On-Demand Big Data Analysis in Digital Repositories

A Lightweight Approach

Zhiwu Xie[1], Yinlin Chen[1], Tingting Jiang[1], Julie Speer[1], Tyler Walters[1], Pablo A Tarazaga[2], and Mary Kasarda[2]

[1]University Libraries and [2]Department of Mechanical Engineering
Virginia Polytechnic Institute and State University, Blacksburg, USA
{zhiwuxie, ylchen, virjtt03, jspeer, tylerw63, ptarazag, maryk}@vt.edu

Abstract. We describe a use and reuse driven digital repository integrated with lightweight data analysis capabilities provided by the Docker framework. Using building sensor data collected from the Virginia Tech Goodwin Hall Living Laboratory, we perform evaluations using Amazon EC2 and Container Service with a Fedora 4 repository backed with storage in Amazon S3. The results confirm the viability and benefits of this approach.

Keywords. Big Data, Data Management, Docker, Scholarly Digital Divide, Use and Reuse Driven Approach.

1 Introduction

The open data and open science movements have gained significant momentum in recent years. Many funding agencies have now instituted new mandates and policies to progressively pressure even the most reluctant researcher towards sharing data for validation and reuse. Eagerly embracing this trend, academic and research libraries are aspiring to lead the management of these research outputs, especially their dissemination, preservation, and curation [1]. To achieve the ultimate goal of openness, it is critical to position the library's role in the larger context of the research data lifecycle [5] to facilitate, not impede the free flow of information, and avoid self-servingly turning library services into extra barriers to entry or even data graveyards.

It has long been argued that shallow openness does not necessarily lead to effective use of information. This is particularly critical for sharing data sets of high volume and high velocity. The "digital divide" not only exists to marginalize the rural population with insufficient IT infrastructure but is also prevalent among academic researchers and domain experts. Lacking appropriate access to big data infrastructure, knowledge, or tools, a large number of these researchers are more and more distanced from participating in data intensive science [3]. This paper addresses the scholarly digital divide by developing a library service that goes beyond open access and preservation. In addition to traditional repository functions, we also provide a flexible and low barrier data analysis infrastructure to allow researchers to submit their own

algorithms for execution against the preserved big data sets in an on-demand fashion. As a result, researchers can focus more on their own research instead of struggling with computing complexities.

2 Use and Reuse Driven Big Data Management

A use and reuse driven approach to manage big data [9] differs from the traditional library repository in that the emphasis is geared more towards serving the researcher's needs to answer domain-specific research questions, instead of building "preservation-ready" systems to satisfy the librarian's urge to document and arrange materials in certain ways to facilitate unspecified future access. The argument is that unless we make fresh data immediately usable and reusable to researchers in their research process, the data will quickly turn cold, become less valuable for long-term preservation, and crowd out limited IT resources for big data management.

Due to this shift of mission and philosophical stance, the OAIS Reference Model [6] is considered inadequate. We need to add an important component missing from the traditional library repository, namely a co-located data analysis infrastructure, to accomplish the goals laid out. Our prior research [9] compared a number of IT infrastructure options with which the use and reuse driven approach may be implemented. Given the IT environment and conditions currently prevalent in most academic libraries, we proposed the public cloud as a viable candidate. Indeed, cloud computing has been attributed to democratizing the science [2] and is well positioned to bridge the scholarly digital divide. This paper furthers our prior exploration by adopting a lightweight approach.

3 A Lightweight Approach for On-Demand Analysis

Our prior approach [9] utilized the computing cloud in its most conventional sense by provisioning virtual machines in lieu of physical machines. We then installed user-supplied analysis code on each of them, and crunched data as if we had a traditional computing cluster on hand. While effective, this approach accrues higher computational overhead since the strong isolation between virtual machines, not essential to data analysis, is enforced nonetheless.

Docker [8], on the other hand, is more lightweight and cheaper since multiple Docker Containers can share the same kernel and application libraries. It requires one extra step from the researchers to "dockerize" their analysis algorithms, but this can usually be automated and does not involve a steep learning curve. We describe the details of our implementation in the next section.

4 Evaluation

The lightweight approach was evaluated using sensor data collected from the Virginia Tech Goodwin Hall Living Laboratory. As the world's most instrumented building

for vibration, the Goodwin Hall facility can accumulate more than 60TB of vibration data per year, forming a fertile ground for researchers to explore how humans interact with the built environment [4]. Since this research field is inherently multidisciplinary and explorative, we must not dictate how researchers build the algorithm and use the data. This makes prebuilt analysis inappropriate, but it does make virtualization and on-demand analysis necessary.

