

تکلیف اول: آشنایی با کلاذ و ذخیره سازهای مرتبط

۶ آبان ۱۴۰۱

۱. Region ها در AWS بزرگ هستند و به طور گسترده در مکان های جغرافیایی جداگانه پراکنده شده اند. Zone های در دسترس مکان های متمایز در یک منطقه AWS هستند که به گونه ای طراحی شده اند که از خرابی در سایر مناطق در دسترس جدا شوند. Local Zones به شما این امکان را می دهد که منابعی مانند محاسبات و ذخیره سازی را در چندین مکان نزدیک به کاربران نهایی خود قرار دهید.

۲.

ا. این موضوع باعث می شود که نرم افزارهای کاربردی که بر روی سرورهای شبکه وجود دارند به راحتی بتوانند از طریق شبکه به اطلاعات موجود در SAN دسترسی پیدا کنند به عبارت دیگر دستگاه های SAN وابستگی دستگاه های ذخیره سازی اطلاعات به سرورها را حذف می کنند و بصورت مجزا می توانند در شبکه ایفای نقش کنند طبیعی است که هر نرم افزاری بر روی هر سروری در شبکه می تواند به این منابع ذخیره سازی دسترسی پیدا کند و همین موضوع باعث بالا بردن کارایی هر چه بیشتر تجهیزات SAN در شبکه می شود. چالش SAN را می توان در هزینه و الزامات مدیریتی آن خلاصه کرد - اختصاص دادن و حفظ یک شبکه اترنت مجزا برای درخواست های فایل metadata و پیاده سازی شبکه کانال فیبر می تواند سرمایه گذاری قابل توجهی باشد.

ب. Block Virtualization فضای ذخیره سازی فرآیند ایجاد یک دیسک مجازی از یک یا چند دیسک فیزیکی است. این دیسک مجازی می تواند مانند هر دیسک دیگری مورد استفاده قرار گیرد و مزیت اضافه شده این است که می توان از چندین دیسک فیزیکی برای بهبود عملکرد یا قابلیت اطمینان استفاده کرد. این عمل در SAN می تواند برای بهبود عملکرد یک سرور یا خوشه ای از سرورها با ارائه فضای ذخیره سازی اختصاصی خود به آنها استفاده شود. یک SAN همچنین می تواند برای ایجاد افزونگی ذخیره سازی، با تکثیر داده ها در چندین دیسک فیزیکی استفاده شود.

• بررسی مزایا و معایب data striping

تکنیکی است که در آن چندین دیسک کوچکتر به عنوان یک دیسک بزرگ عمل می کنند. این فرآیند داده های بزرگ را به بلوک های داده تقسیم می کند و آن ها را در چندین دستگاه ذخیره سازی پخش می کند. نوار دیسک مزیت پایگاه داده های بسیار بزرگ یا فضای جدول تک جدولی بزرگ را با استفاده از تنها یک دستگاه منطقی فراهم می کند.

از مزایای data striping می توان به عملکرد و توان عملیاتی اشاره کرد. تلاقی متوالی زمانی دسترسی های داده ها باعث می شود که توان دسترسی کمتر به داده هر دستگاه ذخیره سازی به طور تجمعی در تعداد دستگاه های ذخیره سازی به کار گرفته شده ضرب شود. افزایش توان عملیاتی به دستگاه پردازش داده اجازه می دهد تا بدون وقفه به کار خود ادامه دهد و در نتیجه مراحل خود را با سرعت بیشتری به پایان برساند. این در بهبود عملکرد پردازش داده آشکار می شود.

از آنجایی که بخش های مختلف داده در دستگاه های ذخیره سازی مختلف نگهداری می شوند، خرابی یک دستگاه باعث خراب شدن توالی کامل داده ها می شود. در واقع، میزان خرابی آرایه ای از دستگاه های ذخیره سازی برابر با مجموع میزان خرابی هر دستگاه ذخیره سازی است. این نقطه ضعف data striping را می توان با ذخیره سازی اطلاعات اضافی، مانند برابری، به منظور تصحیح خطا برطرف کرد. در چنین سیستمی، ضرر به قیمت نیاز به ذخیره سازی اضافی برطرف می شود.

