

می‌خواهم دوتا سرور sql را بصورت ha را بیاندازیم لطفا
راهنمایی کن بصورت کامل و قدم به قدم

Thought for 14s >

عالی — می‌ریم سر اصل مطلب. در ادامه یک راهنمای قدم‌به‌قدم کامل و عملی برای راه‌اندازی **High Availability (HA)** با دو سرور SQL تهیه کردم. من دو گزینه‌ی متداول و عملی را توضیح می‌دهم و بعد قدم‌به‌قدم پیاده‌سازی **Always On Availability Groups (AG)** را — چون برای دو نود در محیط‌های مجازی/کلاود یا دیتاسنترهای امروزی معمول‌ترین و منعطف‌ترین گزینه است. (اگر می‌خواهی به‌جای AG از **Failover Cluster Instance** با **shared storage** استفاده کنی، بخش مقایسه و نکات FCI را پایین گذاشتم).

قبل از هر چیز: این راهنما فرض می‌کند هر دو سرور ویندوزی هستند و در دامنه (Active Directory) عضو شده‌اند و نسخه‌ی SQL مناسب (مثلاً SQL Server 2016+؛ برخی امکانات در Standard/Enterprise فرق دارد) روی هر دو نصب است.

1) خلاصه معماری و گزینه‌ها — کدام را انتخاب کنیم؟

• Always On Availability Groups (AG)

- مزایا: نیاز به اشتراک‌گذاری دیسک ندارد، مناسب VM و کلاود، امکان replica خواندنی، failover سریع، قابل استفاده برای disaster recovery. برای Standard edition می‌توان از **Basic Availability Group** (محدود به 1 DB و 2 replica) استفاده کرد؛ نسخه‌ی کامل AG نیاز به Enterprise برای برخی قابلیت‌ها دارد. 1+ Microsoft Learn

• Failover Cluster Instance (FCI)

- مزایا: failover در سطح instance (تمام دیتابیس‌ها)، برای workloadهای خاص مناسب است. معایب: نیاز به **shared storage (SAN / SMB / S2D)** بین نودها دارد.

1+ Microsoft Learn

برای دو سرور معمولاً توصیه می‌شود از **Always On AG** (یا اگر محدود به Standard هستی — **Basic AG**) استفاده شود چون نصب و نگهداری ساده‌تر در VM / کلاود است و نیازی به **shared storage** ندارد.

2) چک‌لیست پیش‌نیازها (قبل از شروع عملی)

1. نسخه‌ها / لایسنس:

- Windows Server (نسخه سازگار)، و SQL Server 2016 + (برای Basic AG در Standard از SQL2016 + پشتیبانی شده؛ AG کامل برای برخی امکانات نیاز به Enterprise دارد).

Microsoft Learn

2. دامنه (Active Directory): هر دو سرور باید عضو دامنه باشند.

3. Windows Server Failover Clustering (WSFC) باید ایجاد شود (برای AG لازم است یک کلاستر

WSFC داشته باشی). Microsoft Learn

4. شبکه:

- نام DNS و IP ثابت برای هر نود.
- پورت TCP برای دیتابیس mirroring endpoint (پیش فرض 5022) باز باشد بین نودها.

5. حساب‌ها / دسترسی‌ها:

- حساب دامنه برای نصب/پیکربندی (Admin روی سرورها).
- سرویس SQL Server باید با حساب مناسب (Domain account) اجرا شود اگر نیاز به Kerberos/گواهی دارد.

6. زمان‌بندی نسخه‌ها/patch: هر دو سرور بایستی Windows Update و SQL patch های ضروری را داشته باشند.

7. نسخه پایگاه داده: دیتابیس که قرار است AG شود باید در حالت FULL recovery باشد و بکاپ کامل + log آماده باشد.

(منابع MS برای prereqs و WSFC را ببین). 1+ Microsoft Learn

(3) طراحی شبکه و IP

- برای AG معمولاً یک Listener ایجاد می‌کنیم که یک نام DNS و یک یا چند IP دارد — کلاینت‌ها به Listener وصل می‌شوند و پشت‌صحنه به Primary هدایت می‌شوند. Microsoft Learn
- پورت‌ها: 5022 برای database mirroring endpoint، همچنین پورت TCP 1433 را در صورت نیاز (با پورت‌های instance-specific) مدیریت کن.
- اگر کلاستر در subnets/region متفاوت است، ممکن است نیاز به load balancer یا multi-subnet listener باشد (پلتفرم مثل Azure اوپشن‌های مخصوص دارد). ... Documentation

(4) قدم به قدم پیاده‌سازی (Always On AG — سناریوی دو نود)

الف) آماده‌سازی ویندوز و WSFC

1. روی هر دو سرور: نصب نقش (Server Manager → Add Roles and Features → Features → Failover Clustering) یا با PowerShell:

Copy code

powershell

```
Install-WindowsFeature -Name Failover-Clustering -IncludeManagementTools
```

2. بررسی سازگاری کلاستر (Validate-Cluster):
• در یکی از سرورها:

Copy code

powershell

```
Test-Cluster -Node ServerA,ServerB
```

- اگر همه‌ی تست‌ها OK شد، کلاستر را ایجاد کن:

Copy code

powershell

```
New-Cluster -Name MyCluster -Node ServerA,ServerB -StaticAddress 10.0.0.50
```

(در صورت نیاز، از witness (file share witness یا cloud witness) استفاده کن برای quorum).

