

Normal Usage

The requested operation could not be completed due to a file system limitation (Win32Exception)

Ensure the image destination you chose has more free space than the size of your database, and ensure the disk is defragmented.

If you are running on Windows 7/Windows Server 2008 R2, this problem can also occur when creating images to a network path which is on the same machine as the agent creating the image. Upgrade to SQL Clone 0.7.6 or higher to fix this problem.

The files behind the Clone have grown to be very large

The differencing disk does not inflate except when there is a database write. Users have sometimes seen this grow very quickly when they had various automatic maintenance operations set up. For example, if you rebuild indexes, that will cause the entire rebuilt index to be in the diff disk.

The web app is stuck "loading" forever

On Internet Explorer, compatibility mode is automatically activated for unqualified host names. The solution is to disable compatibility mode, or use the fully qualified name for the server.

"The operating system returned error 21(The device is not ready.) ..."

This occurs when the network link between the clone and the image is broken. Unfortunately SQL Server tends to cache the error message. On SQL Server 2014 and above, cycling the database offline and online usually fixes the problem - SQL Clone periodically monitors for this condition and attempts this fix if necessary. Restarting the SQL Server process should always fix this issue. In the future we hope to improve SQL Clone's ability to recover from this state without restarting SQL Server.

"Pad block corrupted" after changing the user that SQL Clone Server runs as

If you wish to change the user running the SQL Clone Server, there's an encryption key stored securely which must be migrated. To do this you'll need two Powershell windows open. One running under the old credentials, and one running under the new credentials. To run Powershell as a different user, find it on the start menu, hold *shift*, and *right click* on the shortcut - "run as a different user" should be one of the options.

Migrate encryption key

```
# Open a powershell window on the Clone Server machine with the old account details
Add-Type -Path "C:\Program Files\Red Gate\SQL Clone Beta\RedGate.SqlClone.Credentials.dll"
$credentialLocker = new-object RedGate.SqlClone.Credentials.WindowsCredentialsLocker
$credentialLocker.GetCredential("SqlCloneDbKey") #Please manually copy this to the clipboard

# Open a powershell on the Clone Server machine with the new account details and paste the key copied above
$keyToMigrate = 'paste key in here'
Add-Type -Path "C:\Program Files\Red Gate\SQL Clone Beta\RedGate.SqlClone.Credentials.dll"
$credentialLocker = new-object RedGate.SqlClone.Credentials.WindowsCredentialsLocker
$credentialLocker.SetCredential("SqlCloneDbKey", $keyToMigrate)
```

Once the system is working again, you may optionally clear the key for the old account to further improve security.