IJCSE International Journal of Computer Sciences and Engineering Open Access
Review Paper Volume-04, Issue-05 E-ISSN: 2347-2693

An Overview of Emerging Analytics in Big Data: In-Situ

Mehjabeen Sultana

Masters in Technology, Department of computer science, Jeddah, KSA

Available online at: www.ijcseonline.org

Received: Apr/21/2016 Revised: May/04/2016 Accepted: May/18/2016 Published: May/31/2016

Abstract—Conventional simulation techniques generate massive amounts of data that are analyzed using various applications. These simulations produce petabytes of data that strains the I/O and storage subsystem. To overcome the high latency in I/O operations, data is analyzed as it is generated, in-situ. This can be successfully achieved by enabling analysis techniques on the same HPC machine that is producing simulation by using the same hardware resources or on a separate analysis machine. In this research paper, we discuss state of the art techniques in this domain and support our conclusion by comparing pros and cons of each approach.

Keywords- Big Data, Service Oriented Approach, Big Data paradigm

I. INTRODUCTION

Due to the availability of powerful computing hardware, our capability to produce high volumes of data as shown in Figure. 1 has outpaced our ability to efficiently store and analyze it. There are large volumes of data that are generated not only from simulations, but also from biometric sensors, customer-shopping patterns and other observations.

On-the-fly analysis of data poses many challenges which typically means changing data capturing codes which is one off the solutions that is rarely accepted. In many scenarios, it is sufficient to analyze a subset of data in detail. Often Big Data analysts encounter false positives in real time even after achieving expected results from these data sets.

To overcome these barriers Big Data OLAP's (Online Analytical Processing) are being hosted in different domains, as OLAPS helps in complex analysis of data and supports business intelligence. For effective automation of data analytics, users are provided with an ability to report the task in domain specific terms and support automatic data aggregation from various sources [1].

The solution is to leverage reusable data assets. In this in-database framework, analysts provide data for constructing predictive models from a pre-integrated repository of data. This repository must allow for retention of historical, detailed data to be used for constructing training sets, to drive model development.

Our concern here is about the I/O performance that has been suffering and has gone down in absolute terms. The implied speed of the I/O subsystem can be altered by including I/O pipelines which provide support for multiplexing, plug-ins for staging applications, framework for large scale calculations.

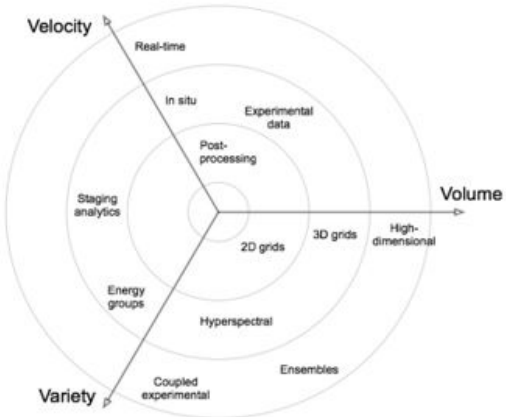

Figure. 1: Big Data paradigm

Scalability and performance, dependability, data access and management are the vital platform features addressed by the underlying operating systems. Big data systems have become pervasive for which design time predictions are insufficient to ensure runtime performance. To address such performance runtime observability and measurement are fraught with peril.

Instrumentation at multiple levels from the above declared entail. Each level provides different information. We explore some of the state of the art practices employed for in-situ data analysis that confront these instrumentations.

II. IN-DATABASE

Hybrid Transaction/analytical processing also known as "in-database" analytics is the combination of database analytics with the functionalities of a Data Warehouse. In this approach, the analytical server is reduced for enhanced performance. All the queries are directly introduced into the database and results are offered on the visualization tools

for analysis and decision making as shown in below Figure. 2

There is a wide availability of tools including the open source tools with provision for data streaming, easy to use in-database mechanisms, extensible development environments.

Statistical tools like R, SAS prefer to compute over numerical and categorized data structured in columnar fashion. SAS and R are typically used for more complex calculations on smaller or medium size data. R has some very good extensions for large data sets and provides a lot more freedom in analytics. But for data management a more database-like tool, SAS is preferred [2]. There has been a plunge in the preference of R over SAS and the surveys conducted delineate R is perceptive.