Microsoft Learn

ب) نصب و پیکربندی SQL Server

1. روی هر دو نود، نصب SQL Server (همان ورژن و همان instance name).
2. در هر instance، از SQL Server Configuration Manager گزینه‌ی **Always On Availability Groups** را فعال کن:

- instance → Properties → Always On High Availability → SQL Server Services → Availability → تیک Restart service → Enable Always On Availability Groups

(این کار باید روی هر نود انجام شود). Microsoft Learn

پ) آماده‌سازی دیتابیس‌ها (در primary)

1. تنظیم recovery model به FULL:

Copy code

sql

```
ALTER DATABASE MyDB SET RECOVERY FULL;
```

2. گرفتن بکاپ کامل و بکاپ تراکنش لوگ:

Copy code

sql

```
BACKUP DATABASE MyDB TO DISK='C:\backups\MyDB.bak' WITH FORMAT;
BACKUP LOG MyDB TO DISK='C:\backups\MyDB.trn';
```

ت) ایجاد database mirroring endpoint (روی هر دو سرور)

- روی Primary و Secondary هر کدام یک endpoint بساز:

Copy code

sql

```
CREATE ENDPOINT [Hadr_endpoint]
STATE=STARTED
AS TCP (LISTENER_PORT = 5022)
FOR DATA_MIRRORING (ROLE = ALL, AUTHENTICATION = WINDOWS NEGOTIATE);
```

(لیسنر پورت را چک کن و فایروال باز باشد). Microsoft Learn

ث) ریستور روی Secondary برای تهیه Secondary databases

- روی Secondary:

Copy code

sql

```
RESTORE DATABASE MyDB FROM DISK='\\primary\backups\MyDB.bak' WITH NORECOVERY;
RESTORE LOG MyDB FROM DISK='\\primary\backups\MyDB.trn' WITH NORECOVERY;
```

(هدف این است که فایل دیتابیس Secondary در حالت NORECOVERY باقی بماند تا به AG اضافه

شود). Microsoft Learn

ج) ایجاد Availability Group (با SSMS یا T-SQL)

- با SSMS: Wizard → New Availability Group → انتخاب دیتابیس‌ها → پیکربندی replicas و failover mode (automatic/manual) و readable secondary → ایجاد listener.
- نمونه T-SQL (ساده شده):

Copy code

sql

```
CREATE AVAILABILITY GROUP [AG_MyApp]
WITH (AUTOMATED_BACKUP_PREFERENCE = PRIMARY)
FOR DATABASE MyDB
REPLICA ON
'SERVERA' WITH (ENDPOINT_URL='TCP://servera:5022', AVAILABILITY_MODE = SYNCHRON
```

'SERVERB' WITH (ENDPOINT_URL='TCP://serverb:5022', AVAILABILITY_MODE = SYNCHRON

Basic AG — اگر از Standard Basic AG استفاده می‌کنی، از گزینه BASIC در CREATE استفاده کن
 محدودیت دارد). 1+ Microsoft Learn

چ) ایجاد Listener

- با SSMS یا T-SQL listener را ایجاد کن (Name و IP که به کلاینت‌ها ارائه می‌شود). نمونه T-SQL:

Copy code

sql

```
ALTER AVAILABILITY GROUP [AG_MyApp]
ADD LISTENER 'AGListener' (WITH IP ((N'10.0.0.60', N'255.255.255.0')));
```

یا از دستورهای Failover Cluster Manager برای اضافه کردن Client Access Point استفاده کن.

1+ Microsoft Learn

ح) بررسی و تست failover

1. در SSMS → Always On Dashboard یا با PowerShell بررسی کن وضعیت replicas و synchronization.
2. Failover آزمایشی: در SSMS روی AG → Failover → follow wizard (برای failover اتوماتیک باید SYNCHRONOUS_COMMIT و automatic failover تنظیم شده باشد).
3. پس از failover، کلاینت‌ها باید بدون تغییر کانکشن به Listener وصل شوند.

5) نکات مهم عملی و خطاهای متداول

- Witness و Quorum: در 2 نود به‌تنهایی، باید حتماً یک witness داشته باشی (file share witness یا cloud witness) تا از مشکلات quorum جلوگیری شود. Microsoft Learn
- سخت‌افزار/شبکه: latency بالا بین نودها باعث مشکلات sync و افزایش تأخیر می‌شود.
- پورت‌ها / فایروال: پورت endpoint (پیش‌فرض 5022) باید باز و قابل دسترسی باشد.
- نسخه متناسب: اگر از Standard استفاده می‌کنی و چند دیتابیس نیاز داری، Basic AG محدودیت دارد؛ در این صورت باید Enterprise یا ترکیب دیگری در نظر بگیری. Microsoft Learn
- بکاپ‌ها: بکاپ full/transaction log را روی replicas مدیریت کن. در AG معمولاً بکاپ خواندنی را می‌شود روی secondary انجام داد (اگر secondary قابل خواندن باشد) تا بار روی primary کم شود.
- امنیت / حساب‌ها: اگر endpoint با certificate یا با Windows auth کار می‌کند باید حساب‌ها و

SPNs را درست تنظیم کنی.

