Scaling - MODIFY FILE failed. Size is greater than MAXSIZE

Last updated by | Charlene Wang | Dec 5, 2022 at 1:55 AM PST

Issue:

CX is unable to scale down from P11 (1 TB limit) to P2 (500 GB limit) due to data file being over the limit.

Error:

ALTER DATABASE PMS2013 MODIFY (SERVICE_OBJECTIVE='P11', MAXSIZE=500GB) Msg 45316, Level 16, State 1, Line 1 MODIFY FILE failed. Size is greater than MAXSIZE. Please query sys.database_files and use DBCC SHRINKFILE to reduce the file size first.

Investigation/Analysis

Several customers have reported this in forums and mitigation has been to shrink file to get it below the limit. Examples of queries are Shrinkfile. However, there are times when this mitigation will not work (we have had many issues with shrinking data files on prem SQL Server). In these cases, the alternate approach is to work with customer to identify which table is contributing to the database growth.

Here is an example of Kusto query that you can use to find out when the database growth happened. While reviewing this incident, I see that data size went above the 500 GB mark ~11/24 @ 7 AM UTC. It has stayed at 726 GB since 11/24 @13:00 UTC to present. Can you check with the customer about specific activity that they did in this timeframe that could explain the database growth?

```
MonDmIoVirtualFileStats
| where TIMESTAMP >= ago(7d)
| where AppName == "b0c1587b9427" and NodeName == "DB.49"
| extend sizeondisk_gb = size_on_disk_bytes/(1024*1024*1024)
| project TIMESTAMP, type_desc ,sizeondisk_gb
| summarize max(sizeondisk_gb) by type_desc, bin(TIMESTAMP, 1h)
| render timechart
```

Mitigation:

Work with the customer to identify specific tables that they can delete data from to get the data file size under 500GB limit for P4.

To get a general idea of the database space consumption, you can rely on the following queries:

```
DECLARE @runtime datetime SET @runtime = GETUTCDATE()

SELECT CONVERT (varchar(30), DATEADD(hh,DATEDIFF(hh,0,@runtime),0), 121) as runtime,
DATEADD(hh,DATEDIFF(hh,0,@runtime),0) as runtime,
db_name() as 'db_name' ,o.name  as 'tbl_name',
o.create_date,
o.modify_date,
dps.index_id,
CONVERT(NUMERIC(10,2),
(dps.reserved_page_count)* 8.0/ 1024 ) AS 'Idx_SizeInMB',
dps.row_count AS 'Idx_RowCount'
FROM sys.objects o JOIN sys.dm_db_partition_stats dps
ON dps.object_id = o.object_id
where o.is_ms_shipped=0 and type='U'
ORDER BY CONVERT (varchar(30), @runtime, 121) ,db_name(), o.name
```

RCA (optional)

More Information (optional)

How good have you found this content?