IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 5, May 2017

HIVELytics: Building Execution Profiles for the Systems Running on HIVE

Pratheeksha P[1], Paramesh R[2]

M. Tech, CSE, GAT, Bengaluru, India[1]

Associate Professor, CSE, GAT, Bengaluru, India[2]

Abstract: Apache Hive could be a wide used information reposition and analysis tool. Developers write SQL like HIVE queries that are regenerate into MapReduce programs to runs on a cluster. Despite its quality, there's very little analysis on performance comparison and diagnose. Part of the explanation is that instrumentation techniques accustomed monitor execution cannot be applied to intermediate MapReduce code generated from Hive question. as a result of the generated MapReduce code is hidden from developers, run time logs are the sole places a developer will get a glimpse of the particular execution. Having an automatic tool to extract info and to come up with report from logs is crucial to know the query execution behavior. The designed a tool to make the execution profile of individual Hive queries by extracting info from HIVE and Hadoop logs. The profile consists of elaborated info regarding MapReduce jobs, tasks and tries happiness to a question. it's keep as a JSON document in MongoDB and might be retrieved to come up with reports in charts or tables. I have run many experiments on AWS with TPC-H knowledge sets and queries to demonstrate that our identification tool is ready to help developers in examination HIVE queries written in numerous formats, running on completely different knowledge sets and organized with different parameters. it's additionally ready to compare tasks/attempts inside constant job to diagnose performance problems.

Keywords: MapReduce, HIVE, Hadoop, JSON.

I. INTRODUCTION

Increasing users in the present social networking sites drop their opinions about the particular site or about the services of the networking sites. This increases the importance of the networking sites in terms of customer satisfaction and transparent behaviour between the network and the user. With help of the present technology users are able to express their positive or negative feedback about the any particular object or service on the internet just by adding a comment to it. Such generated data will help in the improvisation of the services or products by the business holders and to interact with minimal efforts with users. Such exclusive importance has increased the tracking and analysis module for social networking comments that includes the positive or negative sentimental expressions of users. The demand of sentimental text analysis is increased and has taken a step in industrial as well as academic field with betterment of the performance. Eventually the trial and error modules are implemented so that computer can identify the emotions expressed by humans through text format [01].

II.LITERATURE SURVEY

Apache Hive is a popular data analytic extension to HadoopMapReduce framework. The SQL like language makes it a convenient alternative to hand-coded MapReduce programs. Simple HIVE queries are able to achieve similar performance as those of hand-coded MapReduce programs. There could be performance gaps between HIVE query and hand-coded MapReduce program for complex analytic workloads[16].

Over the years, HIVE has made many significant technical advancements in storage format, indexing, SQLto-MapReduce translator and execution engine to improve execution performance. Certain in-depth knowledge may be required to fully utilize such advancements. The performance of complex queries could be unpredictable and sometimes hard to understand for most users. There are cases where small changes in the way a query is written may have big impact on the performance[6].

There are also cases where performance on large data set may vary a lot to performance on small data set using exactly the same query. In addition, HIVE also allows for fine tuning performance by parameters such as reducer number and timing of shuffle. The effect of such parameters, most of the time, can only be evaluated by experimenting. Developers often have to resort to meticulously walking through countless log messages, or perform numerous key word searches to find the information needed[1].

Performance and debugging information may be obtained by instrumenting query execution. This approach is intrusive and often incurs overheads. In addition, these options are only available for hand-coded MapReduce programs. HIVE