(6) مثال های عملی (تکه کدهای مفید)

PowerShell: ایجاد کلاستر (مثالی)

Copy code

powershell

```
# Validate first
Test-Cluster -Node ServerA,ServerB

# Create cluster with static IP
New-Cluster -Name SQLCluster -Node ServerA,ServerB -StaticAddress 10.0.0.50

# Add file share witness (example)
Set-ClusterQuorum -Cluster SQLCluster -NodeAndFileShareMajority "\\fswitness\shar
```

T-SQL: نمونه کامل تر ایجاد AG (نمونه)

Copy code

sql

```
-- اجرا شود Primary روی سرور --
CREATE AVAILABILITY GROUP [AG_MyApp]
FOR DATABASE [MyDB]
REPLICA ON
    N'SERVERA' WITH (ENDPOINT_URL = N'TCP://servera:5022', FAILOVER_MODE = AUTOMATI
    N'SERVERB' WITH (ENDPOINT_URL = N'TCP://serverb:5022', FAILOVER_MODE = AUTOMATI

ALTER AVAILABILITY GROUP [AG_MyApp] GRANT CREATE ANY DATABASE;
```

(این تنها یک نمونه است — برای production باید پارامترها و security را تنظیم کنی). Microsoft Learn

(7) مانیتورینگ، نگهداری و DR

- از SQL Server Always On Dashboard در SSMS استفاده کن.
- لاگ ها و هشدارها را برای replica state, synchronization health و failover بررسی کن.
- اجرای منظم DR test و failover test را در برنامه نگهداری بگنجان.

- در آپدیت و patching: برای min downtime، می‌توان از قابلیت AG برای rolling upgrades استفاده کرد (در AG replica ها را یکی‌یکی آپدیت می‌کنی و failover انجام می‌دهی).

Microsoft Learn

(8) اگر بخوای FCI (Failover Cluster Instance)

- نیاز به shared storage (SMB share یا S2D یا SAN) که در همه نودها قابل دسترسی باشد. پس ساختار و طراحی storage و multipath و تنظیمات SAN مهم هستند. 1+ Microsoft Learn

(9) منابع رسمی که حتماً بخوان

- Microsoft Docs — Prereqs & recommendations for Always On AG — Microsoft Learn
- Microsoft Docs — Windows Server Failover Clustering with SQL Server — Microsoft Learn
- Microsoft Docs — Create/configure AG listener — Microsoft Learn
- Microsoft Docs — CREATE AVAILABILITY GROUP (T-SQL) + examples — Microsoft Learn 1+

(10) جمع‌بندی و پیشنهاد عملی (برای شروع سریع توی محیط خودت)

1. تصمیم نهایی: اگر دیتابیس‌های متعدد و قابلیت‌های پیشرفته می‌خواهی → Always On AG (Enterprise). اگر فقط یک DB و Standard داری → Basic AG.
2. آماده‌سازی WSFC + نصب SQL همسان روی دو نود.
3. ایجاد endpoint، بکاپ، ریستور روی secondary، سپس ساخت AG و Listener.
4. تست failover و مانیتورینگ.

اگر بخوای من می‌تونم همین حالا (بدون منتظر موندن) برایت:

- یک script کامل PowerShell + T-SQL آماده کنم که مراحل اصلی (validate cluster, create cluster, create endpoints, restore backups, create AG + listener) را شامل شود — با پارامترهای قابل تغییر (نام سرورها، مسیر بکاپ، آدرس‌ها).
 - یا یک checklist نهایی با چک‌باکس‌های قابل‌پرینت برای تیم نصب آماده کنم.
- بگو کدوم را می‌خواهی (اسکرپت یا checklist یا هر دو) — من فوراً یک نسخه قابل اجرا و قابل کپی

برایت می‌سازم.

    ...  Sources

من mirroring. را راه انداختم اما با برنامه های که نیاز به file stream دارند مشکل دارند در ضمن san هم ندارم و سرورها هم بصورت vm هستند

باشه علیرضا جان، کاملاً فهمیدم مشکل از کجاست. تو Database Mirroring راه افتاده، اما برنامه‌ها ت FILESTREAM / FileTable استفاده می‌کنن و در Mirroring پشتیبانی نمی‌شه — به همین دلیل با هر برنامه‌ای که نیاز به FILESTREAM دارد ارور می‌ده یا کار نمی‌کند.