Following our prior system architecture, implementation, and evaluation, we added Amazon EC2 Container Service (ECS), Amazon cloud's support for Docker, to our technology stack. We then dockerized the three simple algorithms used to test 1) ingestion of the data into a Fedora 4 based repository and extraction of technical metadata, 2) calculation of the maximum, minimum, mean, and median value for each of the sensor data file, and 3) visualization of the data file by plotting the HDF5 source files into a diagram, then ingesting the diagram back to the repository. The source code developed for this paper is openly available from Github at https://github.com/VTUL/ICADL. The dockerized code can run on any system supporting Docker, but due to the data co-location requirement, it is much cheaper to deploy in a cloud environment.

We evaluated the performance of the analysis algorithms against 160GB sensor vibration data, collected from one single channel over 24 hours. The data are stored in Amazon S3 and linked from a Fedora 4 based metadata repository running on an EC2 instance in the same availability zone. Instead of provisioning 1, 2, 4, 8, and 16 EC2 instances to perform the heavyweight analysis, we ran up to 4 EC2 instances; each runs up to 4 ECS tasks to perform the same analysis. Table 1 shows the total time required for each test to completely analyze all the data.

Table 1. Time in seconds spent to complete the computation of the test cases

	Number of ECS Tasks				
Test	1	2	4	8	16
1	240.33	180.52	65.86	33.75	16.68
2	676.07	612.52	314.93	159.18	74.03
3	83263.11	53572.53	35035.10	10780.09	5447.75

As in the prior study [9], the results clearly show a linear scalability with the increase in the number of ECS tasks. The only significant exception is when the number of ECS tasks equals 2, where the test case is completed within more than half the duration of using a single ECS task. When we double the number of ECS tasks to 4 and further, the linear scalability reverts to what is expected. We initially thought this might be a mistake, but repeated tests show the same phenomenon. This exception may be due to certain Amazon ECS oddities.

Running the same test using ECS also results in about 1/3, 100%, or 10% longer running time respectively than what the heavyweight approach results in using the same number of EC2 instances as ECS tasks. However, the total cost is much lower, since here we mostly use only 1/4 of EC2 instances as was used in the heavyweight approach. The extra time is expected, since the lightweight approach shares the same EC2 instance within multiple ECS tasks. Although data processing may be more effi-

cient by fully utilizing the shared CPU and memory, the network bandwidth and disk I/O becomes a bottleneck. This is clearly illustrated in both Test 1 and Test 2, where copying data from the storage to the processing node then reading into the CPU clearly dominates the workload. When the number crunching outweighs the network and disk I/O, as is the case in Test 3, the efficiency gain of the lightweight approach is much more evident.

The lightweight approach described here is similar to the yt project [7], although the latter is only targeting data size in the range of tens of gigabytes, in which case moving data around is not as expensive as in most big data management scenarios.

5 Summary and Future Work

As the Docker technology gains popularity in IT operations, digital libraries should consider leveraging its strengths to facilitate use and reuse driven data management. The lightweight approach described in this paper allows cheaper and more efficient execution of user submitted algorithms against the big data sets archived in a digital library. We will conduct more experiments using algorithms from the domain scientists to evaluate the actual performance of this approach and gain deeper understanding of its benefits and limitations.

References

1. Akers, K.G. et al.: Building Support for Research Data Management: Biographies of Eight Research Universities. International Journal of Digital Curation. 9, 2, 171–191 (2014).
2. Barga, R. et al.: The Client and the Cloud: Democratizing Research Computing. IEEE Internet Computing. 15, 1, 72–75 (2011).
3. Farcas, C. et al.: Biomedical CyberInfrastructure Challenges. In: Proceedings of the Conference on Extreme Science and Engineering Discovery Environment: Gateway to Discovery. pp. 6:1–6:4 ACM, New York, NY, USA (2013).
4. Hamilton, J.M. et al.: Characterization of Human Motion Through Floor Vibration. In: Catbas, F.N. (ed.) Dynamics of Civil Structures, Volume 4. pp. 163–170 Springer International Publishing (2014).
5. Higgins, S.: The DCC curation lifecycle model. International Journal of Digital Curation. 3, 1, 134–140 (2008).
6. ISO 14721:2003: Open Archival Information System - Reference Model, (2003).
7. Turk, M.J. et al.: yt: A Multi-code Analysis Toolkit for Astrophysical Simulation Data. ApJS. 192, 1, 9 (2011).
8. Turnbull, J.: The Docker Book: Containerization is the new virtualization. James Turnbull (2014).
9. Xie, Z. et al.: Towards Use And Reuse Driven Big Data Management. In: Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries. pp. 65–74 ACM, New York, NY, USA (2015).