• بررسی مزایا و معایب data replication

فرآیندی است که در آن داده های موجود در سرور (های) فیزیکی/مجازی یا نمونه ابری (نمونه اولیه) به طور مداوم در سرور (های) ثانویه یا نمونه ابری (نمونه آماده به کار) تکثیر یا کپی می شوند. سازمان ها داده ها را برای پشتیبانی از دسترسی بالا، پشتیبان گیری و/یا بازیابی فاجعه تکثیر می کنند. بسته به محل نمونه ثانویه، داده ها به صورت همزمان یا ناهمزمان تکرار می شوند. نحوه تکثیر داده ها بر اهداف زمان بازیابی (RTO) و اهداف نقطه بازیابی (RPO) تأثیر می گذارد.

✓ مزایا:

کاهش بار: از آنجایی که داده های تکراری می توانند در چندین سرور پخش شوند، این احتمال را که هر یک از سرورها با درخواست های کاربر برای داده ها غرق شوند را از بین می برد.
بهره وری: سرورهایی که فشار کمتری با پرس و جو دارند می توانند عملکرد بهتری را به کاربران کمتری ارائه دهند.

در دسترس بودن بالا: استفاده از چندین سرور با داده های یکسان، دسترسی بالا را تضمین می کند، به این معنی که اگر یک سرور از کار بیفتد، کل سیستم همچنان می تواند عملکرد قابل قبولی ارائه دهد. افزایش پشتیبانی از تجزیه و تحلیل داده ها: تکرار داده ها در یک انبار داده به تیم های تجزیه و تحلیل توزیع شده قدرت می دهد تا روی پروژه های مشترک برای هوش تجاری کار کنند.
بهبود عملکرد سیستم تست: تکرار داده ها توزیع و همگام سازی داده ها را برای سیستم های آزمایشی که به دسترسی سریع به داده ها نیاز دارند، تسهیل می کند.

تکلیف اول: آشنایی با کلاذ و ذخیره سازهای مرتبط

۶ آبان ۱۴۰۱

✓ معایب:

پول: نگهداری کپی از داده‌های یکسان در مکان‌های مختلف منجر به افزایش هزینه‌های ذخیره‌سازی و پردازنده می‌شود.

زمان: پیاده‌سازی و مدیریت یک سیستم تکثیر داده‌ها نیازمند زمان اختصاصی یک تیم داخلی است. پهنای باند: حفظ یکنواختی بین نسخه‌های داده نیاز به رویه‌های جدیدی دارد و ترافیک را به شبکه اضافه می‌کند.

از دست رفتن داده‌ها: زمانی که داده‌ها یا تکرارها یا به‌روزرسانی‌های نادرست پایگاه داده کپی می‌شوند و در نتیجه، داده‌های مهم حذف یا حساب نشده است، می‌تواند در حین تکثیر، از دست دادن داده رخ دهد. در صورتی که کلید اصلی مورد استفاده برای تأیید کیفیت داده در ماکت خراب یا نادرست باشد، این اتفاق می‌افتد. همچنین اگر اشیاء پایگاه داده در پایگاه داده منبع به درستی پیکربندی نشده باشند، ممکن است رخ دهد.

ناهماهنگی داده‌ها به همین ترتیب، کپی‌های نادرست یا قدیمی می‌توانند باعث عدم هماهنگی منابع مختلف با یکدیگر شوند. این ممکن است منجر به هدر رفتن هزینه‌های انبار داده شود که بیهوده صرف تجزیه و تحلیل و ذخیره داده‌های نامربوط می‌شود.

