



SharePoint 2013 Storage: Breaking the 200GB Barrier

By Nick Cavalancia

TABLE OF CONTENTS

The Problem with BLOBs	2
Making the Case for Remote BLOB Storage.....	3
RBS & SDS: Going Beyond Just BLOB Management	6
SharePoint Storage without Barriers.....	7

If you roll back the clock to when it was first released in 2001, we didn't have the same capacity demands as we experience today.

In the world of SharePoint, SharePoint's value comes from the utility gained by organizations like yours. And, often times, that utility comes in the form of the content that it holds. But, like every solution, there are limits – some hard limits you simply cannot surpass, and others that more impact SharePoint's ability to efficiently provide services.

One of those limitations revolves around the issue of BLOB storage. These unstructured pieces of data play a critical role in your SharePoint environment's ability (or inability, as the case may be) to perform, scale, and keep your users productive. And yet, because SharePoint runs on top of SQL Server (which was never designed to efficiently store, index, search for, and produce files as large pieces of data), for every BLOB that you store in SharePoint, the less efficient the environment becomes.

Now, storage wasn't always an issue with Sharepoint. If you roll back the clock to when it was first released in 2001, we didn't have the same capacity demands as we experience today. And certainly, another factor is simply what users used SharePoint for. Initially, sure, they found it useful as a central repository for lists, libraries, calendars and the like, as well as using it to host documents to collaborate on. But then came in the purpose-driven adoption where organizations no longer just used SharePoint as designed; they began to use it as a platform to meet their own custom needs, taking SharePoint in different directions, stretching the limits of what was possible. We all heard or seen examples of SharePoint being used as portals for channel partner programs, internal applications and even customer-facing websites.

And with that purpose-driven adoption came the need to host more than just basic information in a SQL database; SharePoint today now needs to host far richer content that was even available when it first released. But with the growth of seemingly cheaper and cheaper storage with greater and greater capacity – where you simply threw some additional drives on as the “answer” – it became the easiest option to solve even the SharePoint storage problem. Play the clock forward to today, and your organization is sitting with far too many BLOBs sitting in your SharePoint server's SQL database.

When a user first uploads a file, a new blob gets created within the content database.

So how do BLOBs impact SharePoint and what should you be doing about it?

In this whitepaper, we'll make an attempt to cover the storage and performance issues stemming from the presence of BLOBs in SharePoint, take a look at methods of moving BLOBs to remote storage, and when done properly, cover how properly managing SharePoint storage benefits more than just SharePoint.

The Problem with BLOBs

BLOBs are defined as unstructured data defined by its name and storage location – and your SharePoint environment is full of them. Sure, a Word doc has structure, but not in the database sense. To SQL Server, it's just a bunch of random binary data that its supposed to keep track of as a single data set. Microsoft's definition of a BLOB is any file over 256K in size. Think about that – how many files does your organization have up on SharePoint anywhere over that size? Conservative estimates dictate that BLOBs can consume as much as 95% of your SharePoint storage. Here's why: When a user first uploads a file, a new blob gets created within the content database. For every revision made to that file, when it's saved, SharePoint creates copy of the blob. In 2013, a feature called shredded storage improved storage by saving changes in a distributed set of BLOBs, each proportional to the size of the change made to a document stored in SharePoint. But even with this improvement, when you take into consideration the number of files stored there, and the number of revisions to each, it becomes easy to see how it can all add up.

So you have lots of BLOBs - what's the issue then?

SQL Server, in general requires high IOPS (input/output operations per second) and low latency, which usually translates to expensive, fast storage. But when you add in the need to store BLOBs, which are demonstrably larger than, say, a phone number in a database table, most organizations aren't investing in high IOPS/low latency storage enough to house all their BLOBs.

And then there's Microsoft's recommendation that the content database be no larger than 200GB. Now, it's not a hard ceiling; it's merely a recommendation. But it's there because once you hit that content database size, the expectation should be that you'll see a

Think ahead 5 years and take a guess on how much more data you'll have on your SharePoint servers than today.

degradation in performance. The ceiling is there because Microsoft knows other services will start to degrade such as native SharePoint backups. Once you hit the 200GB limit, it will take much longer to backup a database, shifting the time it takes to backup from hours of time to days. Indexing can also take a lot longer; SQL Server is meant to handle lots of small data. But, when you bring BLOBs into the picture, the time it takes to process that data from the database increases significantly.