این دقیقاً یکی از محدودیت‌های رسمی Mirroring هست:

- ❌ FILESTREAM در Database Mirroring پشتیبانی نمی‌شود
- ✓ فقط در Always On Availability Groups پشتیبانی می‌شود
- ✓ یا باید Failover Cluster Instance (FCI) با storage مشترک داشته باشی (که تو نداری)

و چون گفتی SAN نداری و سرورها روی VM هستند، پس:

🔥 **بهترین و تنها راه‌حل استاندارد برای FILESTREAM در محیط VM بدون SAN:**

👉 **Always On Availability Groups (AG)**

چرا؟

- AG کاملاً از FILESTREAM و FileTable پشتیبانی می‌کند
- نیاز به shared storage ندارد
- روی VM کاملاً OK هست
- دقیقاً برای همین سناریو (بدون SAN + نیاز به HA + نیاز به FILESTREAM) طراحی شده

! نتیجه‌گیری مهم

- Mirroring گزینه اشتباه برای FILESTREAM است

- تنها گزینه درست برای HA در VM بدون SAN = Always On AG

تو الآن چه کاری باید انجام بدی؟

دو سناریو داری:

سناریو ۱ — مهاجرت از Mirroring → به Always On AG

(این بهترین انتخاب است)

مراحل کلی:

1. خاموش کردن Mirroring
 2. ساخت WSFC (کلاستر ویندوز) روی دو VM
 3. فعال کردن Always On SQL در
 4. ایجاد AG
 5. منتقل کردن دیتابیس FILESTREAM به AG
- خبر خوب → همه اینها را بدون SAN می‌توانی انجام بدهی
فقط یک **file share witness** لازم داری برای quorum
(یک سرور ساده یا حتی یک VM کوچک با 2GB RAM کافی است)

سناریو ۲ — اگر نمی‌خواهی AG پیاده‌سازی کنی (غیرتوصیه‌ای)

هیچ روش HA دیگری در SQL برای FILESTREAM وجود ندارد.

- ✓ نه Log Shipping
- ✓ نه Mirroring
- ✓ نه Replication
- ✓ نه Backup/Restore
- ✗ هیچکدام FILESTREAM را هم‌زمان sync و failover نمی‌کنند.

تنها فناوری‌هایی که FILESTREAM را در HA پشتیبانی می‌کنند:

Failover Cluster Instance (FCI) (1)

→ نیاز به SAN / shared storage

→ تو نداری، پس نمی‌شود

Always On Availability Groups (AG) (2

→ تنها گزینه ممکن برای تو

اگر بخواهی من برایت انجام بدهم:

می‌توانم همین الان برایت:

✓ یک نقشه کامل مهاجرت از Mirroring به Always On AG بدهم

✓ تمام دستورات PowerShell + T-SQL

✓ تنظیمات FILESTREAM در AG

✓ ساخت Listener

✓ تنظیم failover

✓ چک‌لیست عملی با مراحل شماره‌ای دقیق برای اجرا

✨ فقط بگو:

نسخه SQL Server شما چیه؟ (Enterprise یا Standard)

و

FILESTREAM در چند دیتابیس فعال است؟

بر اساس آن یک راهنمای دقیق متناسب با محیطت تولید می‌کنم.



سرور ها نسخه enterprise هست

تمام چک لیست های بالا را بطور کامل انجام بده

— باشه علیرضا جان

الان یک چک‌لیست کامل، دقیق، مرحله‌به‌مرحله و اجرایی برای مهاجرت از Mirroring → به Always On Availability Groups (AG) می‌سازم.

این چک لیست شامل همه چیز هست: ویندوز، کلاستر، Listener، Failover، SQL، FILESTREAM، Validation و حتی تست نهایی.

نسخه شما Enterprise است، پس همه قابلیت ها فعال هستند و هیچ محدودیت نسخه ای نداریم.

✓ چک لیست کامل اجرای Always On Availability Groups (نسخه Enterprise + FILESTREAM)

📌 بخش 1 — پیش نیازها (قبل از شروع)

✓ سیستم عامل و شبکه

- هر دو سرور عضو Active Directory Domain هستند.
- هر دو سرور IP Static دارند.
- DNS روی هر دو سرور درست کار می کند.
- زمان (Time Sync) بین دو سرور هماهنگ باشد.
- یک VM کوچک (یا هر سرور دیگر) برای File Share Witness در دسترس است.

✓ SQL Server

- هر دو سرور SQL Server Enterprise نصب شده با همان Instance Name.
- تنظیمات Collation، Patch Level و Version هر دو کاملاً یکسان باشد.
- FILESTREAM در هر دو سرور ویندوز و SQL فعال باشد:

ویندوز

به مسیر:

Copy code

arduino

FSRM → File Services → File Server

SQL

در SSMS:

Copy code

mathematica

Properties → Advanced → Filestream → Full Access Enabled

✓ امنیت

- حساب SQL Server Service باید Domain Account باشد.
- پورت‌های زیر در Firewall هر دو سرور باز باشند:
 - TCP 1433 (یا پورت Instance)
 - TCP 5022 (Endpoint)
 - پورت Listener که بعداً تعریف می‌شود

Mirroring ✓

- وضع موجود Mirroring را بررسی کن
- آخرین Full Backup و Log Backup را نگه دار
- برنامه‌ریزی Downtime (برای مرحله Switch)

بخش 2 — ساخت Windows Server Failover Cluster (WSFC)

✓ 1. نصب Role‌های موردنیاز

روی هر دو سرور:

Copy code

pgsql

Server Manager → Add Features → Failover Clustering

یا با PowerShell:

Copy code

powershell

`Install-WindowsFeature Failover-Clustering -IncludeManagementTools`

✓ 2. اجرای Validate Cluster

روی یکی از نودها:

Copy code

powershell

`Test-Cluster -Node NodeA,NodeB`

همه تست‌ها باید Success باشد.

