

پیاده سازی چهارچوب کد باز Hadoop و بررسی سرویس نگاشت کاهش در آن

پروین احمدی^۱، محمد رضا سلطان آقایی^۲

^۱ دانشجوی کارشناسی ارشد مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد بابل، parvin_ahmadi85@yahoo.com

^۲ استادیار مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد خوراسگان، msoltanaghahi@yahoo.com

چکیده - یکی از جنبه های مهم پردازش ابری که بطور گسترده مورد استفاده قرار می گیرد پردازش روی مجموعه داده های بسیار بزرگ است. برای پردازش این داده ها مدل ها و چهارچوب های گوناگونی طراحی و توسعه داده شده است. هادوپ (Hadoop) یکی از بسترهای کد باز است که برای ذخیره سازی و پردازش حجم اطلاعات بسیار بالا طراحی شده است. در این بستر، امکان ذخیره سازی اطلاعات در چندین سرور با هزینه ای پایین فراهم می شود و به دلیل انعطاف پذیر بودن در محیط های خوشه ای (cluster-based) و همچنین کار با داده های فشرده برای مدیریت سیستم های ابری استفاده می شود. یکی از راه های پیاده سازی تجاری هادوپ، استفاده از Cloudera است که به وسیله آن می توان هادوپ را نصب، راه اندازی، پیکربندی و مدیریت کرد. در این مطالعه، ابتدا بستر Hadoop بوسیله Cloudera پیاده سازی گردید. سپس روش های نگاشت - کاهش (Map - Reduce)، معماری موازی (Parallel Databases)، و روش های ترکیبی مانند HadoopDB تشریح و مورد بررسی قرار می گیرد. با بررسی و مقایسه روش های فوق، به این نتیجه رسیدیم که HadoopDB با ترکیبی از معماری DBMS و MapReduce سعی در بهره گیری از مزایای هر کدام از روش ها دارد. همچنین بدلیل رایگان و متن باز بودن این سیستمها، از هزینه بسیار کمتری نسبت به اکثر پایگاه داده های موازی - تجاری برخوردار است.

کلید واژه - محاسبات ابری (Cloud Computing)، هادوپ (Hadoop)، نگاشت - کاهش (MapReduce)، Cloudera

پردازش ها روی ماشین های مختلف به صورت موازی اجرا می گردد ، داده های هر ماشین توسط خودش پردازش می شود و در آخر نتایج جمع آوری می گردد [10].

شرکت گوگل در پی افزایش حجم تبادل اطلاعات، به دنبال راه حلی برای افزایش سرعت و راندمان سرورهای خود بود که سیستم توزیع (Distribution) منحصر به فردی برای خود ابداع کرد به نام GFS که مخفف Google File System بود. در پی این موفقیت، انجمن توزیع Apache به فکر گسترش این تکنولوژی در سطح وسیع تری افتاد و سیستم هادوپ (Hadoop) به وجود آمد. سپس رایانش توزیع شده ی داده ها روی کلاسترها بوسیله یک مدل برنامه نویسی ساده با نام نگاشت کاهش شکل گرفت [1].

در این تحقیق بستر Hadoop بوسیله Cloudera پیاده سازی گردید [9]. سپس چارچوب نرم افزاری نگاشت کاهش (MAP - Reduce) در آن تشریح گردید [3] که به انجام عملیات

۱. مقدمه

امروزه رشد چشمگیر حجم داده ها را داریم به عنوان نمونه روزانه ۲۰۰ میلیون عکس در فیسبوک ذخیره می شود. (اگر حجم هر عکس را 100KB در نظر بگیریم، این تعداد عکس برابر است با ۲۰ ترابایت در روز) در حال حاضر فیسبوک ۱۴۰ میلیارد عکس را در خود ذخیره کرده است.

