

مطلبی چندی قبل در مورد " [ذخیره سازی فایل‌ها در دیتابیس یا استفاده از فایل سیستم متداول؟](#) " منتشر گردید، جهت برشمردن فواید ذخیره سازی فایل‌ها در دیتابیس ([+](#))، اما معایب این نوع ذخیره سازی بررسی نشدند:

- الف) اختصاص یافتن قسمتی از بافر SQL Server به این امر.
- ب) با توجه به قرار گرفتن داده‌های BLOB در دیتابیس، transaction log قابل توجهی تولید خواهد شد. ([+](#))
- ج) بیش از 2GB را نمی‌توان در فیلدهایی از نوع varbinary(max) ذخیره کرد.
- د) به روز رسانی BLOB ها سبب ایجاد fragmentation می‌شود.

مایکروسافت برای رفع این مشکلات در SQL Server 2008 قابلیت جدیدی را ارائه داده است به نام FileStream که در طی مقالاتی به بررسی آن خواهیم پرداخت.

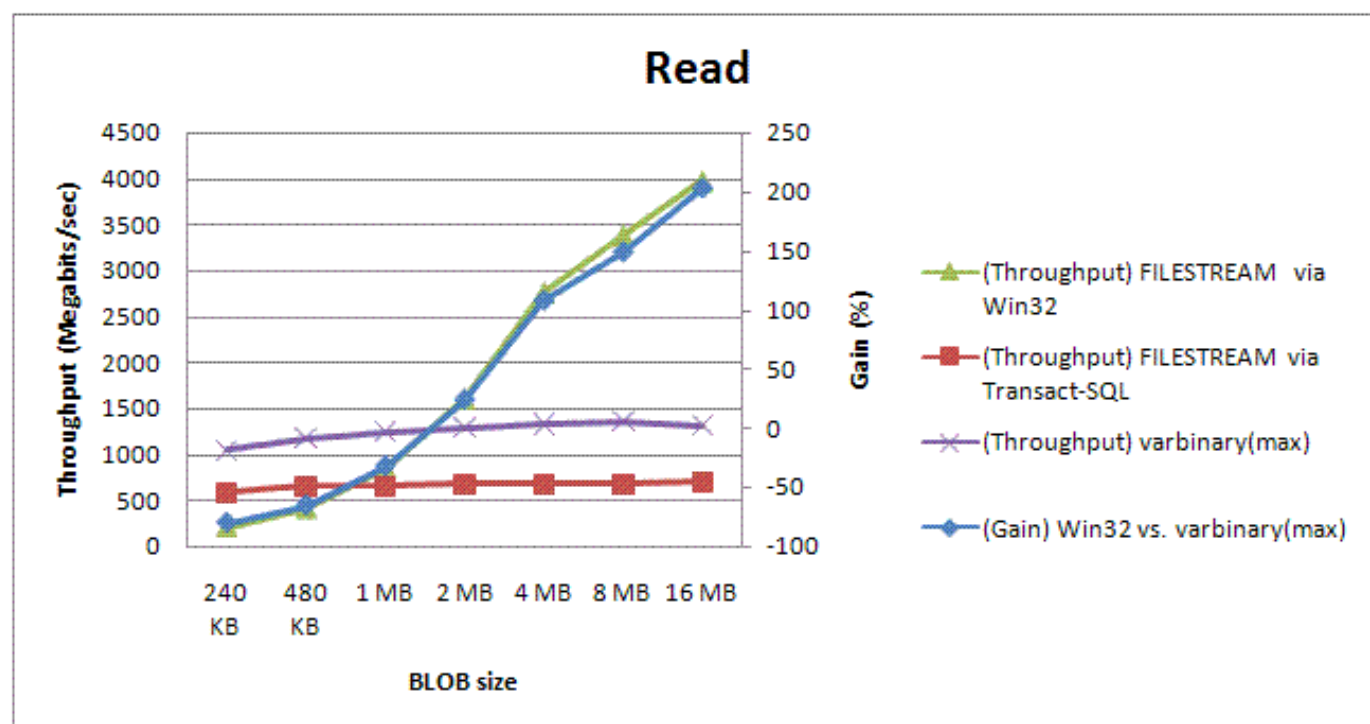
FILESTREAM موتور دیتابیس اس کیوال سرور را با سیستم فایل NTFS یکپارچه می‌کند؛ به این صورت که داده‌های BLOB از نوع varbinary(max) را به صورت فایل بر روی سیستم ذخیره خواهد کرد. سپس با استفاده از دستورات T-SQL می‌توان این فایل‌ها را ثبت، حذف، به روز رسانی، جستجو و بک آپ گیری کرد. این قابلیت نیز از فیلدهای varbinary(max) استفاده می‌کند؛ اما اکنون ویژگی و برچسب FILESTREAM به این نوع فیلدها الصاق خواهد شد. FILESTREAM data باید در FILESTREAM filegroups ذخیره شوند. FILESTREAM filegroups در حقیقت همان پوشه‌های فایل سیستم می‌باشند. به آن‌ها data containers نیز گفته می‌شوند که مرزی هستند بین ذخیره سازی داده‌ها در فایل سیستم و در دیتابیس.

