

واژه‌نامه پایگاه‌های داده NoSQL

Bigtable (جدول بزرگ)

جدول بزرگ یا Bigtable یک پایگاه داده توزیع شده است که در سال ۲۰۰۶ به عنوان یک پایگاه داده ستونی بر مبنای سیستم فایل گوگل (GFS) ساخته شده است. Bigtable و داینامی آموزشی و والدین مستقیم کاساندر هستند. کاساندر را خصوصیات آرایه‌های داده‌ای پراکنده و ذخیره‌سازی روی دیسک با استفاده از SSTable را از این پایگاه داده به ارث برده است. پایگاه داده HBace یا هو یک بازسازی (کلون) از Bigtable است. مقاله کامل گوگل درباره Bigtable را می‌توانید در آدرس زیر بخوانید:

<http://lab.google.com/papers/bigtable.html>

Cassandra (کاساندر)

در افسانه‌های یونان، کاساندر دختر شاه پریام و ملکه هکوبا از خانواده سلطنتی تروا بود. او به حدی زیبا بود که آپولو قدرت دیدن آینده را به او عطا کرد. اما زمانی که او این لطف عاشقانه را نپذیرفت، آپولو او را نفرین کرد تا هر چه در آینده رخ خواهد داد را به دقت پیش‌بینی کند، اما هیچ کس حرف‌های او را باور نکند. او ویرانی شهرش تروا را دید، اما توان مقابله با این اتفاق را نداشت...
نام این پایگاه داده توزیع شده، از روی همین افسانه انتخاب شده است. این انبار داده یکی از پروژه‌های بنیاد آپاچی است که در آدرس <http://cassandra.apache.org> در دسترس است. مراحل اولیه شکل‌گیری آن در ژانویه ۲۰۰۹ طی شد. از خصوصیات کلیدی این پایگاه داده می‌توان به عدم تمرکز، الاستیک بودن، قابلیت تحمل خطا، ثبات قابل تنظیم، بهینه‌سازی و دسترس پذیری بالا اشاره کرد. همچنین این پایگاه داده از ابتدا به گونه‌ای طراحی شده است که به شدت مقیاس پذیر بوده و به سادگی روی سرورهای معمولی که در دیتاسنترهای مختلف پراکنده شده‌اند، توزیع شود. شرکت‌هایی مانند Digg، فیس بوک، Cloudkick، سیسکو، آی بی ام، Reddit، Rackspace، SimpleGeo، Ooyala و OpenX از این پایگاه داده استفاده می‌کنند.

Cluster (خوشه، کلاستر)

به دو یا چند نسخه (Instance) از کاساندر که در هماهنگی با یکدیگر کار می‌کنند، خوشه گفته می‌شود. این نسخه‌ها به کمک Gossip با یکدیگر ارتباط برقرار می‌کنند. وقتی شما یک نود جدید را برای معرفی به کلاستر مورد نظرتان پیکربندی می‌کنید، باید چند کار کوچک انجام دهید. ابتدا باید یک نود Seed معرفی کنید. پس از آن باید پورت‌هایی را تعریف کنید که کاساندر باید برای ارتباط با Thrift و Gossip به آن‌ها گوش دهد. پس از این که کلاستر شما پیکربندی شد، باید از ابزار Node Tool برای کنترل صحت تنظیمات استفاده کنید.

Column (ستون)

ستون، ابتدایی ترین واحد ارائه اطلاعات در مدل داده‌ای کاساندر است. یک ستون ترکیبی سه گانه از یک نام (که گاهی کلید یا key نامیده می‌شود)، یک مقدار و یک برچسب زمانی است. مقادیر یک ستون به همراه برچسب زمانی آن توسط کلاینت پایگاه داده فراهم می‌شود. نوع داده کلید و مقدار، هر دو آرایه‌های بایتی جاوا (Java Byte Array) است. نوع داده برچسب زمانی یک نوع طولانی (Long) اولیه است. برای جلوگیری از بروز مشکل در حالت چندرشته‌ای، ستون‌ها به صورت غیر قابل تغییر (Immutable) تعریف شده‌اند. ستون‌ها در خانواده‌های ستون (Column Families) دسته‌بندی می‌شوند.