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 5, May 2017

generates and executes MapReduce code on the fly, making it impossible for any instrumentation tools to perform source or binary instrumentation.
Most clusters are configured to allow jobs running at the same time sharing the computing resources fairly. Hadoop cluster configured with MRv1 view resources as fixed map or reduce slots on each node. Job scheduling and monitoring is managed by a JobTracker demon running on the master node. TaskTracker demons running on the slave nodes manage the execution of individual tasks. This is considered as ineffective because the JobTracker demon is responsible for managing the resources for the whole cluster as well as monitoring the execution of each individual jobs. Recent versions of the Hadoop introduce a generic resource management framework YARN. Hadoop cluster configured with YARN is also referred to as MRv2. In this setting, YARN is responsible for allocating and managing resources for the whole cluster. Each application running on the cluster has its own ApplicationMaster to monitor execution status and to decide if new attempt should be started.
The run time logs under different managing systems contain similar information with slightly different formats. We focus on logs generated by YARN. YARN framework allows users to configure a log directory in HDFS. All logs of a particular job will be stored in sub-directories identified by user name and jobID. Each attempt is stored in a file identified by the actual node running the attempt. The logs representing the overall job is stored in a file identified by the node running the ApplicationMaster.

III.PROPOSED WORK

In this paper, the present a profiling tool which builds execution profile of individual HIVE queries with information extracted from log files. The profiled is stored in semi structured format and can be retrieved to produce various reports for performance comparison, diagnose and other use. The profiler is a specialized one extracting data from logs produced by running queries. Different from all the existing profilers, focus on profiling individual query to understand performance variation and to diagnose possible problems.

Advantages of proposed system

- profiler focuses on profiling each and every individual query thus giving a developer in depth freedom of analysis.
- It also provides visualization tools to help developers show details of a particular query, job or task.
- profiler has the ability to compare profiles of multiple queries against each other on various aspects.

3.1 Purpose

The purpose of this document is to provide Software Requirement Specification for "Building execution profiles for the systems running on HIVE".

3.2 Scope

The software product produced is an application by name "Building execution profiles for the systems running on HIVE". Apache Hive is a widely used data warehousing and analysis tool. Developers write SQL like HIVE queries, which are converted into MapReduce programs to runs on a cluster. Despite its popularity, there is little research on performance comparison and diagnose. Part of the reason is that instrumentation techniques used to monitor execution cannot be applied to intermediate MapReduce code generated from Hive query. Because the generated MapReduce code is hidden from developers, run time logs are the only places a developer can get a glimpse of the actual execution. Having an automatic tool to extract information and to generate report from logs is essential to understand the query execution behavior [16].
The designed a tool to build the execution profile of individual Hive queries by extracting information from HIVE and Hadoop logs. The profile consists of detailed information about MapReduce jobs, tasks and attempts belonging to a query. It is stored as a JSON document in MongoDB and can be retrieved to generate reports in charts or tables. We have run several experiments on AWS with TPC-H data sets and queries to demonstrate that our profiling tool is able to assist developers in comparing HIVE queries written in different formats, running on different data sets and configured with different parameters. It is also able to compare tasks/attempts within the same job to diagnose performance issues [16]. In this paper, present a profiling tool which builds execution profile of individual HIVE queries with information extracted from log files. The profiled is stored in semi structured format and can be retrieved to produce various reports for performance comparison, diagnose and other use. Profile is a specialized one extracting data from logs produced by running queries. Different from all the existing profilers, and focus on profiling individual query to understand performance variation and to diagnose possible problems.

3.3 SYSTEM DESIGN

Systems style is that the method of process the design, components, modules, interfaces, and knowledge for a system to satisfy such that needs. Systems style may see it because the application of systems theory to development. there's some

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 5, May 2017

overlap with the disciplines of analytic thinking, systems design and systems engineering. If the broader topic of development "blends the angle of selling, design, and producing into one approach to development," then style is that the act of taking the selling info and making the look of the merchandise to be manufactured. Systems style is thus the method of shaping and developing systems to satisfy specific needs of the user.
Until the 1990 systems style had an important and revered role within the processing business. within the 1990 standardization of hardware and code resulted within the ability to make standard systems. The increasing importance of code running on generic platforms has increased the discipline of code engineering. Object-oriented analysis design and style are getting the foremost wide used ways for laptop systems design. The UML has become the quality language in object-oriented analysis and style. it's wide used for modeling package systems and is more and more used for prime planning non-software systems and organizations. System style is one in all the foremost vital phases of computer code development method. the aim of the designing the look is to plan the answer of a tangle such by the need documentation. In different words the primary step within the answer to the matter is that the style of the project.
The design of the system is maybe the foremost vital issue moving the standard of the computer code. the target of the style the planning the look part is to provide overall design of the computer code. It aims to work out the modules that ought to be within the system to meet all the system necessities in an economical manner. The design can contain the specification of these modules, their interaction with different modules and therefore the desired output from every module. The output of the look method could be a description of the software package design.