✓ 3. ایجاد کلاستر

Copy code

pgsql

New-Cluster -Name SQLCLUSTER -Node NodeA,NodeB -StaticAddress 10.0.0.50

4. اضافه کردن File Share Witness ✓

در Failover Cluster Manager:

Copy code

mathematica

More Actions → Configure Cluster Quorum → File Share Witness

بخش 3 — فعال کردن Always On در SQL

روی هر دو سرور:

در SQL Server Configuration Manager:

Copy code

pgsql

SQL Server Services → Instance → Properties → Always On Availability Groups → Ena

بعد از انجام Restart SQL Server instance →

بخش 4 — آماده سازی دیتابیس ها برای AG

1. خاموش کردن Mirroring ✓

در Primary:

Copy code

sql

ALTER DATABASE MyDB SET PARTNER OFF;

در Secondary:

Copy code

sql

```
ALTER DATABASE MyDB SET PARTNER OFF;
```

2. تغییر Recovery Model ✓

Copy code

sql

```
ALTER DATABASE MyDB SET RECOVERY FULL;
```

3. گرفتن Full Backup و Log Backup ✓

Copy code

sql

```
BACKUP DATABASE MyDB TO DISK='C:\Backup\MyDB.bak';
```

```
BACKUP LOG MyDB TO DISK='C:\Backup\MyDB.trn';
```

4. ریستور دیتابیس روی Secondary (با NORECOVERY) ✓

Copy code

sql

```
RESTORE DATABASE MyDB FROM DISK='\\NodeA\Backup\MyDB.bak' WITH NORECOVERY;
```

```
RESTORE LOG MyDB FROM DISK='\\NodeA\Backup\MyDB.trn' WITH NORECOVERY;
```

بخش 5 — ساخت Endpoint برای داده‌ها

روی هر دو سرور:

Copy code

sql

```
CREATE ENDPOINT Hadr_endpoint
```

```
STATE=STARTED
```

```
AS TCP (LISTENER_PORT = 5022)
```

```
FOR DATA_MIRRORING (ROLE = ALL);
```

بخش 6 — ساخت Availability Group

1. با SSMS (توصیه شده) ✓

مسیر:

Copy code

pgsql

Always On High Availability → New Availability Group Wizard

انتخاب‌ها:

• نام AG:

Copy code

nginx

AG_PROD

- انتخاب دیتابیس‌ها: MyDB
- انتخاب Replica ها: NodeA و NodeB
- Availability Mode:
 - Synchronous Commit
 - Failover Mode:
 - Automatic
 - Backup Preferences:
 - Prefer Secondary
 - Listener:
 - نام: AG-LISTENER
 - IP: 10.0.0.60
 - Port: 1433 یا پورت دلخواه

✓ 2. یا با T-SQL (اختیاری)

Copy code

sql

```
CREATE AVAILABILITY GROUP AG_PROD
FOR DATABASE MyDB
REPLICA ON
N'NODEA' WITH (
    ENDPOINT_URL = 'TCP://nodea:5022',
    AVAILABILITY_MODE = SYNCHRONOUS_COMMIT,
    FAILOVER_MODE = AUTOMATIC
),
N'NODEB' WITH (
    ENDPOINT_URL = 'TCP://nodeb:5022',
```

```
ENDPOINT_URL = 'TCP://nodeb:5022',  
AVAILABILITY_MODE = SYNCHRONOUS_COMMIT,  
FAILOVER_MODE = AUTOMATIC  
);
```

بخش 7 — اضافه کردن Listener

در SSMS داخل AG:

Copy code

sql

Add Listener

- نام: AG-LISTENER
- IP: 10.0.0.60
- Port: 1433

بخش 8 — پیگیربندی FILESTREAM و FileTable در AG

✓ مطمئن شو:

- FILESTREAM در هر دو نود فعال است
- مسیر root برای FILESTREAM روی هر دو نود یکسان است
- مجوزها در هر دو نود یکسان باشد
- دیتابیس شامل FILESTREAM بدون مشکل Sync می شود

نکته طلایی:

AG از FILESTREAM کاملاً پشتیبانی می کند ولی مسیر *local FILESTREAM* باید روی همه نودها یکسان باشد
مثال:

Copy code

makefile

D:\Data\FilestreamRoot\

بخش 9 — تست نهایی Failover

تست دستی:

در SSMS:

Copy code

pgsql

AG → Failover → **Select** NodeB **as new Primary**

بررسی کن:

- دیتابیس‌ها آنلاین باشند
- FILESTREAM درست کار کند
- برنامه‌ها با Listener وصل شوند
- Return failover → NodeA

بخش 10 — چک‌لیست بعد از فعال‌سازی

Always On Dashboard 

بررسی:

- Synchronizing
- Synchronous Commit
- Automatic Failover (Yes/Yes)
- Endpoints (Started)

Monitoring 

- SQL Agent Alerts
- AG برای Extended Events
- Health Events

Backup 

Backup jobs را روی **Secondary** تنظیم کن.
(Load را از Primary کم می‌کند.)

