Análise e caracterização das novas ferramentas para computação distribuída na nuvem

Otávio M. de Carvalho, Eduardo Roloff, Marco A. Z. Alves, Philippe O. A. Navaux
Universidade Federal do Rio Grande do Sul
Grupo de Processamento Paralelo e Distribuído
{*omcarvalho,eroloff,mazalves,navaux*}*@inf.ufrgs.br*

***Resumo*—No cenário atual de computação em nuvem o processamento de grandes volumes de dados tem sido dominado pelo modelo de programação MapReduce, originalmente proposto pela Google e largamente adotado através da implementação Apache Hadoop. Ao longo do desenvolvimento e da evolução desse modelo foram identificados deficiência em sua implementação, relacionadas às garantias de confiabilidade sobre os dados e com a existência de certos tipos de processamento de dados, para os quais o modelo de processamento não se encaixa adequadamente. A partir da percepção dessas limitações, inúmeras iniciativas foram desenvolvidas, com o objetivo de estender ou mesmo de substituir o modelo. Neste trabalho, as principais iniciativas para a evolução do modelo foram pesquisadas, representando o estado da arte das ferramentas existentes, e foi proposta uma classificação dessas ferramentas de acordo com suas características. Essa classificação global tem seu foco principal na relação entre o tempo de processamento e o volume de dados processados, buscando detalhar as diferenças nesse conjunto de ferramentas.**

I. Introdução

A iniciativa principal do processamento online de grandes volumes de dados, caracterizado pelo conceito atual de *Big Data* [1], partiu principalmente dos trabalhos iniciais da Google, com a publicação dos seus artigos sobre o *Google File System* [2] e o *MapReduce* [3].

A partir desses trabalhos, foi desenvolvido o projeto *Apache Hadoop* [4], que propôs novas versões dessas abordagens, baseadas em software livre. Esse projeto foi inicialmente desenvolvido por engenheiros do Yahoo e posteriormente entregue à fundação Apache. Desde então, esse projeto cresceu e se tornou o modelo predominante para o processamento de grandes volumes de dados.

Grande parte da adoção destas ferramentas se deve também a outro trabalho da Google, chamado *Google BigTable* [5]. Este trabalho tornou possível o desenvolvimento de ferramentas de armazenamento distribuído de dados operando sobre a infraestrutura do Hadoop. A partir desse trabalho, surgiram ferramentas como o *Facebook Cassandra* [6], que deram origem ao denominado segmento dos bancos de dados NoSQL.

Ainda que o *MapReduce* tenha atingido a sua meta principal de proporcionar alta escalabilidade no processamento distribuído de dados, e desenvolvido a estrutura básica para que uma comunidade crescesse em torno das suas idéias, logo foram percebidos os gargalos que sua infraestrutura pode apresentar [7].

A partir da percepção das limitações do modelo *MapReduce*, surgiram diversas alternativas, focando cada vez mais no tipo de dado a ser processado. Desta forma, três conjuntos de aplicações surgiram: Aplicações que focam no processamento sequencial; Aplicações que realizam consultas interativas sobre grandes conjuntos de dados, e; Aplicações voltadas ao processamento de fluxos contínuos de dados.

Percebendo as constantes evoluções que estão sendo propostas ao modelo *MapReduce*, esse artigo tem como objetivo apresentar as principais ferramentas que representam o estado da arte das propostas de extensão do modelo *MapReduce*. Sugerimos também uma nova abordagem para a caracterização destas propostas, através da relação entre o tempo de processamento e o volume de dados a serem processados.

Este trabalho está organizado da seguinte forma: Na Seção II são discutidas as propostas de agrupamento existentes para caracterizar as ferramentas que estendem o *MapReduce*. A Seção III descreve a nova proposta de agrupamento. Por fim, a Seção IV traz conclusões do trabalho e sugestões de possíveis trabalhos futuros.

II. Discussão e Desafios

Logo após o surgimento da diversa gama de ferramentas que complementam o modelo *MapReduce*, surgiram inúmeras abordagens para caracterizar essas novas aplicações, que se voltam, dentre outras características, para a forma como os dados são consultados, dividindo-os principalmente em 2 grupos: NoSQL (aplicações que processam e armazenam dados de forma distribuída, sem esquemas clássicos de bancos de dados e sem linguagem de consulta SQL) [8] e NewSQL (aplicações que processam e armazenam dados de forma distribuída, com esquemas clássicos de bancos de dados e com sistemas de consulta SQL) [9].