Consistency (ثبات)

ثبات به این معنا است که یک تراکنش هیچ گاه پایگاه داده را در یک حالت ناپایدار یا غیر مجاز ترک نخواهد کرد و همچنین هیچ یک از قیدهای یکپارچگی نقض نخواهند شد. این مورد به یکی از جنبه‌های حیاتی تراکنش‌ها در پایگاه‌های داده رابطه‌ای و یکی از خصوصیات ACID (سرنام Atomic, Consistent, Isolated, Durable به معنی اتمیک، پایدار، ایزوله شده و دارای ماندگاری) اشاره می‌کند. در کاساندر درجه نسبی پایداری براساس موارد زیر محاسبه می‌شود:

N - تعداد نودهایی است که کپی‌های یک داده را نگاه‌داری می‌کنند.

W - تعداد نودهایی که باید دریافت صحیح داده را تأیید کنند.

R - تعداد کپی‌هایی که در هنگام دسترسی به یک شیء داده در عملیات خواندن اطلاعات، با آن‌ها تماس برقرار می‌شود.

- در این صورت اگر $R+W$ از N بزرگتر باشد، ثبات بالایی خواهیم داشت و در غیر این صورت ثبات مشروط خواهد بود.

Decentralized (تمرکز زدایی شده)

کاساندرها به عنوان یک پایگاه داده تمرکززدایی شده در نظر گرفته می شود چرا که هیچ سرور مرکزی ای تعریف نمی کند و در عوض از یک رویکرد نظیر به نظیر استفاده می کند تا از بروز گلوگاه و ایجاد یک نقطه شکست مرکزی جلوگیری کند. تمرکززدایی در کاساندرها بسیار حیاتی به شمار می آید، زیرا امکان افزایش مقیاس یا کاهش آن را برای کاساندرها فراهم می کند. نوها می توانند به دلخواه به شبکه افزوده شده یا از شبکه بیرون روند و این امر کمترین تأثیری در کل پایگاه داده نداشته باشد.

Denormalization (غیر نرمال سازی)

در پایگاه های داده رابطه ای، غیر نرمال سازی یا ایجاد داده های اضافی، فرآیندی است که برای بهبود کارایی برنامه هایی که بیشتر با عملیات خواندن داده ها سروکار دارند، انجام می پذیرد. پردازش تحلیلی آنلاین OLAP (سرنام OnLine Analytical Processing) از جمله این فرآیندها است. در کاساندرها دیدن داده های غیر نرمال شده بسیار طبیعی است؛ چرا که این کار باعث افزایش کارایی شده و کمک می کند که داده ها براساس پرس و جوهای مورد نیاز ساختاردهی شوند. این درست نقطه مقابل پایگاه های داده سنتی رابطه ای است که به صورت مستقل بر پایه مدل شیء (Object Model) ساختاردهی می شوند.

Dynamo (داینامو)

این سیستم که در سال ۲۰۰۶ توسط آمازون و همزمان با Bigtable گوگل ایجاد شده است، یکی از پایه های کاساندرها است. کاساندرها با استفاده از داینامو به یک انبار داده های کلید-مقدار، یک معماری نظیر به نظیر متقارن، کشف داده گمان محور (gossip-based discovery) و ثبات مشروط دست می یابد. شما می توانید مقاله «داینامو: انبار کلید-مقدار کاملاً در دسترس آمازون» را در آدرس زیر ببینید:

http://www.allthingsdistributed.com/2007/10/amazons_dynamo.html

Elastic (کشسانی، الاستیک)

میزان عملیات خواندن و نوشتن داده ها با افزودن ماشین های بیشتر به کلاستر به صورت خطی افزایش می یابد. تحمل خطا یا Fault Tolerance توانایی ادامه کار حتی در صورت خرابی یک یا چند بخش سیستم است. از این توانایی گاهی با نام تنزل مطبوع یا graceful Degradation نیز نام برده می شود و به این معنا است که اگر کارایی سیستم در اثر بروز خطا تنزل پیدا کرد، کاهش کارایی تنها متناسب با بخش از کار افتاده باشد.