اجرای چندین سرور دارای هزینه نگهداری و انرژی ذاتی است. به سازمان یا شخص ثالث نیاز دارد که به این هزینه‌ها رسیدگی کند. اگر شخص ثالثی آنها را مدیریت کند، سازمان در معرض خطر قفل شدن فروشنده یا مشکلات خدماتی خارج از کنترل سازمان قرار می‌گیرد.

در رایانش ابری، elasticity به این صورت تعریف می‌شود: درجه‌ای که یک سیستم می‌تواند با تدارک و جداسازی منابع به شیوه‌ای مستقل، با تغییرات بار کاری (work load) سازگار شود، به طوری که در هر نقطه از زمان، منابع موجود با تقاضای فعلی مطابقت داشته باشد. تا حد ممکن. کشش یک مشخصه تعیین کننده است که محاسبات ابری را از پارادایم‌های محاسباتی پیشنهادی قبلی مانند محاسبات شبکه ای متمایز می‌کند. انطباق پویا ظرفیت، به عنوان مثال، با تغییر استفاده از منابع محاسباتی، برای پاسخگویی به حجم کاری متفاوت، elastic computing (محاسبات الاستیک) نامیده می‌شود.

✓ Elasticity در AWS

Elasticity مبتنی بر حجم در مورد تطبیق مقیاس و افزودن منابع بر اساس تقاضا است. می‌توانید از آمازون EC۲ Spot Fleets استفاده کنید که می‌تواند به شما کمک کند تا نمونه‌ها را شروع کنید یا در صورت عدم نیاز آن‌ها را خاتمه دهید. نه تنها این، بلکه می‌توانید از خوشه‌های آمازون EMR برای بزرگ‌نمایی و مقیاس‌بندی گره‌های یک خوشه به صورت برنامه‌ریزی استفاده کنید.

Elasticity مبتنی بر زمان

Elasticity مبتنی بر زمان همه چیز در مورد آزاد کردن منابع زمانی است که دیگر به آنها نیاز ندارید. همیشه ایده خوبی است که فرآیند الاستیسیته مبتنی بر زمان را خودکار کنید. می‌توانید از زمان‌بندی نمونه AWS برای ایجاد یک زمان‌بندی خودکار برای شروع و توقف برای نمونه EC۲ استفاده کنید. همچنین می‌توانید از API های آمازون EC۲ برای خاتمه برنامه‌ریزی هر یک از نمونه‌ها استفاده کنید. نه تنها این، بلکه می‌توانید از توابع AWS Lambda برای خاموش کردن نمونه‌ها در مواقعی که دیگر مورد نیاز نیستند، استفاده کنید. Amazon CloudWatch یکی از ابزارهای عالی است که می‌تواند به خودکارسازی فرآیند در اینجا کمک کند.

✓ Elasticity در Azure

نصب ساده: از نرم افزار Elastic SaaS به عنوان یک سرویس یکپارچه در Azure استفاده کنید. صورت حساب یکپارچه: هزینه‌های Elastic را از طریق صورت حساب ماهانه Azure مدیریت کنید. Single-Sign on to Elastic: نیازی به ثبت نام جداگانه از پورتال Elastic نیست. ارسال کننده گزارش: فعالیت‌های اشتراک و گزارش‌های منابع را به Elastic منتقل کنید. تجربه مدیریت: یک مکان واحد برای نظارت بر نحوه ارسال گزارش های حمل و نقل خدمات Azure به Elastic ارائه می‌دهد.

استقرار Elastic Agent: عوامل الاستیک را در میزبان ماشین مجازی از طریق یک مرحله تنظیم کنید.

✓ Elasticity در IBM

چگونه IBM Elastic Capacity on Demand می‌تواند به کسب و کار شما کمک کند. تداوم کسب و کار را با گزینه‌های مصرف انعطاف پذیر در سیستم های قدرت IBM® حفظ کنید. ElasticCapacity on Demand به شما امکان می‌دهد با فعال و غیرفعال کردن هسته‌های پردازنده و واحدهای حافظه ظرفیت موقت ایجاد کنید تا به خواسته‌های اوج کسب و کار خود پاسخ دهید.

