

# SQL Server Availability Groups for Dummies



## Contact...



mark.broadbent@sqlcambs.org.uk



@retracement



tenbulls.co.uk

## Likes...



## Guilty pleasures...



## Badges...



Master: SQL Server



## Community...



## Tech editor...



# Agenda



In the beginning



Preparing Windows and SQL for availability Groups



Implementing Availability groups

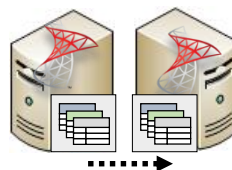


Readable secondaries and offloading backups

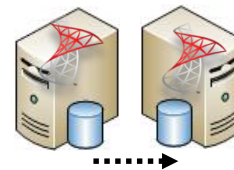
# In the beginning...



- 1995 – Replication ships with SQL Server 6.0
- 1997 – Windows NT introduces Wolfpack making SQL Clustering possible
- 2000 – Immediate updating subscribers with SQL Server 2000
- 2005 – Peer-to-Peer Transactional Replication added in SQL Server 2005
- 2005 – Database Mirroring added in SQL Server 2005



Replication



Database Mirroring

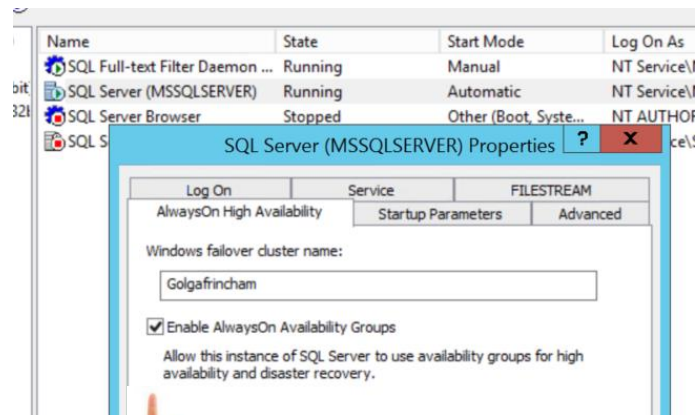


*Availability Groups are...*



# Preparing for Availability Groups

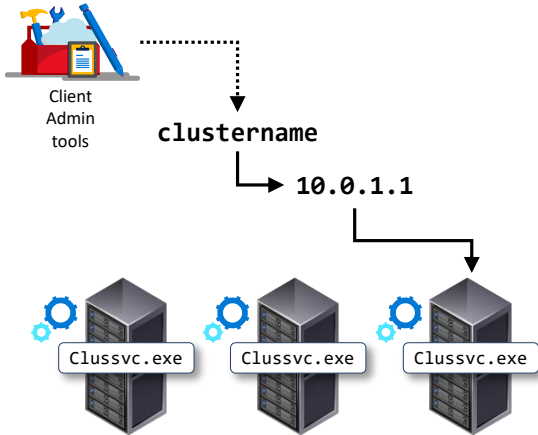
- Setup Windows Servers \*<sub>1</sub>
- Enable Failover-Clustering Feature in Windows
  - Add-WindowsFeature failover-clustering -IncludeAllSubFeature -IncludeManagementTools
  - Or install through GUI
- Create Windows Cluster across servers
- Install SQL Server on each Server \*<sub>1</sub>
- “Enable AlwaysOn Availability Groups” for SQL Service
- (optional) Configure Quorum/ node weighting
- Configure SQL Security