A grande dificuldade deste tipo de classificação, é que ela desconsidera os objetivos para os quais cada uma dessas ferramentas se aplicam. Além disso, grande parte das ferramentas existentes atualmente implementam os dois paradigmas (NoSQL e NewSQL), permitindo a consulta sobre os dados de forma semelhante ao SQL ou no formato das consultas do modelo *MapReduce*.

São perceptíveis também as diferenças na arquitetura destas ferramentas, dadas pela existência de grupos representativos, que podemos classificar como: RDBMSs (*Relational Database Management Systems*) distribuídos, IMDB (*In-Memory Databases*) e aquelas abordagens que utilizam o Hadoop como arquitetura para extensão.

Através do levantamento realizado sobre as ferramentas existentes, percebemos que o grande diferencial entre essas tecnologias é a forma como elas se relacionam com os dados através do tempo. Isto é, a relação entre a velocidade e quantidade de dados das consultas que elas processam. Desta forma, propomos um agrupamento dessas abordagens, caracterizando-as em três grandes grupos: Processamento Batch, Processamento Interativo e Processamento em Tempo Real.

III. Caracterização das Iniciativas de Extensão do Modelo MapReduce

Após o impulso inicial, caracterizado principalmente pelo desenvolvimento das ferramentas utilizadas pela Google para a construção das suas aplicações, foi desenvolvida uma gama extensa de ferramentas abordando o processamento distribuído na nuvem. Estas ferramentas fornecem desde funcionalidades para o processamento de grandes volumes de dados, até abstrações para o processamento de fluxos de dados em tempo real.

A seguir, descrevemos os grandes grupos utilizados para caracterizar essas ferramentas, onde os grupos levam em consideração a velocidade do processamento com relação ao tamanho médio do conjunto de dados sobre os quais elas são aplicadas.

A. Processamento Batch

O Processamento Batch consiste no conjunto de aplicações que tem por objetivo o processamento de grandes volumes de dados de forma sequencial, sem que ocorra nenhuma intervenção durante o processamento.

Representado pelo Apache Hadoop e pelas ferramentas comerciais que surgiram a partir dele, o grupo possui uma intersecção com os sistemas de MPP (*Massively Parallel Processing*), que normalmente também executam tarefas de processamento batch por trás de suas arquiteturas.

B. Processamento Interativo

O Processamento interativo é caracterizado pela necessidade de processar um tamanho intermediário de dados, uma vez que as plataformas dessa categoria operam sobre dados em sistemas de armazenamento distribuídos.

Esta categoria inclui os sistemas de armazenamento de dados distribuídos, incluindo os bancos de dados MPP, os bancos de dados NoSQL e os mais recentes bancos de dados NewSQL.

O avanço destas ferramentas é dirigido principalmente pela incorporação da confiabilidade dos sistemas de banco de dados, e do estado da arte na pesquisa em sistemas de banco de dados.

C. Processamento em Tempo Real

O Processamento em Tempo Real visa suprir as novas necessidades de processamento, uma vez que o volume requisições e dados sendo produzidos é tão grande que, é mais interessante analisá-los como um fluxo contínuo de dados, do que depender de sistemas de armazenamento para realização de análises posteriores.

Após as tentativas iniciais de aplicar esse paradigma sobre o Hadoop, tendo-se percebido que essa abordagem não era eficiente para certos tipos de processamento, foram desenvolvidas novas ferramentas. Estas, são caracterizadas principalmente por iniciativas de CEP (*Complex Event Processing*) [37] que, por possuirem uma arquitetura mais simples e trabalharem com conjuntos de dados menores, prometem performances superiores às das arquiteturas baseadas no Hadoop.

D. Discussão Sobre as Iniciativas

Na Tabela I estão listadas as ferramentas que representam o estado da arte nas evoluções sobre o modelo de processamento *MapReduce*, em conjunto com uma breve descrição das suas funcionalidades e para que tipo de processamento de dados tais ferramentas apresentam maior potencial de processamento.

Nota-se na Tabela I uma tendência de desenvolvimento de ferramentas para o processamento em tempo real, representado pelas ferramentas: Apache S4, Apache Kafka, Apache Storm, Apache Flume, StreamBase CEP e Google Photon. A presença do StreamBase CEP no conjunto das ferramentas é importante pois demonstra que mesmo ferramentas comerciais, não baseadas no Hadoop, hoje oferecem integração com o Hadoop.

Todas as grandes distribuições baseadas no Hadoop, que visam fornecer mecanismos completos de *data warehousing* para concorrer com as ferramentas de MPP, oferecem também o Apache Flume, além de ferramentas para operarem como bancos de dados distribuídos. Por esse motivo, certas ferramentas são marcadas na Tabela I como pertencentes aos 3 segmentos. São elas: Hortonworks HDP, MapR M5, Cloudera CDH e Pivotal HD.

