

# Shrink fails with error 3140 Could not adjust the space allocation for file

Last updated by | Ricardo Marques | Mar 28, 2023 at 2:08 AM PDT

---

## Contents

- [Issue](#)
- [Investigation/Analysis](#)
- [Mitigation](#)
- [Internal reference](#)

## Issue

Customer running a shrink fails with error 3140 - *Could not adjust the space allocation for file.*

Customer tried to perform the shrink in different batch sizes, even small ones.

## Investigation/Analysis

Besides of normal T-SQL activity, a shrink process can also be competing with other processes like Backups.

If a backup is running, the shrink might fail with timeout (shrink will have to wait for the backup to finish)

To check the backup execution on telemetry.

```
MonBackup
| where LogicalServerName =~ "safr000prdazdb02"
| where logical_database_id =~ 'd8a6a7f6-8ab1-4fe0-9012-41089a3010ff'
| where TIMESTAMP > ago(7d) and isnotempty(backup_set_guid)
| where backup_type == 'Diff' //change this to Full if needed
| project backup_start_date, backup_end_date, backup_type, process_id, backup_set_guid,request_id,
first_lsn, last_lsn, logical_database_id, physical_database_id
| order by backup_start_date, backup_end_date
```

On customer side, if the backup is running:

```

SET NOCOUNT ON ;

SELECT      @@SERVERNAME Server,
            [SPID] = SESSION_ID ,
            percent_complete [%] ,
            [DATABASE] = DB_NAME(SP.DBID) ,
            [STATUS] = ER.STATUS ,
            [WAIT] = WAIT_TYPE ,
            wait_resource ,
            reads ,
            writes ,
            logical_reads ,
            command ,
            [INDIVIDUAL QUERY] = SUBSTRING(QT.TEXT,
                                            ER.STATEMENT_START_OFFSET
                                            / 2,
                                            ( CASE WHEN ER.STATEMENT_END_OFFSET = -1
                                                  THEN LEN(CONVERT(NVARCHAR(MAX), QT.TEXT))
                                                  * 2
                                                  ELSE ER.STATEMENT_END_OFFSET
                                                END
                                              - ER.STATEMENT_START_OFFSET )
                                            / 2) ,

            [QUERY] = QT.TEXT ,
            PROGRAM = PROGRAM_NAME ,
            [USER] = NT_USERNAME ,
            HOSTNAME ,
            NT_DOMAIN ,
            START_TIME-- ,
            --QP.query_plan AS xml_batch_query_plan
FROM      sys.dm_exec_requests ER WITH (NOLOCK)
          INNER JOIN sys.sysprocesses SP WITH (NOLOCK) ON ER.SESSION_ID = SP.SPID
          CROSS APPLY sys.dm_exec_sql_text(ER.SQL_HANDLE) AS QT
          -- CROSS APPLY sys.dm_exec_query_plan(ER.plan_handle) QP
WHERE     SESSION_ID > 50
          AND SESSION_ID NOT IN ( @@SPID )
          and command like 'BACKUP%'

ORDER BY  1 ,
          2

```

From customer side, but the backup is not running (historical view):

```

SELECT
    CONVERT(CHAR(100), SERVERPROPERTY('Servername')) AS Server,
    msdb.dbo.backupset.database_name,
    msdb.dbo.backupset.backup_start_date,
    msdb.dbo.backupset.backup_finish_date
FROM
    msdb.dbo.backupmediafamily
    INNER JOIN msdb.dbo.backupset ON msdb.dbo.backupmediafamily.media_set_id = msdb.dbo.backupset.media_set_id
WHERE (msdb..backupset.type = 'D' or msdb..backupset.type = 'I')
and (msdb.dbo.backupset.database_name = 'archetype_rimarqu' or msdb.dbo.backupset.database_name = '50c7b16d-9e
ORDER BY
    msdb.dbo.backupset.backup_start_date

```

## Mitigation

Run the shrink on a period where backups are not running.

## Internal reference

[365166349](#) 