حال این سوالات مطرح میشود که : چگونه فیسبوک داده های ۳۰۰ میلیون کاربر خود را نگهداری و پردازش می کند؟ چگونه فیسبوک در هر ثانیه 600,000 عکس را پردازش می کند؟ چگونه گوگل و یاهو کل اینترنت را ایندکس گذاری کرده و نتایج جستجوی میلیونها کاربر را در کسری از ثانیه نمایش می دهند؟

حل نرم افزاری مشکل پردازش داده های حجیم به کمک رایانش ابری بدین صورت انجام می پذیرد که ابتدا داده ها تقسیم و بر روی ماشین های مختلف ذخیره میشود سپس

لازم را به آن دهیم. سپس با استفاده از دستور service network wizard شبکه ماشین مجازی را دوباره راه اندازی می کنیم. بوسیله ابزاری به اسم putty میتوانیم به ماشین مجازی با استفاده از پروتکل ssh متصل شد.

بعد از باز کردن نرم افزار putty، به آدرس آن ماشین مجازی با استفاده از پروتکل ssh متصل شویم. به جای این که از داخل محیط vmware، فرمانها تایپ شود، سعی می کنیم که از ابزار putty استفاده کرده و آدرس آن ماشینی که می خواهیم به آن وصل شویم را وارد کنیم و به ماشین مجازی متصل شویم:

Host name: root@192.168.1.4

یک ماشین مجازی جدید ایجاد می کنیم که بتوانیم clustering را اجرا کنیم. چون برای cluster کردن به چند ماشین مجازی نیاز داریم. اسم آن را CentOS Client می گذاریم. به تنظیمات ماشین مجازی رفته از فایل centOS 64 استفاده می کنیم و به همین ترتیب در قسمت تنظیمات شبکه network adaptor و گزینه bridg را انتخاب می کنیم. و ماشین مجازیمان را روشی میکنیم.

بعد از انجام عملیات نصب ماشین مجازی دوم گزینه reboot را می زنیم. دوباره تنظیمات مربوط به پیکر بندی شبکه را روی این ماشین مجازی انجام می دهیم. و ماشین مجازی را با استفاده از دستور shutdown -r now مجدداً start می کنیم. یک نسخه دیگر از putty را باز می کنیم. تا به ماشین مجازی جدیدمان وصل شویم.

اولین کار برای نصب تکنولوژی های وب باید قابلیت linux SE را روی ماشین مجازی غیر فعال کنیم. برای غیر فعال کردن SE Linux نیاز داریم که فایل Configuration را تغییر دهیم.

برای این کار با استفاده از دستور vi آن را باز می کنیم :

vi /etc / selinux / config

همانطور که می بینیم فایل selinux باز شد ، مقدار SELINUX= enforcing را به مقدار disable تغییر داده و سپس فایل را save می کنیم. سپس با استفاده از دستور shutdown-rnow ماشین مجازی را reboot می کنیم. حال ماشین مجازی ما قابل استفاده است که با استفاده از putty به آن connect می شویم. حال چک می کنیم که ماشین مجازی به اینترنت وصل شده باشد.

آپاچی هدوپ؛ قابلیت گسترش پذیری به بیش از صدها و هزاران ماشین را دارد و داده های حجیم و پردازشها روی شبکه ای از کامپیوترها به صورت کارآمد توزیع میگرددند. شرکتهایی بزرگی همچون google, facebook, twitter, yahoo, IBM,..... از هدوپ استفاده می کنند. مزایای هدوپ عبارتند از

- عدم نیاز به کامپیوترهای سریع و قدرتمند
- شناسایی کامپیوترهای معیوب در حین پردازش و مدیریت پردازشها
- ارسال محاسبات به سمت کامپیوتر حاوی داده
- قابلیت گسترش و توسعه آسان کلاسترها و ...

۳. پیاده سازی هدوپ بوسیله کلودرا [9]

این پیاده سازی استفاده از تکنولوژی Hadoop برای cluster کردن چندین سیستم تحت سیستم عامل linux در محیط مجازی ساز Vmware Workstation و به صورت محلی را انجام میدهد.