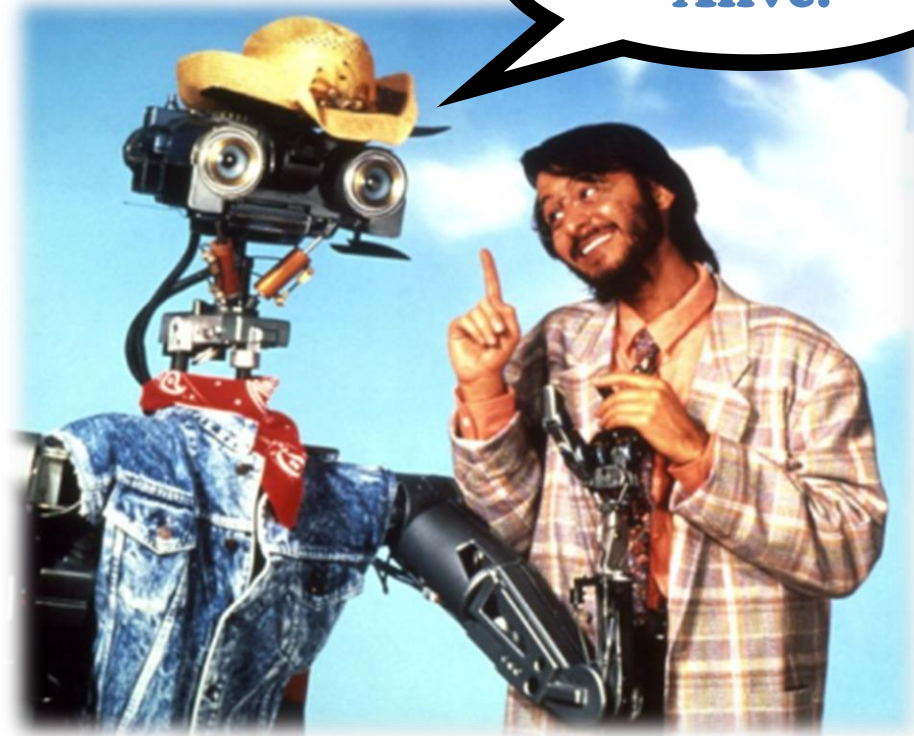
Warning! Enabling AlwaysOn  
requires service restart

*\*<sub>1</sub> If SQL Server is already installed, you do not need to do this step!*

# Windows Cluster & Quorum



- Is one or more nodes down?
  - Connect to node through name or '.'
  - Node weighting
- Is cluster alive (or is just cluster name down)?
  - Connect to Cluster IP
  - Use PowerShell
  - Resolve issues/ force quorum