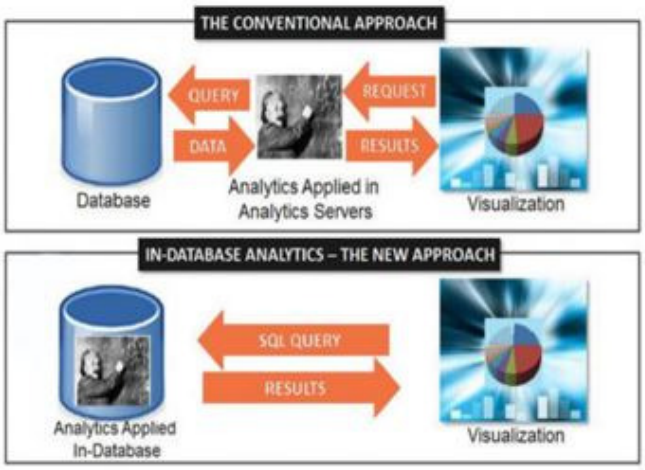

Figure. 2: In-Database Methodology

Whereas Hadoop is an ingeniously simple platform for processing massive amounts of data. It has a basic file system called Hadoop Distributed File System (HDFS) and a programming interface MapReduce. Hadoop is generally for large, straightforward, batch computations with sources like Google File System (GFS), Big Table and HDFS [3]. Hadoop requires minutes to hours of time to complete a typical Hadoop real-time job, making it more adaptable for offline processing [4]. An entire ecosystem of other software has grown up around Hadoop for many purposes, but mainly for extracting, transforming, and loading (ETL Operations) massive amounts of data sets.

While you can perform all of the transactions fast, it still requires for all transactions to reside in one database, hence integrate diverse data.

III. I/O SUBSYSTEM

Exploration of data nowadays occurs mostly through visualization. Many visualization scientist have documented that 50% of visualization time is spent in I/O bottleneck. Because ad-hoc techniques are used to generate the output and due to the overwhelming size of the data, a lot of data is left unexamined. To overcome this, techniques need to be developed that can exploit part of data without reducing the quality of results.

By reading the data from files or work in-situ, the Service-oriented architecture for I/O pipelines provides a framework as shown in Figure. 4 to compile and run individual applications independently termed as "plug-ins". Publish and subscribe can be applied for the services need to be discoverable implementing efficient communication techniques. By processing data using domain specific services without incurring additional data movement, we can manage the cost of I/O.

By considering, an example of SOA approach implemented in CPES project, each of the simulations are developed by different XGC0 and M3D teams with different requirements of libraries and computational resources [5]. SOA approach enables the I/O layer act as communication medium.

For exascale I/O, staging point for data on its way to storage is being adapted. This provides computing resources that complement Service-oriented architecture and can be used for processing data in preparation for analysis. Data staging also offers opportunities for hardware differentiation by supplying huge persistent RAM and ample flash storage as addition memory in staging nodes. Once the data is extracted to staging area, in transit operations can be performed such as complex analysis and in situ visualizations.

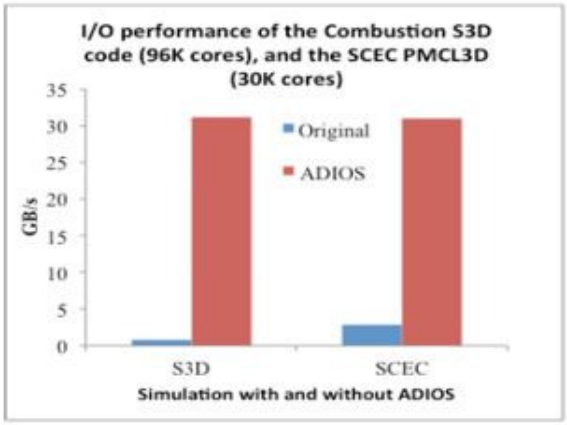

Figure. 3: ADIOS/ Data Spaces performance

Adaptable I/O System (ADIOS) is an abstraction framework with performance as shown in Figure. 3 which provides portable, fast, scalable, metadata rich output with a simple API. It enables to change the I/O methods in real time. It also provisions methods to enable injection of analysis and visualization.

A failure in the usability of I/O optimizations in many I/O libraries has resulted in a severe deficiency in the actual I/O performance realized compared to potential peak throughput.

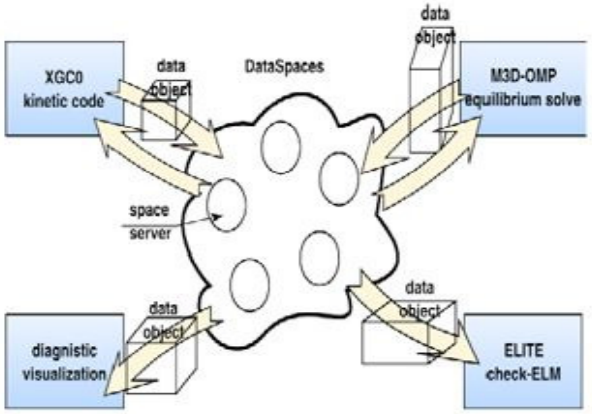

Figure. 4: Multiphysics code coupling (Service Oriented Approach) at Extreme Scale

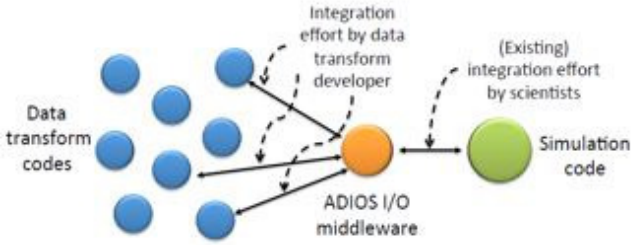

Figure. 5: ADIOS Transform Framework - Positioning in I/O pipeline

Future generation I/O are challenged to partition the optimization task from the actual description of I/O like in Figure. 5.