Hector (هکتور)

پروژه اپن سورسی است که توسط ران تاوری (Rantavory) پایه گذاری شده و روی Github میزبانی می شود. هکتور، کلاستی برای کاساندرها است که به وسیله جاوا نوشته شده است. این پروژه Thrift را پوشانده (wrap کرده) و می تواند قابلیت های Connection Pooler JMX و Fail over را ارائه کند.

Memtable (جدول حافظه)

به نسخه ای از داده ها که به تازگی در جدولی در حافظه نوشته شده باشد، اطلاق می شود. هنگامی که این جدول پر شود به صورت یک SSTable روی دیسک ذخیره خواهد شد.

Node (گره یا نود)

یک نسخه در حال اجرا از کاساندرها را یک نود می گویند. کلاسترهای کاساندرها به صورت معمول تعداد زیادی نود دارند و به همین دلیل گاهی حلقه نودها یا به تنهایی «حلقه» نامیده می شوند. اصطلاح نود به هر سروری در کلاستر کاساندرها اشاره می کند، در صورتی که «رپلیکا» (Replica) به نودی اشاره می کند که یک کپی از برخی داده های یک نود دیگر را در خود داشته باشد.

NoSQL

اصطلاح NoSQL نامی عمومی است که به مجموعه ای از پایگاه های داده اطلاق می شود که از زبان پرس و جوی ساخت یافته SQL (سرنام Structured Query Language) یا مدل داده رابطه ای استفاده نمی کنند. گاهی این اصطلاح را مخفف Not Only SQL می دانند تا تأکید کنند که طرفداران انواع پایگاه های داده غیر رابطه ای معتقدند که پایگاه های داده رابطه ای سنتی تنها راه موجود برای ذخیره سازی داده نیستند، اما این به آن معنا نیست که به خودی خود انتخاب نادرستی باشند. این اصطلاح نخستین بار توسط اریک اوانس از Rackspace به کار رفت. او که یکی از توسعه دهندگان کاساندرها است، پس از آن از به کار بردن این اصطلاح خودداری می کند و به جای آن مایل است اصطلاح BigData یا داده های عظیم را به کار ببرد تا این گروه از پایگاه های داده را نه براساس چیزی که نیستند (سازگار با SQL) بلکه براساس کاری که می کنند (مدیریت مقادیر عظیم داده) تعریف کند. دوره استفاده از این اصطلاح کم و بیش به پایان رسیده است چرا که بسیار گیج کننده است و به نظر می رسد باعث می شود راجع به مجموعه ای از پایگاه های داده بحث کنیم که در عمل میزان شباهت میان اهداف، ایده های طراحی و قابلیت های آن ها بسیار اندک است. بهتر است بگذاریم کاساندرها همان کاساندرها، CouchDB همان CouchDB و Riak همان Riak بمانند.

Partition (پارتیشن)

به صورت کلی، پارتیشن به یک بخش مجزا از شبکه اشاره می‌کند. این جداسازی بخش‌ها در اثر یک شکست یا قطعی در شبکه ایجاد شده و باعث می‌شود یک ماشین نتواند به صورت مستقیم با یک ماشین دیگر تماس برقرار کند. پارتیشن ممکن است در اثر خرابی یک سویچ، روتر یا رابط‌های شبکه به وجود آید. کلاستری از ۵ ماشین {A,B,C,D,E} را تصور کنید که در آن {A,B} روی یک زیر شبکه یا Subnet و {C,D,E} روی زیر شبکه دیگری باشند. اگر سویچی که به گروه دوم متصل است از کار بیفتد، شما پارتیشنی خواهید داشت که این دو زیر گروه را از هم جدا خواهد کرد. کاساندرای پایگاه داده‌ای است که قابلیت تحمل خطاها را دارد و پارتیشن‌های شبکه یکی از این خطاها هستند. به همین دلیل کاساندرای می‌تواند در صورت مواجه شدن با یک پارتیشن به کار خود ادامه دهد و داده‌ها را پس از برطرف شدن مشکل پارتیشن، دوباره با هم ادغام کند.