The design part is followed by 2 sub phases
- High Level style
- Detailed Level style

The below figure 2 shows a general block diagram describing the activities performed by this paper.

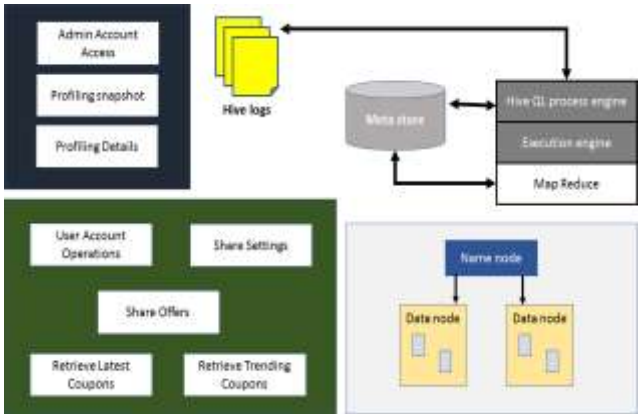

Figure 2: Architectural diagram of proposed work.

The individual blocks are explained as following:
3.3.1 Data Access Layer-Data access layer is the one which exposes all the possible operations on the data base to the outside world. It will contain the DAO classes, DAO interfaces, POJOs, and Utils as the internal components. All the other modules of this project will be communicating with the DAO layer for their data access needs.

3.3.2 Account Operations-Account operations module provides the following functionalities to the end users of our project.
- Register a new seller/ buyer account
- Login to an existing account
- Logout from the session
- Edit the existing Profile
- Change Password for security issues
- Forgot Password and receive the current password over an email
- Delete an existing Account

Account operations module will be re-using the DAO layer to provide the above functionalities.

3.3.3 Coupons Retrieval -Here, the user will be able to retrieve the coupons from various online stores. The service layer will be invoking Rest API call which communicates with various stores online and provides us the coupons information along with the coupon code and the expiry date for retrieving the coupons. The user will have to register /

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 5, May 2017

create their account in the coupons applications before they can retrieve these coupons. The coupons has been categorized into Trending and Latest coupons. Trending coupons are those which will be accessed by most number of users. Latest coupons are those which are recently uploaded by the online stores.

3.3.4 Coupons Sharing-Here, the user will be able to share the coupons they obtained from the previous component with their friends/ relatives. The users will have to provide the email address of the person with whom the user must share the coupons with. The coupons application will be triggering an email to the specified recipient and sends the details like the coupons code, title, description, expiry date, and the link from where the offer can be retrieved from.

3.3.5 Admin Account and Logs Path Configuration-Here, The admin of the coupons application will be able to access the Hive Analysis component by providing his/her access details. The registration module for the Hive Analysis application, because the Hive analysis application is secured and only the admins of the Coupons application must be accessing the Hive analysis application. The admin after logging in to the Hive Analysis application, will be able to configure the path where the hive logs file are present. Hive logs files will be generated upon continued access to Coupons Application.

3.3.6 Hive Analysis-Here, the admin of the coupons application, after logging into the Hive analysis application, will be able to analyze the log files generated by the coupons application. We are providing two sub types of analysis here: Dashboard and Details.In the dashboard page, the admin will be viewing the summary of the logs files generated, like total log files found, total number of successful queries, total number of failure queries, query which took maximum time, and the query which took minimum time. In the details page, the admin will be able to access the profile of individual hive query in detail. Each query will have its own profile. The profile information includes command, command type, start time, end time, total time, result, and the error details if any.