O Apache YARN e o AMPLab BDAS, apesar de abrangerem os 3 segmentos, representam abordagens diferentes. O Apache YARN consiste na evolução do Hadoop, desacoplando certas partes da infraestrutura do Hadoop e permitindo que novos paradigmas rodem diretamente através do sistema, usando apenas partes específicas do mesmo, como o sistema de arquivos ou escalonador. O AMPLab BDAS consiste em uma ferramenta baseada nas inovações do Hadoop, porém com uma arquitetura totalmente reescrita. Essa arquitetura baseia-se em processamento em memória, e desta forma atinge performance superior ao Hadoop, embora forneça ferramentas de análise semelhantes.

Podemos notar também, através da Tabela I, a presença de ferramentas de processamento interativo baseadas em memória, representando uma evolução sobre as ferramentas que operavam sobre o Hadoop, como o Apache Cassandra. Tais ferramentas são representadas por: SAP HANA, VoltDB e AMPLab Spark. Esta mudança é percebida também pelas ferramentas baseadas no Google Dremel, como o Apache Drill e o Cloudera Impala, representando uma busca por formas mais eficientes de se realizar consultas interativas na nuvem.

Além disso, fica claro que grandes bancos de dados MPP estão adotando formas de integração com o Hadoop, como é o caso das seguintes ferramentas: Teradata Aster,

Tabela I
INICIATIVAS DE EXTENSÃO DO MODELO MAPREDUCE E SUAS PRINCIPAIS CARACTERÍSTICAS

Nome	Ano	Descrição	Batch	Interativo	Tempo Real
Teradata Aster [10]	2013	Banco de dados MPP. †	•	•	
Pivotal HD [11]	2013	Conjunto de ferramentas de processamento distribuído. ‡	•	•	•
Google Photon [12]	2013	Sistema para o processamento distribuído de fluxos contínuos de dados.			•
AMPLab BDAS [13]	2012	Conjunto de ferramentas de processamento distribuído em memória. †	•	•	•
Google Spanner [14]	2012	Primeiro banco de dados distribuído com transações externamente consistentes.		•	
Actian ParAccel [15]	2012	Banco de dados MPP. †	•	•	
Cloudera Impala [16]	2012	Sistema para o processamento de consultas interativas. ‡		•	
StreamBase CEP [17]	2012	Ferramenta comercial de processamento complexo de eventos. †			•
Apache Giraph [18]	2012	Ferramenta para o processamento distribuído de grafos. ‡	•		
Apache Drill [19]	2012	Ferramenta para o processamento de consultas interativas.		•	
Apache Flume [20]	2012	Ferramenta para o processamento de fluxos contínuos de dados. ‡			•
Apache YARN [21]	2011	Nova versão de processamento distribuído criada a partir do Apache Hadoop. ‡	•	•	•
SAP HANA [22]	2011	Banco de dados em memória. †		•	
Google Megastore [23]	2011	Banco de dados distribuído que precedeu o Google Spanner.		•	
Apache Storm [24]	2011	Ferramenta para o processamento de eventos complexos.			•
Apache Kafka [25]	2011	Sistema para o processamento de fluxos contínuos de dados. †			•
MapR M5 [26]	2011	Conjunto de ferramentas de processamento distribuído. ‡	•	•	•
Hortonworks HDP [27]	2011	Conjunto de ferramentas de processamento distribuído. ‡	•	•	•
Google Pregel [28]	2010	Sistema distribuído para o processamento de grafos.	•		
Google Percolator [29]	2010	Sistema distribuído para processamento incremental.	•		
Google Dremel [30]	2010	Ferramenta para a análise interativa de dados.		•	
AMPLab Spark [31]	2010	Sistema de processamento de dados distribuído que opera em memória.	•		
VoltDB [32]	2010	Sistema de banco de dados em memória. †		•	
Apache S4 [33]	2010	Ferramenta para o processamento de fluxos contínuos de dados.			•
HP Vertica [34]	2010	Banco de dados MPP. †	•	•	
Apache Hive [35]	2009	Ferramenta para o processamento de consultas interativas. ‡		•	
Cloudera CDH [36]	2009	Conjunto de ferramentas de processamento distribuído. ‡	•	•	•
Apache Cassandra [6]	2009	Sistema de armazenamento de dados distribuído. ‡		•	
Google BigTable [5]	2006	Sistema de armazenamento de dados distribuído.		•	
Apache Hadoop [4]	2005	Sistema de processamento de dados distribuído.	•		
Google MapReduce [3]	2004	Sistema de processamento distribuído que deu origem ao Hadoop.	•		

† Ferramentas que oferecem integração com o Hadoop.
‡ Ferramentas baseadas na infraestrutura do Hadoop.