Replication (تکثیر)

به صورت کلی در سیستم‌های توزیع شده، اصطلاح تکثیر به ذخیره چندین کپی از داده روی چندین ماشین مختلف اطلاق می‌شود. این امر به این دلیل انجام می‌شود که در صورت خرابی یکی از ماشین‌ها یا عدم امکان دسترسی به آن در صورت ایجاد یک پارتیشن، کلاستر هنوز بتواند به داده‌ها دسترسی داشته باشد. کش کردن داده‌ها (Caching) یکی از ساده‌ترین راه‌های تکثیر است. در کاساندرای، تکثیر ابزاری برای فراهم آوردن کارایی بالا، دسترس پذیری و قابلیت تحمل خطا است.

Row (سطر)

در یک خانواده ستون، یک سطر نقشه‌ای مرتب شده است که نام ستون‌ها را با مقادیر آن‌ها انطباق می‌دهد. در یک ابرستون (SuperColumn) یک سطر، نقشه‌ای مرتب شده است که اسامی ابرستون را با نقشه ارتباطی اسم ستون و مقدار ستون متناظر آن پیوند می‌دهد.

Thrift

نام یک کلاینت فراخوانی راه دور پردازنده (RPC سرنام Remote Procedure Call) است که برای ارتباط با سرور کاساندرای مورد استفاده قرار می‌گیرد. این کلاینت به صورت استاتیک رابطی برای سریال سازی را در بسیاری از زبان‌ها از جمله ++C، جاوا، پایتون، پی‌اچ‌پی، روبی، ارلنگ، پرل، هسکل، سی شارپ، کاکائو، اسمال‌تاک و OCaml فراهم می‌کند. همین مکانیسم است که به شما اجازه می‌دهد از طریق تمام این زبان‌ها با کاساندرای ارتباط برقرار کنید. در زمان نوشتن این مطلب احتمال زیادی وجود دارد که پروژه‌ای جدیدتر و فعال‌تر از آپاچی به نام آورو (Avro) جایگزین رابط Thrift شود. یکی از مزیت‌های مهم آورو این است که به ایجاد کدهای استاتیک احتیاجی ندارد.

Hadoop (هادوپ)

هادوپ یک فریم‌ورک نرم‌افزاری است که کاربردهای توزیع شده با داده‌های فراوان را تحت یک مجوز آزاد، پشتیبانی می‌کند. این فریم‌ورک به برنامه‌ها امکان می‌دهد که با هزاران نود و داده‌هایی در اندازه‌های پتابایت کار کنند. هادوپ از مقالات مرتبط با سیستم Map Reduce گوگل و همچنین سیستم فایلی آن GFS، الهام گرفته است.

HDFS

یک سیستم فایلی توزیع شده، مقیاس پذیر و قابل حمل (portable) است که با زبان جاوا برای هادوپ نوشته شده است. این سیستم فایلی برای ارتباطات از لایه TCP/IP استفاده می‌کند و کلاینت‌های آن از RPC برای ارتباط با یکدیگر بهره می‌برند. سیستم فایلی HDFS فایل‌های بزرگ (اندازه ایده‌آل فایل‌ها مضارب ۶۴ مگابایت است) را در ماشین‌های مختلف ذخیره می‌کند. این سیستم برای افزایش قابلیت اعتماد، داده‌ها را در میزبان‌های مختلفی تکثیر می‌کند و به همین دلیل به قابلیت RAID روی میزبان‌ها احتیاجی ندارد. با مقدار پیش فرض تکثیر ۳، داده‌ها روی سه نود ذخیره می‌شوند که از این سه نود، دو نود در یک رک و یکی در رک دیگری واقع شده است.