IV. CONCLUSION

The presented a HIVE profiling tool based on log analysis. This profiler is able to extract information from various log files to build profiles of individual queries. It also provides visualization tools to help developers show details of a particular query, job or task and to compare profiles of multiple queries against each other on various aspects. Our experiment shows that it can be used in performance analysis, diagnose and parameter selection. Though not presented in the experiment, our profiler can be a useful tool to test the impact of new software features. It can effectively replace the hand drawn charts and tables in reports of new feature and new improvements.

REFERENCES

[1] hive. [Online]. Available: http://hive.apache.org/Apache
[2] Apache hadoop. [Online]. Available: http://hadoop.apache.org/
[3] Apache hadoopnextgenmapreduce (yarn). [Online]. Available: http://hadoop.apache.org/docs/r2.2.0/hadoop-yarn/hadoop-yarnsite/ YARN.html
[4] Tpc-h benchmark. [Online]. Available: http://www.tpc.org/tpch/
[5] M. Poess and C. Floyd, "New tpc benchmarks for decision support and web commerce," SIGMOD Rec., vol. 29, no. 4, pp. 64–71, Dec. 2000. [Online]. Available: http://doi.acm.org/10.1145/369275.369291
[6] R. Lee, T. Luo, Y. Huai, F. Wang, Y. He, and X. Zhang, "Ysmart: Yet another sql-to-mapreduce translator," in Distributed Computing Systems (ICDCS), 2011 31st International Conference on. IEEE, 2011, pp. 25– 36.
[7] Y. Huai, A. Chauhan, A. Gates, G. Hagleitner, E. N. Hanson, O. O'Malley, J. Pandey, Y. Yuan, R. Lee, and X. Zhang, "Major technical advancements in apache hive," in Proceedings of the 2014 ACM SIGMOD international conference on Management of data. ACM, 2014, pp. 1235–1246.
[8] A.Srivastava and A. Eustace, ATOM: A system for building customized program analysis tools. ACM, 1994, vol. 29, no. 6.
[9] Q. Gao, F. Qin, and D. K. Panda, "Dmtracker: finding bugs in largescale parallel programs by detecting anomaly in data movements," in Proceedings of the 2007 ACM/IEEE conference on Supercomputing. ACM, 2007, p. 15.
[10] H. Herodotou and S. Babu, "Profiling, what-if analysis, and costbased optimization of mapreduce programs," Proceedings of the VLDB Endowment, vol. 4, no. 11, pp. 1111–1122, 2011.
[11] Btrace: A dynamic instrumentation tool for java. [Online]. Available: https://kenai.com/projects/btrace
[12] X. Zhao, Y. Zhang, D. Lion, M. Faizan, Y. Luo, D. Yuan, and M. Stumm, "lprof: A nonintrusive request flow profiler for distributed systems," in Proceedings of the 11th Symposium on Operating Systems Design and Implementation, 2014.
[13] P. Barham, R. Isaacs, R. Mortier, and D. Narayanan, "Magpie: Online modelling and performance-aware systems." in HotOS, 2003, pp. 85–90.
[14] R. Fonseca, G. Porter, R. H. Katz, S. Shenker, and I. Stoica, "X-trace: A pervasive network tracing framework," in In NSDI, 2007.
[15] R. R. Sambasivan, A. X. Zheng, M. De Rosa, E. Krevat, S. Whitman, M. Stroucken, W. Wang, L. Xu, and G. R. Ganger, "Diagnosing performance changes by comparing request flows." in NSDI, 2011.
[16] Profiling Apache HIVE Query from Run Time Logs, GivannaPutriHaryono School of Information Technologies The University of Sydney NSW 2008 Email: ghar1821@uni.sydney.edu.au, yingzhou School of Information Technologies The University of Sydney NSW 2008 Email: ying.zhou@sydney.edu.au978-1-4673-8796-5/16/$31.00 2016 IEEE

BIOGRAPHIES

Pratheeksha P M. Tech CSE, GAT Department of Computer Science and Engineering, RR Nagar, Bengaluru.

Paramesh R Associate Professor, GAT Department of Computer Science and Engineering, RR Ngar, Bengaluru.