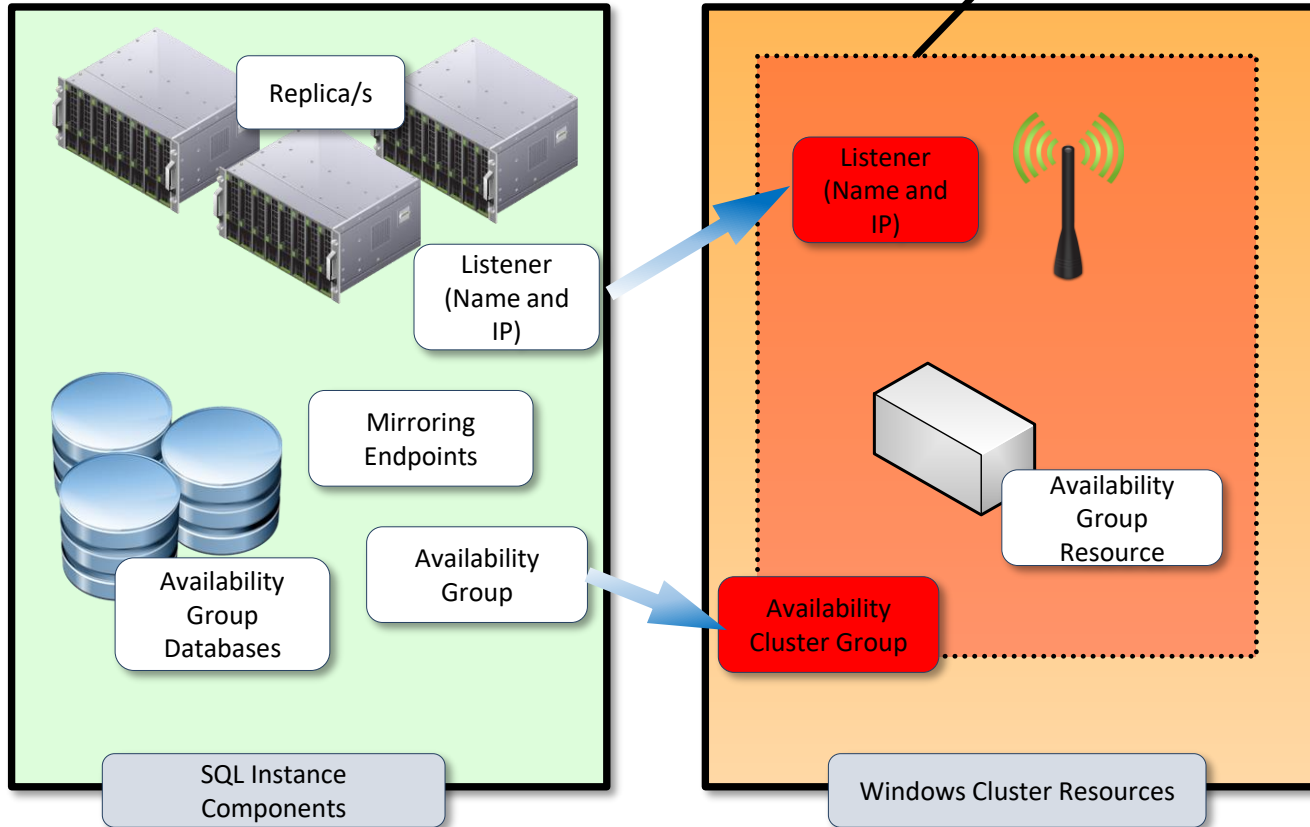


# Demo

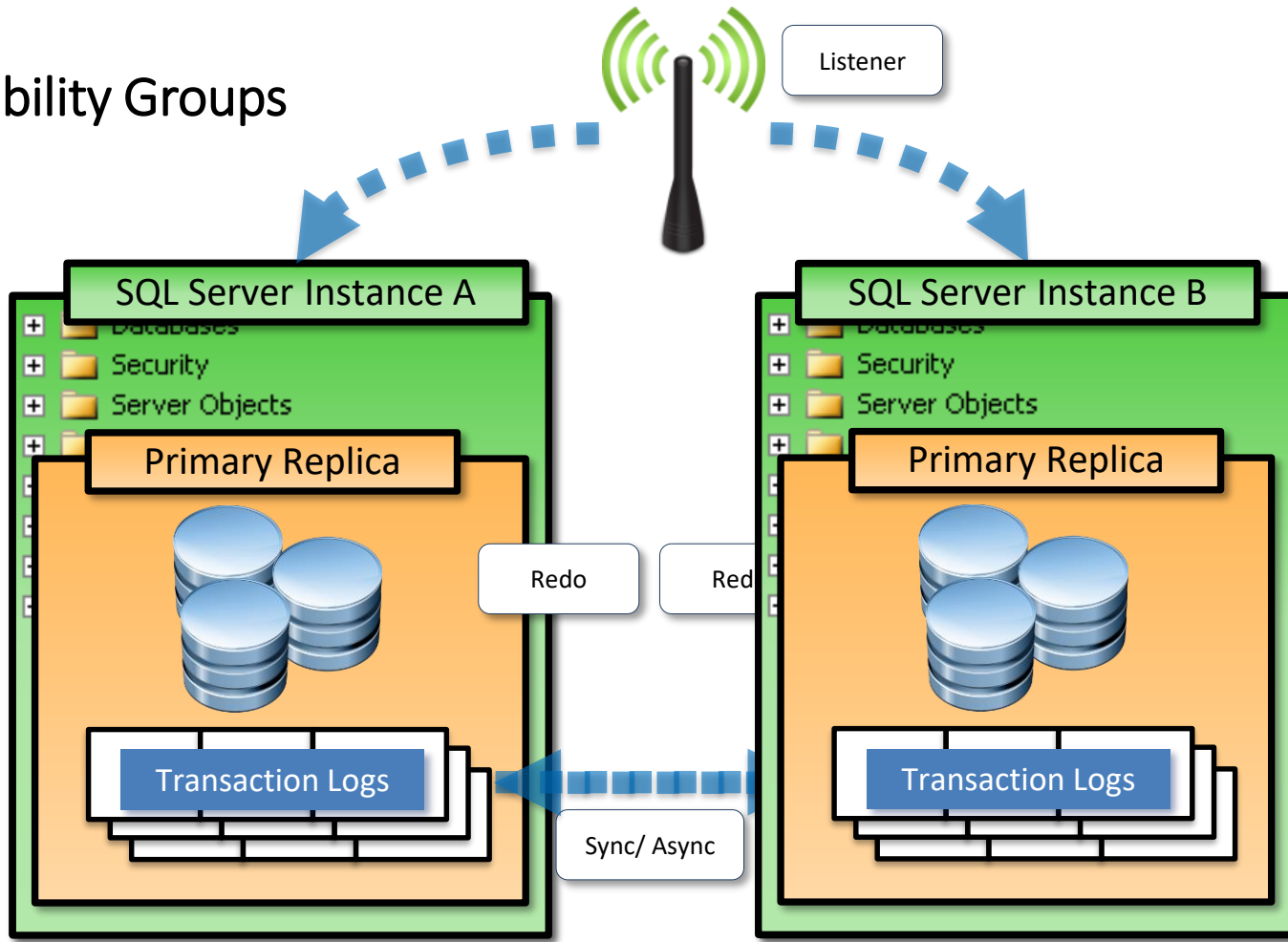