برای نصب این تکنولوژی تمامی سیستم عامل هایی که قرار است تکنولوژی Hadoop روی آنها نصب و پیاده سازی شوند نیاز دارند که از سیستم عامل ۶۴ بیتی استفاده کنند. برای انجام این کار نسخه ای از linux به نام Centos انتخاب شده است که دارای نسخه ۶۴ بیتی می باشد، این سیستم عامل بر روی محیط Vmware نصب میگردد و در نهایت Hadoop روی این سیستم عامل نصب و پیاده سازی می شود. بعد از به پایان رسیدن نصب CentOS با زدن گزینه reboot ماشین مجازی ما restart می شود.

اولین کاری که باید انجام دهیم تنظیمات شبکه ماشین مجازی است. به linux وصل شده و با root user connect, میشود. ابتدا باید ماشین مجازی را به شبکه اینترنت وصل کنیم تا بتوانیم فایل های مورد نیاز را download کنیم.

برای پیکر بندی شبکه با استفاده از دستورات، فایلی که پیکر بندی شبکه روی آن است را باید تغییر دهیم. فایل /etc/sysconfig/network-scripts/ifcfg-eth0 مشخص کننده اولین کارت شبکه نصب شده روی سیستم است. با تغییر configuration آن می توانیم به آن IP وصل شده و مشخصات

مجازی را وارد کنیم، هر IP داخل یک خط جدا وارد می شود، و خود مدیریت به ماشین ها متصل می شود.

بعد از اضافه کردن تمامی ماشین ها گزینه continue را میزنیم. در این قسمت نیاز داریم username و password ssh برای login به محیط مدیریت را به قسمت manager بدهیم که با استفاده از آن به hostها و ماشین های دیگر متصل شود و تضمینات و برنامه های مورد نظرش را روی آن نصب کند.

به طور پیش فرض از user root استفاده می کند. و در پایین ماشین هایی که قبلاً آنها را وارد کرده ایم می دهیم. این password و user name روی کلاینت قرار دارد. برای اتصال سرور به client میباشد، بر روی start installation کلیک می کنیم. خود این محیط به صورت اتوماتیک به ماشین های دیگر متصل شده و با استفاده از اینترنت نرم افزار های مورد نیازش را download، و بر روی آن نصب می کند.

بعد از download حدود یک GB اطلاعات از اینترنت ماشین client ما تمامی نرم افزار های مورد نیاز برای پیکر بندی و اجرا شدن Hadoop توسط cloudera manager بر روی آن دانلود شد و نصب شد. در ادامه بر روی continue کلیک می کنیم.

سرویس هایی که می خواهیم بر روی cluster اجرا شود از لیست مربوطه شامل HDFS (Hadoop distribution file) Map reduce و... با استفاده از core Hadoop انتخاب و نصب می شود. پس از نصب Hadoop می توان یک file zip برای تنظیمات client را دریافت کرد. همینطور در قسمت administrator می توانیم یک سری تنظیماتی مانند تغییر password، logهایی که برای شبکه clustery انجام داد.

SCP (Secure Copy) فرمانی است در Linux که برای کپی کردن فایل ها بین ماشین ها از آن استفاده می شود، که برای نصب Hadoop مورد نیاز است. در سیستم های client این فرمان باید نصب باشد. با استفاده از دستور Yum install openssh-client فرمان مورد نظر نصب می گردد [9].

برای نصب Hadoop از نرم افزار cloudera استفاده می کنیم که فقط بر روی سیستم عامل linux نصب می شود. این نرم افزار از سایت www.cloudera.com قابل دانلود میباشد. پس از نصب نرم افزار cloudera manager باید به فایل مورد نظر زیر مجوز اجرا صادر شود:

```
Chmod u+x cloudera-manager-installer.bin
```

