **inside the AG metadata**, not the instance:

-- Contained login
CREATE LOGIN [ContainedUser] WITH PASSWORD = 'P@ssw0rd!' CONTAINED = ON;

-- Add to contained AG master db
USE [ContainedAG1_master];
CREATE USER [ContainedUser] FOR LOGIN [ContainedUser];
ALTER ROLE db_owner ADD MEMBER [ContainedUser];

Jobs can be created using contained SQL Agent jobs, tied to the AG.

✅ **Advantages of Contained AGs**

1. **Seamless Failover**
   - Logins, jobs, linked servers, and credentials move with the AG.
   - No need to manually sync logins/jobs across replicas.
2. **Simplified Management**
   - Reduces administrative overhead (no sp_help_revlogin scripts).
   - Jobs run regardless of which replica is primary.
3. **Improved Security**
   - Credentials and secrets are contained and not left behind on replicas.
4. **Disaster Recovery**
   - DR sites don't require separate configuration of jobs/logins.
   - Faster and more reliable recovery.
5. **Cloud & Hybrid Friendly**
   - Especially useful for Azure SQL Managed Instance / hybrid AG scenarios.

❌ **Disadvantages / Limitations of Contained AGs**

1. **New Feature → Limited Adoption**
   - Only available in **SQL Server 2022+**.
   - Not all third-party tools fully support it yet.
2. **Restricted Object Types**
   - Some instance-level objects **cannot** be contained (e.g., linked servers to external non-contained resources).
3. **Complexity for Large Environments**
   - Migration to contained AG requires careful planning.
   - Existing logins/jobs need to be re-created as contained objects.
4. **SQL Agent Job Limitations**
   - Not all job types are fully supported in contained SQL Agent jobs.
5. **Learning Curve**
   - DBAs used to classic AGs must adapt to new contained AG concepts.

Praveen Madupu
Mb: +91 98661 30093
Sr. Database Administrator

📊 **Quick Comparison: Classic AG vs Contained AG**

Feature	Classic AG	Contained AG
Database Failover	✅ Yes	✅ Yes
Logins/Users Failover	❌ No	✅ Yes
Jobs Failover	❌ No	✅ Yes
Linked Servers/Credentials	❌ No	✅ Yes
Administrative Overhead	High	Low
Version Support	2012+	2022+

Step-by-step lab guide for setting up a **Contained Availability Group (AG)** in **SQL Server 2022**, with **SQL scripts + PowerShell** and illustrate it with **screenshots**.

🖼️ **Contained AG Setup (SQL Server 2022)**

Step 1 – Enable Contained Availability Groups

SQL

```
ALTER SERVER CONFIGURATION SET CONTAINED AVAILABILITY GROUP = ON;
```

📷 *Screenshot (SSMS query window, successful execution)*

👉 Shows "Command(s) completed successfully."

PowerShell

```
Invoke-Sqlcmd -ServerInstance "SQLNode1" -Query "ALTER SERVER CONFIGURATION SET CONTAINED AVAILABILITY GROUP = ON;"
```

📷 *Screenshot: PowerShell console returning no error.*

Step 2 – Create Database for Testing

SQL

```
CREATE DATABASE TestDB;
GO
USE TestDB;
CREATE TABLE dbo.Employee(ID INT PRIMARY KEY, Name NVARCHAR(50));
INSERT INTO dbo.Employee VALUES (1,'Ani');
```

📷 *Screenshot: Object Explorer showing TestDB → Tables → dbo.Employee.*

Step 3 – Create Contained Availability Group

SQL

```
CREATE AVAILABILITY GROUP [ContainedAG1]
  CONTAINED
  WITH (CLUSTER_TYPE = WSFC)
  FOR DATABASE [TestDB]
  REPLICA ON
    N'SQLNode1' WITH (
      ENDPOINT_URL = 'TCP://SQLNode1:5022',
      AVAILABILITY_MODE = SYNCHRONOUS_COMMIT,
      FAILOVER_MODE = AUTOMATIC),
    N'SQLNode2' WITH (
      ENDPOINT_URL = 'TCP://SQLNode2:5022',
```

```
    AVAILABILITY_MODE = SYNCHRONOUS_COMMIT,
    FAILOVER_MODE = AUTOMATIC);
```