Cluster preparation, and SQL security



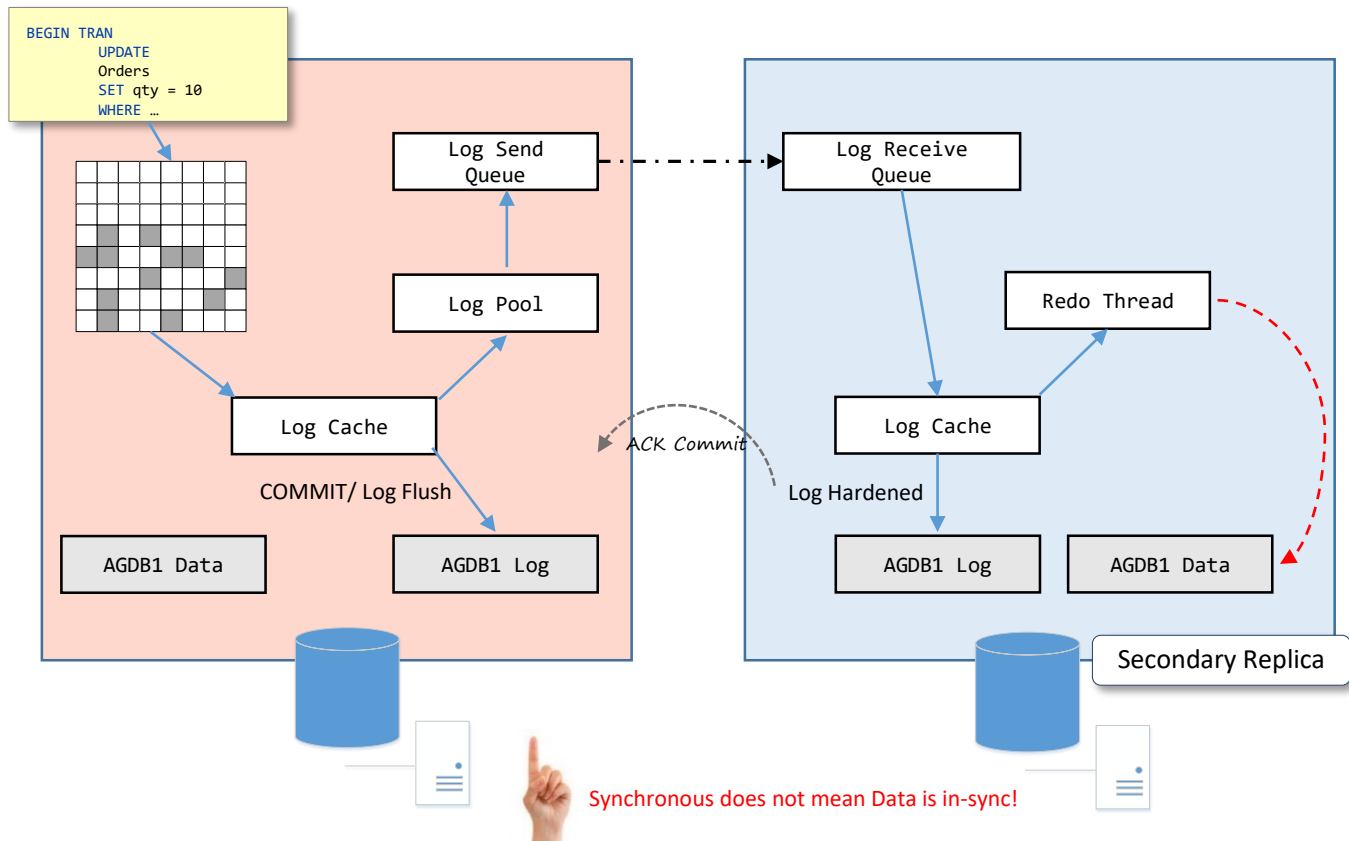
# Availability Group consists of...