و سپس با دستور زیر اجرا می شود:

```
./cloudera-manager-installer.bin
```

بدین صورت وارد محیط نصب cloudera می شویم. که محیط گرافیکی است خود installer به اینترنت متصل شده و فایل های مورد نیاز روی cloudera manager را download میکند و بر روی ماشین مورد نظر ما نصب می کند. پس از نصب cloudera manager برای استفاده بر روی server آماده است. حال برای مدیریت cloudera manager نیاز به محیط تحت وب داریم، وارد مرورگر می شویم و به محیط مدیریتی cloudera متصل می شویم. برای این کار نیاز داریم که IP ماشینی که cloudera manager روی آن نصب شده را بنویسیم و از پورت 7180 به آن محیط مدیریتی دسترسی پیدا کنیم. اگر از مرورگرمان نتوانستیم به یک محیط مدیریتی متصل شویم دلیل آن این است که firewall linux این اجازه را به ما نمیدهد بنا بر این در ماشین server با استفاده از دستور Service iptables stop دیواره آتش linux را غیر فعال می کنیم.

برای ورود به محیط مدیریتی user name و password پیش فرض admin-admin است. وارد کرده و login را می زنیم. وارد محیط مدیریتی شده و در این قسمت بر روی گزینه continue کلیک می کنیم. در این قسمت مشخص کننده host که می خواهیم محیط مدیریتی آنها را مدیریت کند. نسخه free edition قابلیت ساپورت ۵۰ عدد سیستم را دارد. در صورتی که نیاز به بیش از ۵۰ سیستم باشد لازم به خریداری license می باشد در پایین این صفحه مدیریتی IP.add new host to cluster سیستم هایی که می خواهیم محیط مدیریتی آنها را مدیریت کند برای clustering وارد می کنیم. برای این کار یک linux به اسم client CentOS داشتیم که IP آن را دارد کرده و بر روی گزینه find host کلیک می کنیم.

میبینیم که ماشین ما در قسمت بالا اضافه شده و آماده پیکر بندی است در این قسمت ما میتوانیم چندین IP ماشین

چارچوب p2p Map Reduce را معرفی می کنیم که از مدل P2P برای مدیریت مشارکت نودهای نوبت دار ، نقایص مستر ، و بازیابی کار Map Reduce به روش غیر متمرکز اما موثر ، استفاده می کند.

هدف از چارچوب P2P Map Reduce دو چیز است. (۱) اداره کردن نقایص مستر به وسیله تکرار وضعیت کار روی یک مجموعه از مسترهای پشتیبان (۲) حمایت از کاربردهای Map Reduce روی شبکه های پویا که این شبکه های پویا از نودهایی ایجاد شده اند که به هم متصل می شوند و سیستم را با نرخ غیر قابل پیش بینی ترک می کنند [6]

۶. معماری موازی (Parallel Databases)

معماری موازی به نوعی از معماری گفتم می شود که در این نوع معماری سعی شده است که با پیاده سازی روش های کار موازی کارایی سیستم را افزایش دهیم. پردازش های موازی می تواند در بخش های مختلف و وظایف مختلف از قبیل خواندن داده ها، تولید فهرست ها و پردازش پرس و جو ها پیاده سازی شود. همچنین در این نوع سیستم ها داده ها می توانند از لحاظ قرار گیری فیزیکی در قسمت های مختلف قرار گرفته باشند. ایده اصلی در معماری موازی استفاده از پردازش های موازی در مراحل مختلف تا آنجا که امکان دارد برای بهبود کارایی سیستم است [4].

۷. معماری ترکیبی مورد استفاده

همانگونه که قبلا گفته شد، هدف اصلی در این مطالعه، طراحی معماری سیستمی که از بهترین ویژگیهای هر دو رویکرد استفاده کند. به این معنی که از مکانیزم های پایگاه داده های موازی برای کارایی و بهره وری بالاتر استفاده کند و از ویژگیهای سیستم های مبتنی بر Map Reduce برای مقیاس پذیری، تحمل خطا و انعطاف پذیری بالا بهره بگیرد.