تکلیف اول: آشنایی با کلاذ و ذخیره سازهای مرتبط

۶ آبان ۱۴۰۱

فعال سازی ها را سفارش دهید و IBM بلافاصله کد فعال سازی را برای دانلود در سیستمتان در دسترس شما قرار می دهد. فعال سازی ها را می توان در هر سیستمی که مدیریت می کنید به اشتراک گذاشت. به سادگی هر تعداد از فعال سازی ها را از قبل خریداری کنید و سیستم به طور خودکار استفاده از آن ها را ردیابی می کند و در صورت مصرف آنها به شما هشدار می دهد و افزایش ظرفیت سخت افزاری POWER8 و POWER9 شما را آسان می کند.

IBM Elastic Storage® System (ESS) ذخیره سازی نرم افزاری است که استقرار ذخیره سازی سریع و بسیار مقیاس پذیر برای هوش مصنوعی و داده های بزرگ را آسان تر می کند. IBM ESS می تواند ناکارآمدی را کاهش دهد، هزینه های خرید را کاهش دهد، مدیریت ذخیره سازی را ساده تر کند، سیلوهای داده را حذف کند، از بارهای کاری متعدد پشتیبانی کند و عملکرد بالایی را در سراسر سازمان شما ارائه دهد.

ب.

Elasticity	Scalability
فقط برای پاسخگویی به بالا و پایین شدن ناگهانی حجم کار برای مدت زمان کم	برای پاسخگویی به افزایش ایستا در حجم کار
برای پاسخگویی به تغییرات پویا (جایی که نیاز به منابع می تواند افزایش یا کاهش یابد)	همیشه برای رسیدگی به افزایش حجم کاری در یک سازمان
معمولا توسط شرکت های کوچک استفاده می شود که حجم کار و تقاضا فقط برای یک دوره زمانی خاص افزایش می یابد	توسط شرکت های غول پیکر و بزرگی استفاده می شود که دایره مشتریان آن ها به طور مداوم رشد می کند تا عملیات را به طور موثر انجام دهند.
یک برنامه ریزی کوتاه مدت و فقط برای مقابله با افزایش غیرمنتظره تقاضا یا تقاضاهای فصلی	یک برنامه ریزی بلند مدت و فقط برای مقابله با افزایش مورد انتظار در تقاضا

۵.

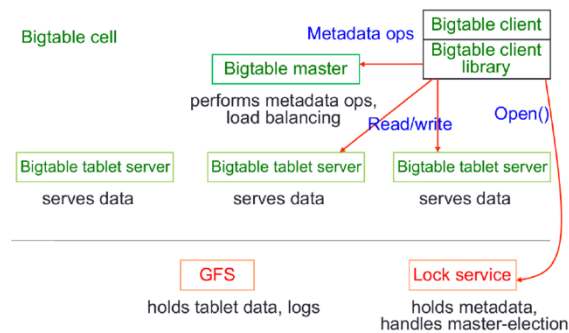
ا.

✓ ساختار و موارد کاربرد:

BigTable یک سیستم ذخیره سازی توزیع شده است که توسط Google برای ذخیره حجم عظیمی از داده ها و مقیاس کردن هزاران سرور ذخیره سازی توسعه یافته است. برای نمایه سازی وب، جستجوی شخصی، Google Earth، Google Analytics، Google Finance استفاده می شود. سیستم از GFS برای ذخیره داده های کاربر و همچنین اطلاعات سیستم استفاده می کند. برای تضمین عملیات خواندن و نوشتن اتمی، BigTable از سرویس قفل توزیع شده Chubby استفاده می کند. Megastore به طور گسترده در داخل گوگل استفاده می شود. فلسفه طراحی اساسی سیستم این است که داده ها را به گروه های موجودیت تقسیم کند و هر پارتیشن را به طور مستقل در مراکز داده واقع در مناطق مختلف جغرافیایی تکرار کند. سیستم از Full ACID Semantics در هر پارتیشن پشتیبانی می کند و تضمین های سازگاری محدودی را در بین پارتیشن ها ارائه می دهد.