بخش 11 — چک‌لیست امنیتی (Security Hardening)

بخش 11 - چک لیست امنیتی (Security Hardening)

- غیرفعال کردن دسترسی عمومی به Endpoints
- تنظیم SPN برای Kerberos
- تنظیم TLS 1.2
- فعال سازی Log Shipping به عنوان Disaster Recovery (اختیاری)
- تنظیم گواهی برای Endpoints (در محیط های بین سازمانی)

بخش 12 - چک لیست نگهداری (Maintenance)

- Jobs:
 - Integrity Check
 - Index Rebuild
 - Statistics Update
 - Backup Cleanup
 - تست failover هر 3 ماه یکبار
 - نگه داشتن مستندات AG

اگر بخواهی، می توانم برایت بسازم: 🎯

پکیج کامل اسکریپت های اجرای AG شامل: 🛠️

- PowerShell ساخت کلاستر
- اسکریپت ساخت Endpoint
- اسکریپت ساخت AG
- اسکریپت Listener
- اسکریپت های Failover Test
- اسکریپت چک سلامت AG

یا حتی یک فایل PDF / Word کامل و آماده استفاده در شرکت برات بسازم.

علیرضا جان فقط بگو:

می خواهی اسکریپت کامل اجرایی را بدهم
یا

نسخه PDF/Word مستندسازی شده؟



بله لطفا

Thought for 7s

عالی علیرضا جان — بستهٔ اجرایی آماده شد.

دانلود فایل (شامل اسکریپت PowerShell، اسکریپت‌های T-SQL، و راهنمای README):

[Download the package \(ag_migration_package.zip\)](#)

داخل زیپ:

- cluster_create.ps1 — راه‌اندازی WSFC (پی‌کربندی اولیه، Validate، ساخت کلاستر، تنظیم quorum با File Share Witness). قبل از اجرا متغیرها را ویرایش کن.
- backup_and_prepare.sql — روی Primary اجرا کن: خاموش کردن mirroring (در صورت وجود)، تنظیم FULL recovery، گرفتن full & log backup به یک UNC قابل دسترسی.
- restore_on_secondary.sql — روی Secondary اجرا کن: ریستور با NORECOVERY تا DB آماده اضافه شدن به AG شود.
- create_endpoints_and_ag.sql — ساخت endpoint (روی هر نود) و ایجاد Availability Group + Listener (روی Primary). مقادیر نام سرورها، DB و آدرس‌ها را قبل از اجرا ویرایش کن.
- listener_and_tests.sql — کوئری‌های بررسی وضعیت و نکات تست failover.
- filestream_notes.md — نکات حیاتی درباره FILESTREAM / FileTable (مسیرها، ACL‌ها و تست بعد از failover).
- enable_alwayson_note.txt — یادداشت گام‌به‌گام برای فعال کردن Always On از طریق SQL Server Configuration Manager (این مرحله را حتماً دستی انجام بده و سرویس را ری‌استارت کن).

نکات مهم و فوری:

1. قبل از اجرا همه‌ی فیلدهای CONFIG در فایل‌های SQL/PS1 را با مقادیر محیط خودت جایگزین کن (نام نودها، مسیر UNC، آدرس‌های IP، نام دیتابیس).
2. تایمینگ/پنجره نگهداری برای قطع mirroring و تبدیل به AG را مشخص کن — در این مرحله DB offline یا در NORECOVERY خواهد بود.
3. من پیشنهاد می‌کنم ابتدا این بسته را روی محیط تست (یا snapshot VM) اجرا کنی و پس از

تایید، به PROD بروی.

4. اگر می‌خواهی من این اسکریپت‌ها را بر اساس نام دقیق نودها، listener IP و لیست دیتابیس‌ها شخصی‌سازی کنم، فقط مقادیر را بده تا فایل‌ها را آماده و دوباره ارسال کنم.

می‌خواهی که:

- همین الان فایل‌ها را بر اساس نام نودها و IP‌های محیطت **خودکار تنظیم** کنم؟ (اگر بله، فقط نام نودها، Listener IP و مسیر بکاپ shared را بفرست)
 - یا یک **PDF مستند** از README هم تولید کنم و داخل زیپ بگذارم؟
- هر کدام رو بخوای انجام میدم — بگو کدوم رو می‌خوای تا فوراً آماده کنم.



اگر تعداد سرورها سه تا شود

باشه علیرضا جان — اگر تعداد سرورها سه‌تا شود، معماری و اجرای Always On Availability Groups به‌صورت زیر عوض می‌شود و قابلیت‌های بیشتری باز می‌شود. کامل و قدم‌به‌قدم برات توضیح می‌دهم.

وقتی تعداد سرورها 3 تا باشد (SQL Server Always On) با سه نود

سه سرور معمولاً یعنی:

- Node1 → Primary
- Node2 → Secondary (Sync, Automatic Failover)
- Node3 → Secondary (Sync, Manual Failover یا Async)

این مدل بهترین حالت High Availability + Disaster Recovery را بدون SAN فراهم می‌کند.