It's assumed that you're reading this whitepaper because you're either near hitting the content database ceiling, or because you're experiencing performance issues and found out about the ceiling as part of your research. In either case, you've probably come to realize hitting the ceiling comes pretty quickly - audio, video, and graphics files, for starters, are all monsters in size, let alone Word docs with those same graphics inside. From an SQL Server standpoint, in some ways, those file just don't belong in there.

Then, is this storage issue an obstacle or an opportunity?

There's good and bad news in all of this. The good news is even with the recommended 200GB limit, you do have options to get you well past that number. The bad news is this isn't a static problem, where once you address it today, it goes away forever. The reason? You're simply addressing the storage needs of your organization today. Think ahead 5 years and take a guess on how much more data you'll have on your SharePoint servers than today. 2x? 3x? 5x? Even 7x? One thing is certain - data growth is only going one direction. So, you need to be proactive and think about solving the problem long-term.

So, how do you get past the 200GB limit without impacting performance?

Making the Case for Remote BLOB Storage

As mentioned previously, the 200GB content database size isn't really a true limit - as in SharePoint will cease to function when you go over. You should be thinking of it as more the start of a conversation about how to properly plan your SharePoint storage than the issue of the current content database size.

When you choose to go the RBS route, your focus needs to be on organizing where the data should be stored.

How, then, do you grow your SharePoint environment storage past 200GB?

The answer lies in a technology that's been around since 2007 and in its current iteration since SQL Server 2008 – Remote BLOB Storage (RBS). At its simplest, you can think of RBS as a means to offload BLOB data to a file server, while providing seamless access to that remote data to SharePoint users.

But it's more than just about offloading large files. And it's definitely not just about adding more disks. When you choose to go the RBS route, your focus needs to be on organizing where the data should be stored. Think about it – you have content you need to collaborate on today, so that should be on very fast storage like flash-based drives. But, say, 90 days out from today, those files aren't as important, so they could be moved to slower storage, such as traditional spinning disks. And then, in a few quarters or a year from now, you may want it all archived.

When it comes down to it, you should be thinking about your SharePoint storage from a few perspectives:

- **Capacity** – Sure, the number today is 200GB, but you need to be thinking about capacity on an ongoing basis as the size of data stored grows and as the organization's needs change.
- **Scalability** – This is more about how quickly can you get to the higher capacity, as well as what's the right way to do it. It's not just about adding more drives. It's about an ability to both increase storage and have an ability to properly (and quickly) identify which BLOBS need to reside there.
- **Performance** – You can't just add on a filer and call it done. While it will initially help, the reality is you need to be thinking about software-defined storage to create tiers of storage and an intelligent way to place BLOBS on the various tiers.

RBS out of the box provides the most basic of functionality. Using only file size minimum as the criteria, BLOBS can be moved to a single storage location. There's no real granular control to either define which data

does and doesn't need RBS, nor where that data should be located (other than a single remote storage location).

Then how can RBS help with performance, as well as storage?

RBS: Better with Software-Defined Storage

The move to remote storage alone will help speed up your SQL server – indices are generated more quickly, searches are faster, retrieval of data should be faster. But you can't just stop there. The "where" previously mentioned as in "where should the data be stored" isn't just about whether BLOBs should be in an SQL database somewhere or not; it's about that issue of having faster and slower tiers of storage.

A software-defined storage (SDS) infrastructure would certainly help here, by establishing those tiers of storage – from faster to slower – as not every file needs to be offloaded to the same tier. It would allow that more frequently used content to reside on faster storage, and less used content on slower storage. As organizational needs grow, storage can be added on the fly, paying as you grow, without needing to update pointers and stubs within SQL Server.

While SDS can definitely help today, it also provides the flexibility needed for tomorrow. By taking an inventory of how much content exists within SharePoint and looking at storage trends around how quickly is data being added, you can calculate a projected rate of storage growth per month. So, say, you want to budget for what will you need in 5 years – SDS helps you prepare to restructure storage to accommodate that level of growth.

Do keep in mind that even with all of the benefits SDS brings to the table, the out of the box RBS lacks the management granularity to establish policy, as well as the intelligence and automation necessary to have files moved from one storage location to another without IT having to touch the file. So if you want to take advantage of the performance and scalability gains SDS can provide, you're going to need to look at a third-party RBS solution.

