

روشی را که مایکروسافت برای پرداختن به مقوله NoSQL تاکنون انتخاب کرده است، قرار دادن ویژگی‌هایی خاصی از دنیای NoSQL مانند امکان تعریف اسکیمای متغیر، داخل مهم‌ترین بانک اطلاعاتی رابطه‌ای آن، یعنی SQL Server است، که در ادامه به آن خواهیم پرداخت. همچنین در سمت محصولات پردازش ابری آن نیز امکان دسترسی به محصولات NoSQL کاملی وجود دارد.

Azure table storage (1)

Azure table storage در حقیقت یک Key-value store ابری است و برای کار با آن از اینترفیس پروتکل استاندارد OData استفاده می‌شود. علت استفاده و طراحی یک سیستم Key-value store در اینجا، مناسب بودن اینگونه سیستم‌ها جهت مقاصد عمومی است و به این ترتیب می‌توان به بازه بیشتری از مصرف کنندگان، خدمات ارائه داد. پیش از ارائه Azure table storage، مایکروسافت سرویس خاصی را به نام SQL Server Data Services که به آن SQL Azure نیز گفته می‌شود، معرفی کرد. این سرویس نیز یک Key-Value store است؛ هرچند از SQL Server به عنوان مخزن نگهداری اطلاعات آن استفاده می‌کند.

SQL Azure XML Columns (2)

فیلدهای XML از سال 2005 به امکانات توکار SQL Server اضافه شدند و این نوع فیلدها، بسیاری از مزایای دنیای NoSQL را درون SQL Server رابطه‌ای مهیا می‌سازند. برای مثال با تعریف یک فیلد به صورت XML، می‌توان از هر ردیف به ردیفی دیگر، اطلاعات متفاوتی را ذخیره کرد؛ به این ترتیب امکان کار با یک فیلد که می‌تواند اطلاعات یک شیء را قبول کند و در حقیقت امکان تعریف اسکیمای پویا و متغیر را در کنار امکانات یک بانک اطلاعاتی رابطه‌ای که از اسکیمای ثابت پشتیبانی می‌کند، میسر می‌شود. در این حالت در هر ردیف می‌توان تعدادی ستون ثابت را با یک ستون XML با اسکیمای کاملاً پویا ترکیب کرد. همچنین SQL Server در این حالت قابلیت را ارائه می‌دهد که در بسیاری از بانک‌های اطلاعاتی NoSQL میسر نیست. در اینجا در صورت نیاز و لزوم می‌توان اسکیمای کاملاً مشخصی را به یک فیلد XML نیز انتساب داد؛ هر چند این مورد اختیاری است و می‌توان یک un typed XML را نیز بکار برد. به علاوه امکانات کوئری گرفتن توکار از این اطلاعات را به کمک XPath ترکیب شده با T-SQL، نیز فراموش نکنید.

بنابراین اگر یکی از اهداف اصلی گرایش شما به سمت دنیای NoSQL، استفاده از امکان تعریف اطلاعاتی با اسکیمای متغیر و پویا است، فیلدهای نوع XML اس کیوال سرور را مدنظر داشته باشید. یک مثال عملی: فناوری Azure Dev Fabric's Table Storage (نسخه Developer ویندوز Azure که روی ویندوزهای معمولی اجرا می‌شود؛ یک شبیه ساز خانگی) به کمک SQL Server و فیلدهای XML آن طراحی شده است.

SQL Azure Federations (3)

در اینجا منظور از Federations در حقیقت همان پیاده سازی قابلیت Sharding بانک‌های اطلاعاتی NoSQL توسط SQL Azure است که برای توزیع اطلاعات بر روی سرورهای مختلف طراحی شده است. به این ترتیب دو قابلیت Partitioning و همچنین Replication به صورت خودکار در دسترس خواهند بود. هر Partition در اینجا، یک SQL Azure کامل است. بنابراین چندین بانک اطلاعاتی فیزیکی، یک بانک اطلاعاتی کلی را تشکیل خواهند داد.

هرچند در اینجا Sharding (که به آن Federation member گفته می‌شود) و در پی آن مفهوم «عاقبت یک دست شدن اطلاعات» وجود دارد، اما درون یک Shard یا یک Federation member، مفهوم ACID پیاده سازی شده است. از این جهت که هر Shard واقعا یک بانک اطلاعاتی رابطه‌ای است. اینجا است که مفهوم برنامه‌های Multi-tenancy را برای درک آن باید در نظر داشت. برای نمونه یک برنامه وب را در نظر بگیرید که قسمت اصلی اطلاعات کاربران آن بر روی یک Shard قرار دارد و سایر اطلاعات بر روی سایر Shards پراکنده شده‌اند. در این حالت است که یک برنامه وب با وجود مفهوم ACID در یک Shard می‌تواند سریع پاسخ دهد که آیا کاربری پیشتر در سایت ثبت نام کرده است یا خیر و از ثبت نام‌های غیرمجاز جلوگیری به عمل آورد.

در اینجا تنها موردی که پشتیبانی نشده است، کوئری‌های Fan-out می‌باشد که [پیشتر](#) در مورد آن بحث شد. از این جهت که با نحوه خاصی که Sharding آن طراحی شده است، نیازی به تهیه کوئری‌هایی که به صورت موازی بر روی کلیه Shards برای جمع آوری اطلاعات اجرا می‌شوند، نیست. هر چند از هر shard با استفاده از برنامه‌های دات نت، می‌توان به صورت جداگانه نیز کوئری

گرفت.