مزایای سیستم FileStream چیست؟

- الف) سیستم transaction مختص به خود را داشته، به همین جهت سبب رشد غیر منطقی حجم فایل transaction log دیتابیس اصلی نمی‌شوند.
- ب) هنگام به روز رسانی فیلدهایی از این دست، صرفاً ایجاد یا حذف یک فایل مد نظر است؛ بنابراین fragmentation ایجاد شده در این حالت بسیار کمتر از روش استفاده از فیلدهایی از نوع varbinary(max) می‌باشد.
- ج) استفاده از NT system cache جهت کش کردن اطلاعات که سبب بالا بردن بازدهی بانک اطلاعاتی خواهد شد.
- د) از buffer pool اس کیوال سرور در این حالت استفاده نشده (مطابق قسمت ج) و این حافظه جهت امور روزمره‌ی اس کیوال سرور کاملاً مهیا خواهد بود.
- ه) محدودیت 2GB فیلدهایی از نوع varbinary(max) با توجه به ذخیره سازی این نوع BLOBs در فایل سیستم، دیگر وجود نخواهد داشت.

چه زمانی بهتر است از FileStream استفاده شود؟

- الف) فایل‌هایی که ذخیره می‌شوند به طور متوسط بیش از یک مگابایت حجم داشته باشند. (برای کمتر از این مقدار BLOBs varbinary(max) کارایی بهتری را ارائه می‌دهند). هر چند این مرز یک مگابایت مطابق اطلاعات books online است اما تجربیات کاری نشان می‌دهند که این سقف را باید 256 کیلوبایت در نظر گرفت.
- ب) قابلیت خواندن سریع اطلاعات فایل‌ها مد نظر باشد (بررسی کارایی مطابق تصویر زیر از MSDN). سیستم NTFS نسبت به SQL Server در خواندن فایل‌های حجیم سریعتر عمل می‌کند.
- ج) اگر از یک معماری middle tier در برنامه‌های خود در حال استفاده‌اید.
- د) زمانی که نیاز باشد تا اطلاعات relational و non-relational در یک تراکنش مورد استفاده قرار گیرند.



نکاتی را که باید هنگام ذخیره سازی اطلاعات در FileStream در نظر داشت

الف) هنگامی که یک جدول حاوی فیلدی از نوع FileStream می باشد، باید دارای فیلد ID منحصر بفرد نیز باشد.

ب) data containers ای که پیش از این در مورد آنها صحبت شد، نباید تو در تو باشند.

ج) FILESTREAM filegroups بر روی درایوهای فشرده شده نیز می توانند قرار داشته باشند.

FileStream از دیدگاه امنیت

امنیت داده های FileStream در اس کیوال سرور دقیقاً همانند امنیت سایر اطلاعات ذخیره شده در دیتابیس است (دسترسی در حد جدول و یا فیلد). اگر کاربری دسترسی به فیلد FileStream در یک جدول داشته باشد، می تواند آن فایل را گشوده و استفاده کند. رمزنگاری بر روی این ستون ها پشتیبانی نمی شود. تنها اکانتی که اس کیوال سرور تحت آن در حال اجرا است دسترسی به FILESTREAM container دارد. همچنین توصیه شده است که به هیچ اکانت دیگری این دسترسی داده نشود. زمانیکه یک دیتابیس آغاز و مشغول به کار می شود، اس کیوال سرور دسترسی به FILESTREAM data container را محدود خواهد کرد و دسترسی به این اطلاعات تنها از طریق دستورات T-SQL و یا OpenSqlFileStream API میسر خواهد بود. بدیهی است زمانیکه اس کیوال سرور متوقف شود، این اطلاعات بدون هیچگونه محدودیتی قابل دسترسی بوده و تنها محدودیت های سیستمی به آنها اعمال خواهند شد (که این مورد باید مد نظر باشد).

نگهداری FileStream

FileStream به صورت فیلدهای varbinary(max) یکپارچه با دیتابیس ذخیره می شود؛ بنابراین نحوه ی تهیه پشتیبان از آنها همانند روش های متداول است بدون هیچگونه تغییری (و این اطلاعات در بک آپ دیتابیس لحاظ می شوند). اگر نیاز بود هنگام تهیه پشتیبان از این نوع داده ها بک آپ گرفته نشود، می توان از partial backup با پارامترهای مربوطه استفاده کرد.

ادامه دارد ...

نظرات خوانندگان

نویسنده: fahimi

تاریخ: ۲۲:۲ ۱۳۹۱/۰۶/۰۷

سلام

از آموزش بسیار عالی و کاربردی جنابعالی کمال تشکر را دارم
من آموزش شما را در delphi & C# پیاده کردم و به خوبی اجرا شد
سوال: آیا دستورات Update , Delete
علاوه بر تیبیل فایل‌ها را هم تاثیر دار
به هر حال من دستور delete را اجرا کردم در تیبیل رکورد حذف شد ولی فایل مرتبط با رکورد حذف نشد
با تشکر

نویسنده: وحید نصیری

تاریخ: ۲۲:۱۰ ۱۳۹۱/۰۶/۰۷

همین‌طور. این فایل‌ها سر فرصت توسط سیستم [FILESTREAM garbage collection](#) حذف خواهند شد.

نویسنده: حسین مرادی نیا

تاریخ: ۲۲:۵۷ ۱۳۹۱/۰۶/۰۷

سلام؛

اگر از Entity Framework در حالت CodeFirst استفاده نمایم میتوان از قابلیت‌های FileStream در آن استفاده کرد؟

نویسنده: وحید نصیری

تاریخ: ۲۳:۴۴ ۱۳۹۱/۰۶/۰۷

بله. خاصیت مورد نظر را به صورت byte array تعریف کنید:

```
public class Image
{
    public int Id { get; set; }
    public byte[] Data { get; set; }
}
```