If you want to take advantage of the performance and scalability gains SDS can provide, you're going to need to look at a third-party RBS solution.

The expense of implementing both RBS and SDS needs to go beyond just helping you get passed the immediate database limitations.

RBS & SDS: Going Beyond Just BLOB Management

Some of you who may be either approaching the 200GB barrier, or have just passed it, may be looking for more reasons than to just offload BLOBs from SQL Server. That's good – it means that moving storage alone isn't enough. The expense of implementing both RBS and SDS needs to go beyond just helping you get passed the immediate database limitations. The good news is there are actually a number of other reasons for implementing RBS with SDS.

- **Meeting SLAs** – There are two ways to look at meeting agreed upon service levels. The obvious availability of SharePoint as a whole immediately comes to mind. By offloading storage of BLOBs to an external SDS infrastructure, you potentially eliminate single points of failure within a single SQL Server, thereby helping to maintain service availability. And by using SDS, which utilizes its own storage redundancy, you ensure availability of BLOBs. Additionally, the right speed storage can ensure that you meet delivery service levels for the first pieces of data. Some organizations look for delivery in as little as tens of milliseconds.
- **Managing Storage and Performance** – Putting RBS and SDS together helps find the balance between these two. Just adding a solid RBS solution will help increase performance initially (by moving BLOBs out of the content database), but do little in the long run. Putting in SDS in place equally only helps short term if there is no management of the BLOBs over time to increase the performance in accessing current needed BLOBs over old, unused one. It's the combination of the two that optimizes RBS's use of storage and the performance to deliver BLOBs quickly, as well as to share infrastructure with other applications.
- **Seamlessly Scaling SharePoint** – Think about this from a user experience perspective. As your storage needs grow, RBS alone would provide a seamless, more productive way to access BLOBs. But in the long run, you want to be able to plug in storage on the fly. So if you have 200TB today, every time you need to add another 100TB of storage, the SDS should auto balance the allocation and usage of storage, with users blissfully unaware of the addition.

By putting data into a database where it doesn't exactly belong, eventually you're going to have a performance issue.

- **Protecting SharePoint Data** – Backups of SharePoint will run much faster when content has been offloaded, as only a few pieces of metadata and the stub need to be backed up. You'll obviously need to backup the file shares separately, but 3rd party solutions can make sure the two backup sets are in sync.
- **Meeting Archive, Compliance, and Governance Requirements** – While you're probably thinking about examples like patient or personal information stored in documents, the concept of meeting these requirements is really about elevating the conversation from just managing SharePoint content, to thinking about managing the lifecycle of your SharePoint content. With an RBS solution that can identify the types of content to be moved off, establishing policies to define what needs to be maintained, and an SDS infrastructure providing varying tiers of storage to meet archive, compliance, and governance needs, you'll be able to find a balance of meeting these requirements, while still making content accessible in a timely fashion.

It's a no-brainer. Putting these two technologies together help you achieve not just improvements around storage and performance, but also to achieve organizational goals around scalability, data protection, archiving, and compliance with regard to SharePoint.

SharePoint Storage without Barriers

It's inevitable that, by putting data into a database where it doesn't exactly belong, eventually you're going to have a performance issue. You'll buy yourself some time by moving it onto a storage medium that better handles large data sets. But you still should expect that, at some point, even with offloaded storage, performance will degrade.

To put your SharePoint environment in a position where storage size and performance simply aren't an issue, you're going to need to take a proactive and intelligently designed approach utilizing a RBS implementation that allows you to identify the content that is, and isn't, important to your organization, and marry it with a storage environment that creates pools and tiers of storage that parallel speed of access to the content's importance. By properly implementing RBS and SDS together, you create a SharePoint environment with a

limitless ability to grow without the performance hits normally associated with increasing storage, while simultaneously improving SharePoint's capacity to meet the organization's needs, whether it's 5 weeks, months, or years from now. ■

With nearly 20 years of enterprise IT experience, Nick Cavalancia is an accomplished consultant, speaker, trainer, writer, and columnist and has achieved certifications including MCSE, MCT, MCNE and MCNI. He has authored, co-authored and contributed to over a dozen books on Windows, Active Directory, Exchange and other Microsoft technologies. He has spoken at conferences such as the Microsoft Exchange Conference, TechEd, Exchange Connections, and on countless webinars and at tradeshow around the world.