✓ معماری:

BigTable



این سیستم از سه جزء اصلی تشکیل شده است:

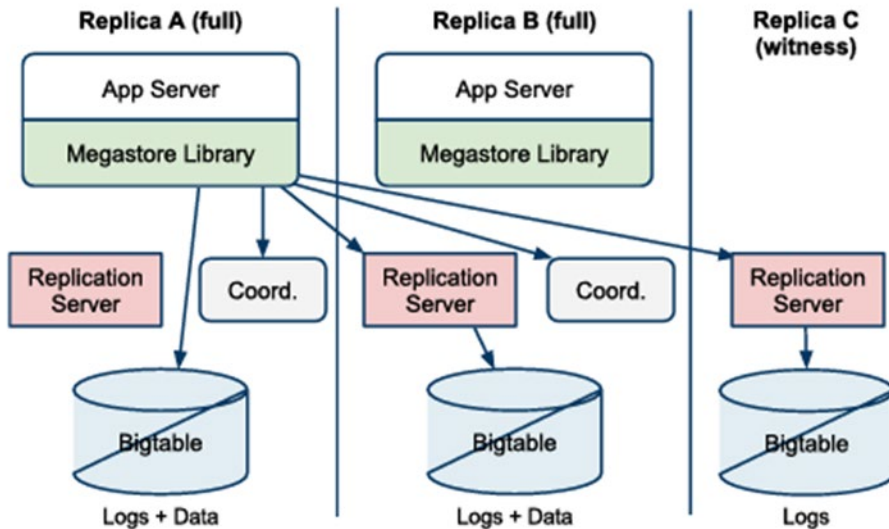
- کتابخانه ای که برای دسترسی به سیستم به کلاینت های برنامه مرتبط است
- یک سرور اصلی: کل سیستم را کنترل می کند، تبلت ها را به سرورهای تبلت اختصاص می دهد و بار را بین آنها متعادل می کند، جمع آوری زباله را مدیریت می کند و ایجاد و حذف خانواده جدول و ستون را مدیریت می کند.
- تعداد زیادی سرور تبلت:

❖ جدول Bigtable بر اساس کلیدهای ردیف به بسیاری از تبلت ها تقسیم می شود.

❖ هر تبلت حاوی بلوک هایی از ردیف های به هم پیوسته است.

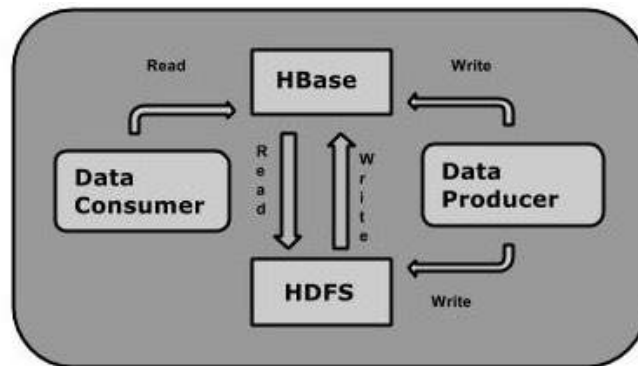
MegaStore:

MegaStore از طریق یک کتابخانه مشتری و سرورهای کمکی مستقر می شود. برنامه ها به کتابخانه مشتری پیوند می خورند، که Paxos و الگوریتم های دیگر را پیاده سازی می کند: انتخاب یک ماکت برای خواندن، گرفتن یک ماکت عقب افتاده و غیره. هر سرور برنامه یک ماکت محلی تعیین شده دارد. کتابخانه مشتری، عملیات Paxos را روی آن ماکت با ارسال مستقیم تراکنش ها به Bigtable محلی، پایدار می کند. برای به حداقل رساندن سفرهای رفت و برگشت گسترده، کتابخانه Paxosoperations از راه دور را به سرورهای تکراری واسطه بدون حالت که با Bigtable های محلی خود در ارتباط هستند، ارسال می کند. خرابی های کلاینت، شبکه یا Bigtable ممکن است باعث شود نوشتن در حالت نامشخصی رها شود. سرورهای Replication به صورت دوره ای نوشته های ناقص را اسکن می کنند و مقادیر no-op را از طریق Paxos پیشنهاد می کنند تا آنها را کامل کنند.