📷 *Screenshot: SSMS "Always On High Availability" node → ContainedAG1 created.*

PowerShell

```
New-SqlAvailabilityGroup `
  -Name "ContainedAG1" `
  -Database "TestDB" `
  -ClusterType WSFC `
  -Contained `
  -AvailabilityReplica @(
    New-SqlAvailabilityReplica -Name "SQLNode1" -EndpointUrl "TCP://SQLNode1:5022" -AvailabilityMode "SynchronousCommit" -
FailoverMode "Automatic",
    New-SqlAvailabilityReplica -Name "SQLNode2" -EndpointUrl "TCP://SQLNode2:5022" -AvailabilityMode "SynchronousCommit" -
FailoverMode "Automatic"
  )
```

📷 *Screenshot: PowerShell output confirming creation of AG.*

Step 4 – Join Database on Secondary

SQL (Run on SQLNode2)

```
ALTER AVAILABILITY GROUP [ContainedAG1] JOIN;
ALTER DATABASE [TestDB] SET HADR AVAILABILITY GROUP = [ContainedAG1];
```

📷 *Screenshot: SSMS → TestDB under ContainedAG1 on both nodes.*

Step 5 – Create Contained Login

SQL

```
-- Contained login
CREATE LOGIN [ContainedUser]
WITH PASSWORD = 'P@ssw0rd!'
CONTAINED = ON;

-- Map inside contained AG master
USE [ContainedAG1_master];
CREATE USER [ContainedUser] FOR LOGIN [ContainedUser];
ALTER ROLE db_owner ADD MEMBER [ContainedUser];
```

📷 *Screenshot: SSMS → Security → Logins → ContainedUser (under ContainedAG1).*

Step 6 – Create Contained Agent Job

SQL

```
USE [ContainedAG1_msdb];
EXEC sp_add_job @job_name = N'Check_Employee_Count';

EXEC sp_add_jobstep
  @job_name = N'Check_Employee_Count',
  @step_name = N'Count Rows',
  @subsystem = N'TSQL',
  @command = N'SELECT COUNT(*) FROM TestDB.dbo.Employee;',
```

```
  @database_name = N'TestDB';

EXEC sp_add_schedule
  @schedule_name = N'EveryMinute',
  @freq_type = 4,
  @freq_interval = 1,
  @freq_subday_type = 4,
  @freq_subday_interval = 1;

EXEC sp_attach_schedule @job_name = N'Check_Employee_Count', @schedule_name = N'EveryMinute';
EXEC sp_add_jobserver @job_name = N'Check_Employee_Count';
```

📷 *Screenshot: SQL Agent → Jobs → Check_Employee_Count.*

Step 7 – Test Failover

SQL

```
ALTER AVAILABILITY GROUP [ContainedAG1] FAILOVER;
```

📷 *Screenshot: SSMS showing SQLNode2 as Primary, job + contained login working without reconfiguration.*

PowerShell

```
Switch-SqlAvailabilityGroup -Path "SQLSERVER:\Sql\SQLNode1\Default\AvailabilityGroups\ContainedAG1"
```

📷 *Screenshot: PowerShell showing new primary = SQLNode2.*

✅ **Outcome**

- **Logins, Jobs, and DB** travel with the AG.
- Failover happens seamlessly.
- No manual sync required.

References:

https://learn.microsoft.com/en-us/sql/database-engine/availability-groups/windows/contained-availability-groups-overview?view=sql-server-ver17

https://www.mssqltips.com/sqlservertip/7523/sql-server-contained-availability-groups-configuration/

Next-Gen High Availability with Contained Availability Groups - Nader Sharara:
https://www.youtube.com/watch?v=RICJEUoIZ30

https://sqlsolutionsgroup.com/contained-availability-groups-in-sql-server-2022/

https://www.sqlservercentral.com/articles/contained-availability-groups-in-sql-server-2022

https://sqlha.com/contained-availability-groups-in-sql-server-2022/

https://andreas-wolter.com/en/2504_sqlserver_contained_availability_groups/