4 OData

اگر به CouchDB و امکان دسترسی به امکانات آن از طریق وب دقت کنید، در محصولات مایکروسافت نیز این دسترسی REST API پیاده سازی شده‌اند.

OData یک RESTful API است برای دسترسی به اطلاعاتی که به شکل XML یا JSON بازگشت داده می‌شوند. انواع و اقسام کلاینت‌هایی برای کار با آن از جاوا اسکریپت گرفته تا سیستم‌های موبایل، دات نت و جاوا، وجود دارند. از این API نه فقط برای خواندن اطلاعات، بلکه برای ثبت و به روز رسانی داده‌ها نیز استفاده می‌شود. در سیستم‌های جاری مایکروسافت، بسیاری از فناوری‌ها، اطلاعات خود را به صورت OData در اختیار مصرف کنندگان قرار می‌دهند مانند Azure table storage، کار با SQL Azure از طریق WCF Data Services (جایی که OData از آن نشأت گرفته شده)، Azure Data Market (برای ارائه فیدهایی از اطلاعات خصوصاً رایگان)، ابزارهای گزارش‌گیری مانند SQL Server reporting services، لیست‌های شیرپوینت و غیره. به این ترتیب به بسیاری از قابلیت‌های دنیای NoSQL مانند کار با اطلاعات JSON بدون ترک دنیای رابطه‌ای می‌توان دسترسی داشت.

5 امکان اجرای MongoDB و امثال آن روی سکوی کاری Azure

امکان توزیع MongoDB بر روی یک Worker role سکوی کاری Azure وجود دارد. در این حالت بانک‌های اطلاعاتی این سیستم‌ها بر روی Azure Blob Storage قرار می‌گیرند که به آن‌ها Azure drive نیز گفته می‌شود. همین روش برای سایر بانک‌های اطلاعاتی NoSQL نیز قابل اجرا است. به علاوه امکان اجرای Hadoop نیز بر روی Azure وجود دارد. مایکروسافت به کمک شرکتی به نام HortonWorks نسخه ویندوزی Hadoop را توسعه داده‌اند. HortonWorks را افرادی تشکیل داده‌اند که پیشتر در شرکت یاهو بر روی پروژه Hadoop کار می‌کرده‌اند.

6 قابلیت‌های فرا رابطه‌ای SQL Server

الف) فیلدهای XML (که در ابتدای این مطلب به آن پرداخته شد). به این ترتیب می‌توان به یک اسکیمای انعطاف پذیر، بدون از دست دادن ضمانت ACID رسید.

ب) فیلد HierarchyId برای ذخیره سازی اطلاعات چند سطحی. برای مثال در بانک‌های اطلاعاتی NoSQL سندگرا، یک سند می‌تواند سند دیگری را در خود ذخیره کند و الی آخر.

ج) Sparse columns؛ ستون‌های اسپارس تقریباً شبیه به Key-value stores عمل می‌کنند و یا حتی Wide column stores نیز با آن قابل مقایسه است. در اینجا هنوز اسکیمای وجود دارد، اما برای نمونه علت استفاده از Wide column stores این نیست که واقعا نمی‌دانید ساختار داده‌های مورد استفاده چیست، بلکه در این حالت می‌دانیم که در هر ردیف تنها از تعداد معدودی از فیلدها استفاده خواهیم کرد. به همین جهت در هر ردیف تمام فیلدها قرار نمی‌گیرند، چون در اینصورت تعدادی از آن‌ها همواره خالی باقی می‌مانند. مایکروسافت این مشکل را با ستون‌های اسپارس حل کرده است؛ در اینجا هر چند ساختار کلی مشخص است، اما مواردی که هر بار استفاده می‌شوند، تعداد محدودی می‌باشند. به این صورت SQL Server تنها برای ستون‌های دارای مقدار، فضایی را اختصاص می‌دهد. به این ترتیب از لحاظ فیزیکی و ذخیره سازی نهایی، به همان مزیت Wide column stores خواهیم رسید.

د) FileStreams در اسی کیوال سرور بسیار شبیه به پیوست‌های بانک‌های اطلاعاتی NoSQL سندگرا هستند. در اینجا نیز اطلاعات در فایل سیستم ذخیره می‌شوند اما ارجاعی به آن‌ها در جداول مرتبط وجود خواهند داشت.

7 SQL Server Parallel Data Warehouse Edition

SQL PDW، نگارش خاصی از SQL Server است که در آن یک شبکه از SQL Serverها به صورت یک وهله منطقی SQL Server در اختیار برنامه نویس‌ها قرار می‌گیرد.

این نگارش، از فناوری خاصی به نام MPP یا massively parallel processing برای پردازش کوئری‌ها استفاده می‌کند. در اینجا همانند بانک‌های اطلاعاتی NoSQL، یک کوئری به نود اصلی ارسال شده و به صورت موازی بر روی تمام نودها پردازش گردیده (همان مفهوم Map Reduce که پیشتر در مورد آن بحث شد) و نتیجه در اختیار مصرف کننده قرار خواهد گرفت. نکته مهم آن نیز در عدم نیاز به نوشتن کدی جهت رخ دادن این عملیات از طرف برنامه نویس‌ها است و موتور پردازشی آن جزئی از سیستم اصلی است. تنها کافی است یک کوئری SQL صادر گردد تا نتیجه نهایی از تمام سرورها جمع آوری و بازگردانده شود.

این نگارش ویژه تنها به صورت یک Appliance به فروش می‌رسد (به صورت سخت افزار و نرم افزار باهم) که در آن CPUها،

فضاهای ذخیره سازی اطلاعات و جزئیات شبکه به دقت از پیش تنظیم شده‌اند.