# Availability Groups



# Availability Group DB Data Movement



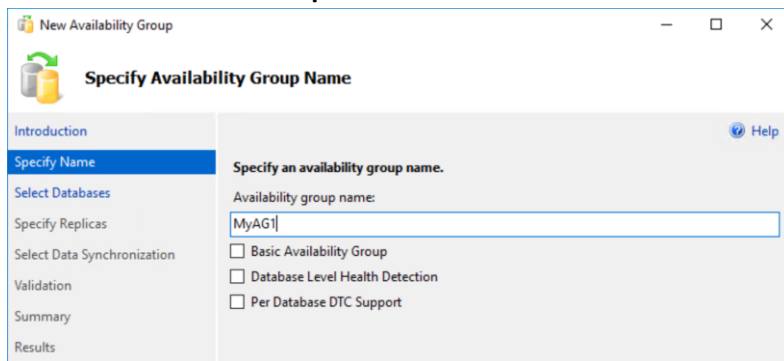
Synchronous does not mean Data is in-sync!

Feature	SQL 2012	SQL 2014	SQL 2016	SQL 2017	SQL 2019
Replicas	5 (max 2 sync with automatic failover, rest async)	9 (max 3 sync with 2 automatic failover, rest async)	9 (max 3 sync with automatic failover, rest async)	9 (max 3 sync with automatic failover, rest async)	Max 5 sync with automatic failover, rest async
Automatic Page Repair	YES	YES	YES	YES	YES
SSIS Catalog Support		Partial	Full	Full	Full
Replica Wizard		(on-premises) Replica Only	Azure Replica and (on-premises) Replica	Only via T-SQL was Removed in SSMS 17.0+	Only via T-SQL was Removed in SSMS 17.0+
Database level failover			YES	YES	YES
Basic Availability Groups			YES	YES	YES
MSDTC support			YES	YES	YES
Log transport optimizations			YES	YES	YES
Read-only routing load balancing			YES	YES	YES
gMSA support			YES	YES	YES
Linux Support				YES	YES
Read-scale AG				YES	YES
Availability Groups on K8s					YES (not RTM)



# Basic Availability Groups

- Standard Edition (or Enterprise)
- 2 replicas (configured on setup)
  - No upgrade to advanced AG
  - Cannot add or remove replicas
  - **No support for replicas prior to 2016**
  - Sync or Async
- No offloaded backups and no readable replica (snapshot possible but licensing needed!)
- 1 Availability Group with 1 Database replica