IV. GRAPHIC PROCESSING UNIT (GPU)

The cost of data movement, both from the application to storage and from storage to analysis or visualization, is a warning to effective use of the data. Input costs, especially to visualization, can make up to 80% of the total running time for scientific workflows often increasing the time required to extract information from the simulation data.

In-situ analysis and visualization implies extremely limited ability to move data from HPC resources by coupling simulation codes and visualization codes. This requires exceptionally limited memory space and renders fine degree of parallelism that can disrupt most processing models.

In-situ visualization libraries enable the user to connect directly to a running simulation, examine the data, do numerical queries and create graphical output while the simulation executes.

The practices of visualization have seen success in pseudocolor, isosurfaces, volume rendering. Para View and Vislt which are open source, platform independent, distributed and parallel tools, have strong similarities: they are both based on VTK as a general purpose scientific visualization toolkit (even if they use different version / patches), use Python as a scripting-glue language and they both implement their different user interface using Qt library [6].

Both tools try to hide the complexity of underlying data-flow VTK pipeline structure by providing functionality at a higher level (data loading, filtering and visualization algorithms). However they differentiate slightly regarding the target user: VisIt is a higher level tool, designed for scientists while ParaView is more flexible but need computer scientist's skills to be extended and customized [7]. AVS/Express, SCIRun and CONVISE and some of the other visualization tools that are being adapted in-situ for quick outputs.

V. CONCLUSION

The biggest challenge here is to determine the insight we wish to gain from in-situ analytics. It is not about getting once arms around all the data which is generated by the simulations, as the aggregate may not be that important, it is finding the news in the detail that is interesting.

In choosing the appropriate solution, we need to differentiate the needs. If we think simple visualizations are all that is required, there are many options available in the data analytics industry.

Challenges are bound to be faced, when we really don't have an idea how to manage and analyze huge data, clear understanding of in-situ mining and modelling techniques, which enables to identify the patterns for problems and opportunities that can work on in order to make a learned decision. As a first step, just take any example data set that is available and apply the above discussed techniques to see what useful insights can be inferred.

As said in big data analytics you will be bound to have these challenges but that doesn't mean that there are no solutions for it, every problem is seen as a discovery which is generally good as it can help forecast and come up with better solutions.

REFERENCES

[1] Sergey V. Kovalchuk1, Artem V. Zakharchuk1, Jiaqi Liao1, Sergey V. Ivanov1, Alexander V. Boukhanovsky "A Technology for BigData Analysis Task Description using Domain-Specific Languages "

[2] The SAS versus R Debate, http://insidebigdata.com/2014/03/01/sas-versus-r/, March 1, 2014.

[3] Heba Aly, Mohammed Elmogy and Shereif Barakat , "Big Data on Internet of Things: Applications, Architecture, Technologies, Techniques, and Future Directions", Nov 2015, Vol 4 No. 06, ISSN : 2319-7323

[4] Understanding BigData Processing and Analytics, http://www.developer.com/db/understanding-big-data-processing-and-analytics.html, September 9, 2013

[5] Scott Klasky at. al., "In situ data processing for extreme scale computing"

[6] Marzia Rivia,*, Luigi Caloria, Giuseppa Muscianisia, Vladimir Slavnicb, "In-situ Visualization: State-of-the-art and Some Use Cases" 1ORNL, 2 U.T. Knoxville, 3LBNL, 4Georgia Tech, 5Sandia Labs, 6 Rutgers, 7NREL, 8Kitware, 9UCSD, 10PPPL, 11UC Irvine, 12U. Utah, 13 Cal. Tech, 14Auburn University, 15NCSU

[7] The Scalable Data Management, Analysis and Visualization (SDAV) Institute, http://sdav-scidac.org/, SciDAC PI meeting 2015

[8] Khanh Nguyen Kai Wang Yingyi Bu Lu Fang Jianfei Hu Guoqing Xu, "FACADE: A Compiler and Runtime for (Almost) Object-Bounded Big Data Applications " University of California, Irvine

[9] Kwan- Liu Ma, Chaoli Wang, Hongfeng Yu, Anna Tikhonova, "In-Situ Processing and Visualization for Ultrascale Simulations" Department of Computer Science, University of California at Davis, One Shields Avenue, Davis, CA 95616 SciDAC Institute for Ultrascale Visualization (IUSV)

[10] 2015 SAS vs. R Survey Results, http://www.burtchworks.com/2015/05/21/2015-sas-vs-r-survey-results/, May 21, 2015

[11] David Loshin, " Big Data Analytics ",Morgan Kaufmann Publishers In, ISBN: 9780124186644

AUTHOR PROFILE

Mehjabeen Sultana is an aspirant of research and inquisitive in High Performance Computing. She is fanatical about personal growth and career development. After working as Software Developer, Mehjabeen has found her true passion for teaching and exploring technology. She loves to share and collaborate with fellow HPC researchers.