Actian ParAccel e HP Vertica.

A única ferramenta desenvolvida pela Google que não possui correspondentes (ou extensões) é o Google Spanner. Logo, nenhuma outra ferramenta comercial ou de código aberto fornece um banco de dados com transações externamente consistentes operando em escala global.

IV. CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho mostra que o ambiente de aplicações distribuídas para o processamento na nuvem não limita-se ao Hadoop, e está sendo constantemente estendido por ferramentas que vão além do modelo *MapReduce*.

A proposta de caracterização, nos três grandes grupos sugeridos, facilita o processo de seleção das ferramentas e ajuda a determinar quais apresentam potencial para serem utilizadas por aplicações distribuídas na nuvem.

Entretanto, não é possível afirmar se as implementações atuais convergirão para grandes conjuntos de ferramentas, oferecendo características de processamento diferentes para cada tipo de conjunto de dados e necessidade de tempo de resposta, ou se evoluirão ainda para um conjunto mais heterogêneo de modelos e ferramentas.

O que pode-se notar atualmente é que, a partir do impulso das ferramentas que culminaram na criação do Apache Hadoop, o universo de possibilidades das ferramentas de processamento distribuído passou a contar com uma estrutura sólida para o desenvolvimento de novas abordagens. Devido principalmente à criação de novos sistemas de arquivos distribuídos, como o HDFS [38], da criação de novos sistemas de escalonamento de tarefas, como o Apache Zookeeper [39], e de ferramentas de bancos de dados que suportam novas organizações sobre os dados, como as colunares do Apache Cassandra.

Como trabalhos futuros é sugerida a avaliação do desempenho das ferramentas analisadas ao longo deste trabalho, visando definir quais possuem potencial para posterior utilização em aplicações paralelas na nuvem.

REFERÊNCIAS

[1] A. Jacobs, "The Pathologies of Big Data," *Communications of the ACM*, vol. 52, no. 8, pp. 36–44, 2009.

[2] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The Google File System," in *Int. Conf. SIGOPS Operating Systems Review*, vol. 37. ACM, 2003, pp. 29–43.

[3] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," in *Symp. on Operating System Design and Implementation (OSDI)*, 2004, pp. 137–150.

[4] "Apache Hadoop," http://hadoop.apache.org, acessado em Julho de 2013.

[5] F. Chang, J. Dean, S. Ghemawat *et al.*, "BigTable: A Distributed Storage System For Structured Data," in *USENIX Symp. On Operating Systems Design and Implementation*, vol. 7. USENIX Association, 2006, pp. 15–15.

[6] A. Lakshman and P. Malik, "Cassandra: A Decentralized Structured Storage System," *ACM SIGOPS Operating Systems Review*, vol. 44, no. 2, pp. 35–40, 2010.

[7] A. Pavlo, E. Paulson, A. Rasin, D. J. Abadi, D. J. DeWitt, S. Madden, and M. Stonebraker, "A Comparison of Approaches to Large-Scale Data Analysis," in *SIGMOD Int. Conf. on Management of Data*. ACM, 2009, pp. 165–178.

[8] J. Han, E. Haihong, G. Le, and J. Du, "Survey on NoSQL Database," in *Int. Conf. on Pervasive Computing and Applications*. IEEE, 2011, pp. 363–366.

[9] M. Stonebraker, "New Opportunities for New SQL," *Communications of ACM*, vol. 55, pp. 10–11, 2012.

[10] "Aster SQL-H," http://www.asterdata.com/sqlh/, acessado em Julho de 2013.

[11] "Pivotal HD Enterprise," http://www.gopivotal.com/pivotal-products/pivotal-data-fabric/pivotal-hd, acessado em Julho de 2013.

[12] R. Ananthanarayanan, V. Basker, S. Das, A. Gupta *et al.*, "Photon: Fault-Tolerant and Scalable Joining of Continuous Data Streams," in *SIGMOD Int. Conf. On Management Of Data*, New York, NY, USA, 2013, pp. 577–588.

[13] "AMPLab Berkeley BDAS," https://amplab.cs.berkeley.edu/software/, acessado em Julho de 2013.