ایده اصلی در این سیستم، استفاده از Map Reduce به عنوان لایه ارتباطی و هماهنگ کننده فعالیت ها بین چندین گره می باشد که برای این منظور از Hadoop که یک نمونه پیاده سازی شده معروف بر اساس روش Map Reduce میباشد،

۴. نگاشت کاهش (Map Reduce) [3]

یک مدل برنامه نویسی موازی جهت پردازش داده های روی کلاسترها می باشد که از دو فاز اصلی شامل فاز نگاشت (map) و فاز کاهش (reduce) تشکیل شده است.

فاز اول Map: نود Master ورودی را دریافت می کند و آن را به قسمت های کوچکتر تبدیل می نماید و آن ها را بین نودهای worker تقسیم می کند. نود worker خود ممکن است این مرحله را تکرار کند و در نتیجه یک ساختار درختی به وجود می آید.

فاز Reduce: نود Master جواب قسمت های کوچک شده را دریافت می کند و آنها را با یکدیگر ترکیب می کند تا خروجی مورد نظر تشکیل شود.

برای پیاده سازی این دو مرحله نیاز به دو توابع Map و Reduce میباشد. این روش، یک مدل برنامه نویسی ساده است که برای حل مسائل محاسباتی در مقیاس وسیع و نیز به صورت توزیعی، مورد استفاده قرار می گیرد. و بستری امن و مقیاس پذیر برای توسعه کاربردهای توزیعی فراهم میکند.

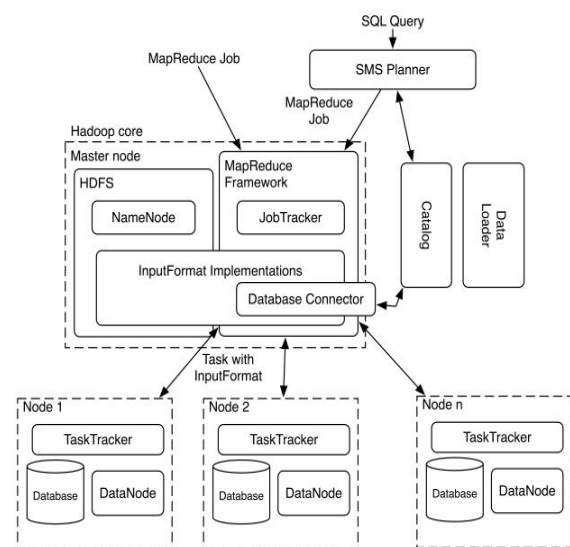
پیاده سازیهای Map Reduce مدل Master/Slave هستند. بدین مفهوم که گره کاربر یک کار را به گره سرویس دهنده می دهد و گره سرویس دهنده چند تا از کارگران بیکار را انتخاب می کند و به هر کدام از آنها یک وظیفه map یا reduce را محول می کند. وقتی تمام وظایف map و reduce کامل شدند ، گره سرویس دهنده نتیجه را به گره کاربر بر می گرداند. اگر کارگری دچار نقص شود ، وظیفه اش را مجددا کارگر دیگری اجرا می کند. اما پیاده سازیهای جاری Map Reduce از عهده سرویس دهنده های معیوب بر نمی آیند.

۵. P2P Map Reduce [5]

هدف این است که ببینیم چگونه می توان معماری master slave ای را که در پیاده سازیهای Map Reduce استفاده می شود ، اصلاح کرد ، به گونه ای که برای سناریوهای ابری پویا مانند Interclouds مناسب تر باشد

استفاده کرده است. از طرفی هر گره از یک پایگاه داده DBMS عادی استفاده می کند. بدین جهت به این روش ترکیبی، HadoopDB گفته میشود [7].