مزیت‌های سه‌سروره (Node AG-3)

✓ Failover خودکار واقعی بین دو نود اصلی

Primary ↔ Secondary (sync)

✓ یک نود جداگانه برای Disaster Recovery

Third node (async) → معماری 3-way protection

✓ Load Balancing برای Read-Only Queries

با سه نود، OLTP روی primary بماند
Reporting / Read-Only روی node2 یا node3

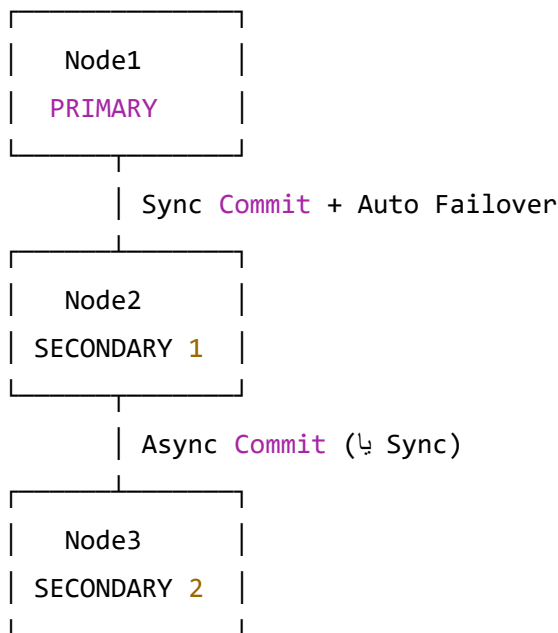
✓ Backup Offloading

Backup ها روی یکی از secondary ها اجرا می شود تا فشار از primary برداشته شود.

معماری پیشنهادی

Copy code

sql



نقش‌ها در WSFC (کلاستر ویندوز)

سه نود داخل یک کلاستر:

Quorum: Node Majority •

- دیگه File Share Witness لازم نیست (اگر بخواهی می‌توانی داشته باشی ولی ضروری نیست)

حالت‌های Failover

حالت‌های پیشنهادشده:

Failover Mode	Sync/Async	Node
—	—	Primary
Automatic Failover	Synchronous	Node2
Manual Failover	Sync یا Asynchronous	Node3

اگر FILESTREAM دارید → حتماً این نکته را رعایت کنید

- مسیر FILESTREAM روی هر سه نود باید یکسان باشد
مثال:

Copy code

makefile

```
D:\SQL\Data\FilestreamRoot\
```

- دسترسی‌ها و ACL‌ها یکسان باشند
- در صورت تعداد بالا، FileTable با سه نود هم درست کار می‌کند

چک‌لیست ایجاد AG با 3 سرور

1 ساخت WSFC با سه نود

Copy code

pgsql

```
New-Cluster -Name SQLCLUSTER -Node Node1,Node2,Node3 -StaticAddress 10.0.0.50
```

2 فعال کردن Always On روی هر سه نود

SQL Configuration Manager → Enable Always On → Restart Service

3 ساخت Endpoint روی هر سه سرور

پورت مثلاً 5022:

Copy code

sql

```
CREATE ENDPOINT Hadr_endpoint  
STATE=STARTED  
AS TCP (LISTENER_PORT = 5022)  
FOR DATA_MIRRORING (ROLE = ALL);
```

4 Backup → Restore روی دو Secondary

روی Primary بکاپ:

Copy code

sql

```
BACKUP DATABASE MyDB TO DISK='\\backup\MyDB.bak';  
BACKUP LOG MyDB TO DISK='\\backup\MyDB.trn';
```

روی Node2 و Node3 ریستور با NORECOVERY:

Copy code

sql

```
RESTORE DATABASE MyDB FROM DISK='\\backup\MyDB.bak' WITH NORECOVERY;  
RESTORE LOG MyDB FROM DISK='\\backup\MyDB.trn' WITH NORECOVERY;
```

5 ساخت AG با سه Replica

Copy code

sql

```
CREATE AVAILABILITY GROUP AG_PROD  
FOR DATABASE MyDB  
REPLICA ON  
N'Node1' WITH (  
    ENDPOINT_URL='TCP://Node1:5022',  
    AVAILABILITY_MODE=SYNCHRONOUS_COMMIT,  
    FAILOVER_MODE=AUTOMATIC  
),  
...
```

```

N'Node2' WITH (
    ENDPOINT_URL='TCP://Node2:5022',
    AVAILABILITY_MODE=SYNCHRONOUS_COMMIT,
    FAILOVER_MODE=AUTOMATIC
),
N'Node3' WITH (
    ENDPOINT_URL='TCP://Node3:5022',
    AVAILABILITY_MODE=ASYNCHRONOUS_COMMIT,
    FAILOVER_MODE=MANUAL
);

```

6 ساخت Listener

مثلاً:

Copy code

yaml

AG-LISTENER

IP: 10.0.0.60

Port: 1433

📌 حالت توصیه شده برای کاربردهای سنگین FILESTREAM

بهترین حالت:

- Node1 → Primary
- Node2 → Secondary Sync
- Node3 → Secondary Async (برای Load Balancing Read-Only)

اگر فایل‌ها خیلی سنگین هستند (مثلاً چند ترابایت):

- Node3 را Sync نکن چون سنگین می‌شود
- Node2 را Sync نگه‌دار که Failover سریع و امن انجام شود

📌 آیا اسکرپت سه سرور هم می‌خواهی؟

می‌توانم همین بسته زیپ را:

- برای 3 سرور بازنویسی کنم
- Auto-deployment کامل، بسازم

.....replica های سرور

- AG سه Replica با Listener و تنظیمات کامل تولید کنم
 - حتی یک فایل PDF مستند پیاده سازی حرفه ای هم کنارش بسازم
- کافی است بگویی:

♦ اسکریپت سه سروره را می خواهی؟

اگر بله →

اسم سه سرورت را بده (مثلاً: SQL1، SQL2، SQL3)
و بگو Listener چه اسمی باشد و چه IP بگیرد.
هر چیزی نیاز داری، من کامل آماده اش می کنم.