[14] J. C. Corbett, J. Dean, Epstein *et al.*, "Spanner: Google's Globally-Distributed Database," in *Proc. of OSDI*, vol. 1, 2012.

[15] "ParAccel Analytic Platform: Platform Overview," http://www.paraccel.com/resources/Datasheets/ParAccel-Analytic-Platform.pdf, acessado em Julho de 2013.

[16] "Introducing Cloudera Impala - SQL-on-Hadoop That's Fashionably Early," http://cloudera.com/content/cloudera/en/campaign/introducing-impala.html, acessado em Julho de 2013.

[17] "StreamBase: Complex Event Processing," https://www.streambase.com/wp-content/uploads/downloads/datasheets/StreamBase_Brochure_General_Overview.pdf, acessado em Julho de 2013.

[18] "Apache Giraph," http://giraph.apache.org/, acessado em Julho de 2013.

[19] "Apache Drill," http://incubator.apache.org/drill/, acessado em Julho de 2013.

[20] "Apache Flume," http://flume.apache.org/, acessado em Julho de 2013.

[21] "Apache YARN," http://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html, acessado em Julho de 2013.

[22] F. Färber, S. K. Cha, J. Primsch, C. Bornhövd, S. Sigg, and W. Lehner, "SAP HANA Database: Data Management For Modern Business Applications," *ACM Sigmod Record*, vol. 40, no. 4, pp. 45–51, 2012.

[23] J. Baker, C. Bond, J. C. Corbett *et al.*, "Megastore: Providing Scalable, Highly Available Storage for Interactive Services," in *Int. Conf. on Innovative Data system Research (CIDR)*, 2011, pp. 223–234.

[24] J. Leibiusky, G. Eisbruch, and D. Simonassi, *Getting Started With Storm*. O'Reilly Media, Inc., 2012.

[25] J. Kreps, N. Narkhede, and J. Rao, "Kafka: A Distributed Messaging System for Log Processing," in *Proc. of the NetDB*, 2011.

[26] "MapR M5," www.mapr.com/products/mapr-editions, acessado em Julho de 2013.

[27] "Hortonworks HDP," http://hortonworks.com/products/hdp/, acessado em Julho de 2013.

[28] G. Malewicz, M. H. Austern, Bik *et al.*, "Pregel: A System for Large-Scale Graph Processing," in *SIGMOD Int. Conf. on Management of Data*. ACM, 2010, pp. 135–146.

[29] D. Peng and F. Dabek, "Large-Scale Incremental Processing Using Distributed Transactions and Notifications," in *USENIX Symp. on Operating Systems Design and Implementation*, 2010.

[30] S. Melnik, A. Gubarev, Long *et al.*, "Dremel: Interactive Analysis of Web-Scale Datasets," *Proc. of the VLDB Endowment*, vol. 3, no. 1-2, pp. 330–339, 2010.

[31] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster Computing With Working Sets," in *USENIX Conf. on Hot Topics in Cloud Computing*, 2010, pp. 10–10.

[32] L. VoltDB, "VoltDB Technical Overview," 2010.

[33] L. Neumeyer, B. Robbins, A. Nair, and A. Kesari, "S4: Distributed Stream Computing Platform," in *IEEE Int. Conf. on Data Mining Workshops*. IEEE, 2010, pp. 170–177.

[34] A. Lamb, M. Fuller, R. Varadarajan, N. Tran, B. Vandiver, L. Doshi, and C. Bear, "The Vertica Analytic Database: C-Store 7 Years Later," *Proc. of the VLDB Endowment*, vol. 5, no. 12, pp. 1790–1801, 2012.

[35] A. Thusoo, J. S. Sarma, Jain *et al.*, "Hive: A Warehousing Solution Over a Map-Reduce Framework," *Proc. VLDB Endow.*, vol. 2, no. 2, pp. 1626–1629, 2009.

[36] "Cloudera CDH," http://www.cloudera.com/content/cloudera/en/products/cdh.html, acessado em Julho de 2013.

[37] A. Margara and G. Cugola, "Processing Flows of Information: From Data Stream to Complex Event Processing," in *Int. Conf. on Distributed Event-based Systems*. ACM, 2011, pp. 359–360.

[38] "HDFS Architecture Guide," http://hadoop.apache.org/common/docs/current/hdfs_design.pdf, acessado em Julho de 2013.

[39] P. Hunt, M. Konar, F. P. Junqueira, and B. Reed, "ZooKeeper: Wait-Free Coordination for Internet-Scale Systems," in *USENIX Annual Technical Conf.*, vol. 8, 2010, pp. 11–11.