# Demo

Create and configure Availability Groups

# Readable Secondaries

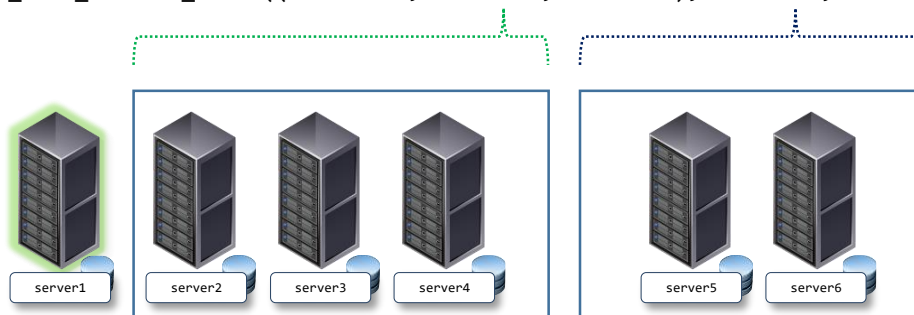
- Enabling Readable Secondary adds version pointer to each row (on change)
  - Query isolation level transparently mapped to SNAPSHOT
  - Risk of page splits and IO overhead
- Stale or missing statistics on primary cannot be auto-updated by query on secondary (since database is read only)
  - Auto-created in TempDB (must be explicitly dropped or instance restart) - most recent statistics in replica preferred
- DDL operations can block redo thread and reporting workloads!

Availability Replicas						
Server Instance	Role	Availability Mode	Failover Mode	Connections in Primary Role	Readable Secondary	Session Timeout (seconds)
SERVER4\SQL01	Secondary	Synchron...	Autom...	Allow all conne...	No	10
SERVER5\SQL01	Primary	Synchron...	Autom...	Allow all connecti...	No	10

# Read only routing –and load balancing

- Read only routing list must be configured on primary role (that listener will connect to)
- Secondary roles must be configured with Read-Only Routing url
- Applications must
  - Connect through Listener
  - Connection string must specify Application Intent=ReadOnly
- Parentheses identifies load balanced set

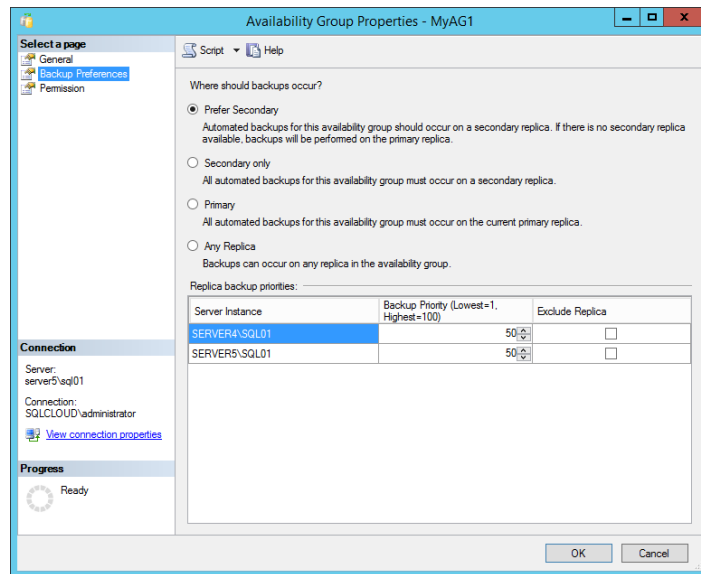
```
READ_ONLY_ROUTING_LIST=(( 'server2', 'server3', 'server4'), 'server5', 'server6')
```





# Backups on Database Secondary Replicas

- “Offload” backups to secondary replicas
  - Use *sys.fn\_hadr\_backup\_is\_preferred\_replica(<db\_name>)*
  - Secondary replicas must be in SYNCHRONIZED or SYNCHRONIZING state
  - Store to central location
- By default secondaries preferred
- Do not support Differential or Full without COPY\_ONLY



# Demo

Read-only routing and backups

## In Summary...

- Availability groups are not all Butterflies and Unicorns so choose an HA solution appropriate to your requirements.
- Failover Clustering is a very mature technology but requires specialist skill and understanding. It does not provide scalability. In the context of AGs it is relatively easy to manage
- Availability Groups partially provide reporting query scalability and arguably provide a level of DR – so AGs can be thought of as bringing HADR and Scale to a solution.

Thank you for listening!

Email: [contactme@sturmovik.net](mailto:contactme@sturmovik.net)

Twitter: [@retracement](https://twitter.com/retracement)

Blog: <http://tenbulls.co.uk>

Slideshare: <http://www.slideshare.net/retracement>

Demo: [https://github.com/retracement/availability\\_groups\\_for\\_dummies](https://github.com/retracement/availability_groups_for_dummies)