HBase

پایگاه داده هادوپ برای دسترسی خواندن و نوشتن تصادفی است. HBase پایگاه داده توزیع شده، غیر رابطه‌ای و این سورس است که از روی BigTable گوگل مدل سازی و به وسیله جاوا نوشته شده است. این پایگاه داده به عنوان جزئی از پروژه هادوپ در بنیاد نرم‌افزار آپاچی توسعه داده شده است و روی HDFS اجرا می‌شود و قابلیت‌هایی همانند BigTable را برای هادوپ به ارمان می‌آورد. به این معنا که امکان ذخیره سازی حجم عظیمی از داده‌های پراکنده را به روشی که تحمل خطا را نیز دارد، فراهم می‌کند. پایگاه داده HBase قابلیت‌های فشرده سازی، اجرای عملیات داخل حافظه و فیلترهای انفجاری به صورتی ستون محور را؛ در ست همان گونه که مقاله اصلی BigTable تشریح کرده است، فراهم می‌کند. جدول‌های HBase می‌توانند به عنوان ورودی و خروجی عملیات Map Reduce که روی هادوپ در حال اجرا هستند به کار برده شوند. از طریق API‌های جاوا، REST، آورو و یا گیت‌وی‌های Thrift می‌توان به این جدول‌ها دسترسی پیدا کرد.

Hive (هایو)

پرس وجوها و جداول شبه SQL که روی مجموعه‌های عظیم داده عمل می‌کنند. هایو آپاچی یک زیرساخت برای انبارهای داده (data warehouse infrastructure) است که بر مبنای هادوپ و برای خلاصه‌سازی، پرس وجو و تحلیل داده‌ها ساخته شده است. اگرچه در ابتدا توسط یکی از شبکه‌های اجتماعی مشهور توسعه داده شده، اما اکنون به عنوان یک پروژه آپاچی در سایر شرکت‌ها نظیر نت‌فلیکس هم مورد استفاده قرار می‌گیرد. هایو اکنون در Amazon Elastic Map Reduce که از سرویس‌های آمازون است به کار می‌رود. هایو امکان تحلیل مجموعه‌های عظیم داده که روی سیستم‌های فایلی سازگار با هادوپ؛ نظیر S3 آمازون، ذخیره شده‌اند را فراهم می‌آورد. هایو در عین پشتیبانی کامل از MapReduce، زبانی شبیه به SQL با نام HiveQL را نیز در اختیار کاربران قرار می‌دهد. هایو برای سریع تر کردن پرس وجوها از اندیس‌ها؛ از جمله اندیس‌های Bitmap استفاده می‌کند.

Pig (پیگ)

پیگ پلتفرمی سطح بالا برای ایجاد برنامه‌های MapReduce روی هادوپ است. زبان مورد استفاده در این پلتفرم پیگ لاتین (Pig Latin) نامیده می‌شود. پیگ لاتین برنامه‌نویس را از کدنویسی MapReduce براساس جاوایی نیاز کرده و این قابلیت‌ها را به صورت سیستم نگارشی سطح بالایی عرضه می‌کند که بسیار شبیه SQL در سیستم‌های RDBMS است. پیگ لاتین را می‌توان به کمک UDF (سرنام User Defined Functions یا توابع تعریف شده توسط کاربر) گسترش داد که کاربر می‌تواند این توابع را در جاوا پیاده‌سازی کرده و بعدتر مستقیماً از طریق پیگ لاتین فراخوانی کند. پیگ در اصل در سال ۲۰۰۶ در مراکز تحقیقاتی یاهو برای محققان توسعه داده شد تا روشی ساده برای ایجاد و اجرای عملیات MapReduce روی مجموعه‌های بسیار عظیم داده در اختیار داشته باشند. در سال ۲۰۰۷ این پروژه به بنیاد نرم‌افزاری آپاچی منتقل شد.