سوال فنی

وقتی یک درخواست

می رسد کدام سرور اول جواب می دهد و اگر بار زیاد باشد چطور

تقسیم با می شود

واگر یک سرور قطع شد چه میشود

علیرضا جان، این یکی از مهم ترین سؤال های Always On هست، و خیلی ها اشتباه برداشت می کنند.
جواب را کاملاً دقیق و ساده توضیح می دهم.

✓ وقتی یک درخواست (Query / Connection) می رسد، کدام سرور جواب می دهد؟

جواب کوتاه و قطعی:

✓ همیشه فقط و فقط Primary Replica درخواست های Read/Write را جواب می دهد

و هیچ تقسیم بار خودکاری بین Primary و Secondary ها انجام نمی شود
(مگر شما تنظیم کنید که Query ها Read-Only باشند، که پایین توضیح می دهم).

یعنی:

Copy code

sql

Application → Listener → Primary → پاسخ

Secondary ها فقط برای موارد زیر هستند:

- Failover در مواقع قطعی
- خواندن داده‌ها (در صورت فعال بودن Read-Only Routing)
- گرفتن Backup
- گزارش‌گیری

🔥 اما سؤال مهم: پس بار (Load) چطور تقسیم می‌شود؟

در Always On بار نوشتن (Write) هیچ‌وقت تقسیم نمی‌شود چون دیتابیس یک Primary واحد دارد.

اما بار خواندن (Read) را می‌توان بین Secondary ها تقسیم کرد.

🎯 دو روش تقسیم بار (Load Balancing) در Always On

(1) روش استاندارد: Read-Only Routing

اگر Application از Connection String به صورت درست استفاده کند (مثلاً .NET, EF Core):

Copy code

less

Application → Listener → Routing → Secondary #1 یا #2 یا #3

تنظیم معماری معمول:

- Primary → فقط Write
- Secondary 1 → Read-Only
- Secondary 2 → Read-Only
- Secondary 3 → DR (در صورت وجود)

✳ در این حالت بار با هوشمندی تقسیم می‌شود

هنگامی که Secondary از کلاً فقط Read-Only به رده منتقل می‌شود

و اگر یک Read-Intent Listener در سر بیستد، و Read-Intent Listener به پیچیده‌ترین سناریو.

Read-Intent Listener (2

این Listener فقط برای Read است.

مثال:

- Listener اصلی = برای نوشتن و کارهای معمول
- Listener دوم = ارسال کاربران Read به Secondaryها

🔥 اگر یکی از سرورها قطع شود چه اتفاقی می‌افتد؟

بستگی به نوع Replica دارد:

🟢 اگر Primary قطع شود

اگر حداقل یک Replica SYNC + Automatic Failover وجود داشته باشد
(مثلاً Node1 Primary → Node2 Sync):

✓ Automatic Failover اتفاق می‌افتد

Node2 → Primary جدید می‌شود
Listener → به صورت خودکار به Node2 وصل می‌شود
Application → بدون تغییر Connection String ادامه می‌دهد

🟠 اگر Secondary Sync قطع شود

Primary مشکلی ندارد و آنلاین می‌ماند
Failover خودکار غیرفعال می‌شود
(چون فقط یک Sync وجود دارد)

🟡 اگر Secondary Async قطع شود

هیچ اتفاقی نمی‌افتد

فقط همان Replica از چرخه Read یا Backup خارج می شود

☆ فلو کامل مدیریت قطع سرور

سرور	نوع	قطع شود چه می شود؟
Primary	Sync	Failover خودکار → Secondary Sync Primary جدید می شود
Secondary Sync	Sync	Primary کار می کند اما Automatic Failover غیرفعال می شود
Secondary Async	Async	فقط از رقابت خارج می شود، هیچ تاثیری روی Primary ندارد

🔥 بهترین معماری در 3 نود

Copy code

pgsql

Node1 → Primary (Write)

Node2 → Secondary Sync (Auto Failover)

Node3 → Secondary Async (Read Load + Backup)

🧪 مثال واقعی مسیر درخواست ها

درخواست Read/Write:

Copy code

sql

App → Listener → Primary → پاسخ

درخواست Read-Only (اگر تنظیم شده باشد):

Copy code

scss

App → Listener → Secondary2 یا Secondary3 → پاسخ (Load Balanced)

قطع Node1 (Primary):

Copy code

sql

Node2 → Primary می‌شود

Listener → تغییر مسیر می‌دهد

App → ادامه می‌دهد Down بدون

اگر بخواهی، می‌تونم برایت:

✓ Read-Only Routing برای سه نود تنظیم کنم

✓ Connection string مناسب برنامه‌ات را تولید کنم

✓ اسکریپت Load Balancing Read را آماده کنم

✓ Failover Testing Plan برای AG بنویسم

فقط بگو آیا تنظیم‌های Load Balancing را می‌خواهی؟