از آنجا که HadoopDB از Hadoop استفاده میکند، معماری کلی آن مشابه Hadoop است، با این تفاوت که پیمانه هایی به آن اضافه شده است. شکل زیر معماری سیستم HadoopDB را با جزئیات آن نمایش میدهد:



شکل ۱ ساختار HadoopDB [7]

۸. نتیجه گیری

در مسئله چگونگی انجام عملیات محاسباتی و تحلیلی بر روی حجم عظیمی از داده ها که بین هزاران گره توزیع شده اند دو رویکرد کلی مطرح شد. در رویکرد اول، Map Reduce یک مدل برنامه نویسی ساده برای حل مسائل مختلف محاسباتی در مقیاس وسیع بوده و به صورت توزیعی مورد استفاده قرار میگیرد. در رویکرد معماری موازی ایده اصلی بر اساس استفاده از پردازش موازی در مراحل مختلف برای بهبود کارایی سیستم است. هدف اصلی این تحقیق، طراحی معماری سیستمی که از بهترین ویژگیهای هر دو رویکرد استفاده کند. در این روش، از مکانیزم های پایگاه داده های موازی برای کارایی و بهره وری بالاتر استفاده گردید و از ویژگیهای سیستم های مبتنی بر Map Reduce برای مقیاس پذیری، تحمل خطا و انعطاف پذیری بالا بهره گرفته شد. در این معماری (HadoopDB) از Map Reduce به عنوان لایه ارتباطی و هماهنگ کننده فعالیت ها بین گره ها استفاده شد از طرفی هر گره از یک DBMS عادی استفاده می کنند. با مقایسه نتایج مشاهده شده و مقایسه با نتایج آزمایش ها مرجع [4] نتیجه میشود که سیستم پیشنهادی، دارای مقیاس پذیری بالاتر نسبت به parallel databases می باشد همچنین توانایی تحمل خطا در حد Hadoop را دارد و از طرفی کارایی در حد parallel databases را نیز برآورده میکند.

سپاسگزاری

اینجانب پروین احمدی بر خود لازم می دانم از جناب آقای دکتر محمد رضا سلطان آقایی و آقای دکتر فرامرز صافی که صبورانه راهنمای من بوده و یاریام نمودند تشکر نمایم.

مراجع

- [1] Survey on Google File System Naushad UzZaman. Text on Website of Google, www.Scholar.Google.com
- [2] Pavlo, A. and Paulson, E. and Rasin, A. and Abadi, D.J. and DeWitt, D.J. and Madden, S. and Stonebraker, M., A comparison of approaches to large-scale data analysis, In Proceedings of the 35th SIGMOD International Conference on Management of Data, pp165-178, 2009, ACM.

که ترکیب شده است از پایگاه داده هایی که در هر یک از گره ها قرار دارد و پیمانه های اضافه شده که شامل موارد زیر میباشند:

- Database Connector: توابع لازم برای اتصال به پایگاه داده های مختلف، همانند JDBC و ODBC و غیره.
- Catalog: در این پیمانه، اطلاعات مربوط به توصیف پایگاه داده ها نگهداری میشود.
- SMS Planner: این پیمانه که نام کامل آن SQL to Map Reduce to SQL میباشد، وظیفه تبدیل درخواستهای ورودی با فرمت SQL به کارهای Map Reduce را بر عهده دارد.

- [3] Stonebraker, M. , Abadi, D. , DeWitt, D.J. , Madden, S., Paulson, E., Pavlo, A., and Rasin, A., MapReduce and parallel DBMSs: friends or foes?, Communications of the ACM, 53(1), pp 64-71, 2010.
- [4] Azza Abouzeid, Kamil BajdaPawlikowski, Daniel Abadi, Avi Silberschatz, Alexander Rasin, HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads, Yale University.
- [5] Jeffrey Dean and Sanjay Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, OSDI, 2004.
- [6] Nick Antonopoulos, Cloud Computing Principles, Systems and Applications, part II, chapter 7, Lee Gillam, Peer-to-Peer Cloud Provisioning: Service Discovery and Load-Balancing, Professor A.J. Sammes, BSc, MPhil, 2010.
- [7] Dhruba Borthakur, The Hadoop Distributed File System: Architecture and Design, The Apache Software Foundation, 2007.
- [8] HDFS Java API:
<http://hadoop.apache.org/core/docs/current/api/>
- [9] Website: <http://www.cloudera.com>
- [10] Cloudera Introduction to Apache Hadoop