ب.

HBase یک پایگاه داده ستون محور است که بر روی فایل سیستم Hadoop ساخته شده است. این یک پروژه منبع باز است و به صورت افقی مقیاس پذیر است. HBase یک مدل داده است که شبیه جدول بزرگ گوگل (Google BigTable) است که برای دسترسی تصادفی سریع به حجم عظیمی از داده‌های ساختار یافته طراحی شده است. از تحمل خطا ارائه شده توسط سیستم فایل (HDFS) Hadoop استفاده می‌کند. این بخشی از اکوسیستم Hadoop است که دسترسی تصادفی خواندن/نوشتن بلادرنگ به داده‌ها در سیستم فایل Hadoop را فراهم می‌کند. می‌توان داده‌ها را به صورت مستقیم یا از طریق HBase در HDFS ذخیره کرد. مصرف کننده داده به صورت تصادفی با استفاده از HBase به داده‌ها در HDFS دسترسی پیدا می‌کند. HBase در بالای سیستم فایل Hadoop قرار دارد و دسترسی خواندن و نوشتن را فراهم می‌کند.



HBase یک پایگاه داده ستون محور است و جداول موجود در آن بر اساس ردیف مرتب شده‌اند. طرح جدول فقط خانواده‌های ستونی را تعریف می‌کند که جفت مقادیر کلیدی هستند. یک جدول دارای چندین خانواده ستون است و هر خانواده ستون می‌تواند هر تعداد ستون داشته باشد. مقادیر ستون‌های بعدی به طور پیوسته روی دیسک ذخیره می‌شوند. هر مقدار سلول جدول دارای مهر زمانی است. به طور خلاصه، در یک HBase :

- ✓ جدول مجموعه‌ای از ردیف‌ها است.
- ✓ ردیف مجموعه‌ای از خانواده‌های ستون است.
- ✓ خانواده ستون‌ها مجموعه‌ای از ستون‌ها هستند.
- ✓ ستون مجموعه‌ای از جفت‌های ارزش کلیدی است.

ویژگی‌های HBase :

- ✓ مقیاس پذیری به صورت خطی
- ✓ داشتن پشتیبانی از شکست خودکار
- ✓ فراهم کردن خواندن و نوشتن مداوم
- ✓ ادغام با Hadoop هم به عنوان منبع و هم به عنوان مقصد
- ✓ داشتن API جاوا آسان برای Client
- ✓ فراهم کردن تکثیر داده‌ها در میان خوشه‌ها فراهم

محل استفاده از HBase :

Apache HBase برای دسترسی تصادفی و بلادرنگ خواندن/نوشتن به داده‌های بزرگ استفاده می‌شود. میزبان میزهای بسیار بزرگ در بالای خوشه‌های سخت افزاری کالا است. Apache HBase یک پایگاه داده غیر رابطه‌ای است که از Bigtable گوگل مدل شده است. Bigtable در سیستم فایل Google فعال می‌شود، همچنین Apache HBase در بالای Hadoop و HDFS کار می‌کند.

کاربردهای HBase :

✓ هر زمان که نیاز به نوشتن برنامه‌های سنگین وجود داشته باشد از آن استفاده می‌شود.

✓ نیاز داشتن به دسترسی تصادفی سریع به داده‌های موجود

شرکت‌های استفاده کننده از HBase:

✓ شرکت‌هایی مانند فیسبوک، توییتر، یاهو و ادوبی از HBase به صورت داخلی استفاده می‌کنند.