Oozie (اوزی)

نام گردش کاری است که عملیات‌های مختلف هادوپ وابسته به هم، براساس آن انجام می‌شوند. اوزی یک سرویس مختصات دهی و گردش کار است که مدیریت عملیات مختلف پردازش داده را به عهده دارد. اوزی سرویسی قابل گسترش، مقیاس پذیر و حساس به داده است که می‌تواند نیازها و ارتباطات میان عملیات در حال اجرا روی هادوپ (نظیر HDFS، پیگ و MapReduce) را هماهنگ کند. اوزی خیلی از کارها را به انجام می‌رساند اما دو کار را نمی‌تواند انجام دهد:

- فراهم آوردن راه‌حلی برای پردازش مستقل از هادوپ

- فراهم کردن یک API جدید برای پردازش پرس وجوها

Sqoop (اسکوپ)

پایگاه‌های داده و انبارهای داده را با هادوپ ادغام می‌کند. پروژه اسکوپ آپاچی ایجاد شده است تا داده‌های حجیم را به صورتی کارا میان هادوپ و سیستم‌های ذخیره‌سازی ساخت یافته بیرونی نظیر RDBMS و انبارهای داده (Data warehouse) جابه‌جا کند، چرا که این پایگاه‌های داده به سادگی از طریق هادوپ قابل دسترسی نیستند. اسکوپ اکنون در حال طی مراحل Incubation (مراحل اولیه پذیرش یک پروژه) در بنیاد نرم‌افزاری آپاچی است.

Zookeeper

Zookeeper سرویسی برای برنامه‌های توزیع شده است. به عبارت دیگر سرویسی مرکزی است که نگهداری اطلاعات پیکربندی، نام‌گذاری، فراهم آوردن همسان‌سازی توزیع شده و همچنین ایجاد سرویس‌های گروهی را بر عهده دارد. تمام این سرویس‌های متفاوت به نوعی در برنامه‌های توزیع شده مورد استفاده قرار می‌گیرند.

Hue

فریم‌ورک و SDK رابط کاربری برای برنامه‌های بصری هادوپ است. Hue هم یک رابط کاربری مبتنی بر وب هادوپ است و هم فریم‌ورکی برای ایجاد برنامه‌های تعاملی وب. این فریم‌ورک شامل یک File Browser برای دسترسی به HDFS، یک برنامه JobSub و JobBrowser برای ایجاد یا مشاهده عملیات Map Reduce و یک برنامه Beeswax برای تعامل با هایو است. علاوه بر همه این‌ها، رابط وبی آن بیشتر بر مبنای ابزارک‌های توصیفی (Declarative Widgets) ساخته شده است که نیازی به جاوااسکریپت ندارد و یادگیری آن بسیار ساده است.

Map Reduce

یک فریم‌ورک نرم‌افزاری است که در سال ۲۰۰۴ توسط گوگل عرضه شد تا محاسبات توزیع شده روی مجموعه عظیمی از داده‌های ذخیره شده روی کلاسترهای کامپیوتری را پشتیبانی کند. برخی از بخش‌های این فریم‌ورک در برخی کشورها به صورت حق اختراع انحصاری (Patent) ثبت شده‌اند. این فریم‌ورک از توابع map و reduce که به صورت معمول در برنامه‌نویسی رویه‌ای مورد استفاده قرار می‌گیرند، الهام گرفته است؛ هرچند که عملکرد آن‌ها در اینجا هیچ شباهتی به فرم اولیه ندارد. کتابخانه‌های MapReduce به زبان‌های مختلفی از جمله ++C، سی شارپ، ارلنگ، جاوا، OCaml، LabView، پرل، پایتون، پی‌اچ‌پی، روبی، #F، R و سایر زبان‌ها نوشته شده‌اند.