GeoConvention 2014 FOCUS | adapt refine sustain

Bringing the Cloud Underground: Lessons for Bringing the Next IT Revolution to Geoscience

Grant Sanden and Yannai Segal

Enersoft Inc.

Summary

This article describes the emerging technology of cloud computing and the opportunity for cloud computing to advance the practice of geoscience. We discuss the underlying technologies and industry trends and ability of cloud computing to provide average practitioners with the computational resources to solve a new scale of problems. We discuss our experience with deployment of a cloud-based solution for large-scale geostatistical problem solving and provide guidance to others looking at cloud computing for geoscience applications.

Introduction

The field of geoscience is increasingly involving complex theoretical models requiring compute intensive applications/algorithms to handle such complexity and huge data sets generated in the process (Kumar et al. 2012). Improvements in seismic measurement, well logging and laboratory analysis is producing new measurements and at higher resolutions than ever before resulting in upwards of gigabytes of data per well. New forms of analysis are required to process this large volume in data (typically while holding it in memory) and, while computational power is increases with each generation of hardware, the computational resources available to the average practitioner has kept up with the demand produced by emerging methods. Although supercomputing may be a solution for certain large-scale problems, the average practitioner (or even researcher) is not likely to have access to anything beyond a workstation-class desktop.

Cloud computing is defined as *a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction* (Mell and Grance 2011). Cloud computing provides for computational cost-efficiency through use of highly-utilized pooled resources. A Platform-as-a-Service model provides users with the flexibility to access computing resources as needed (typically by allowing running software to create and manage run-ready virtual machines). Cloud-based software is typically delivered under a Software-as-a-Service (SaaS) model with a server-side application accessed by multiple clients. This client/server model is especially appropriate for geoscience applications that are likely to be largely dependent on centrally-stored large data sets.

Theory

Our experience with cloud computing is the result of a project with the goal of testing the performance of new geostatistical modeling methodologies to determine optimal workflows for various resource types. Our goal is to use real-world data sets significantly larger than those typically used in academia and to experiment with many combinations of workflow steps and parameters. We realized that existing platforms and conventional workstation resources would not be sufficient for this project and that cloud computing had the potential to provide required computing resources at reasonable cost.

In order to support our project goals we acquired the rights and source code to a custom software platform that was hosting geological data for about 10,000 wells utilizing cloud storage technologies. Over the course of a year we worked to significantly enhance the storage capabilities of the platform, and to add the capability to create an integrated task-based computational engine with an interface to the data. As of this writing we are beginning to use this platform for full-scale testing.

The following section highlights several key learnings that may be of interest to geoscience practitioners, researchers, or scientific software designers.

Examples

Cost Analysis of On-Demand Cloud Computing

In this section we analyze the cost of on-demand cloud computing in comparison to buying and operating in-house resources.

There are many public cloud service providers including Amazon (AWS), Google (Compute Engine) and Microsoft (Azure) who provide similar services at comparable prices. We chose to work primarily with Microsoft's Azure because of integration with development tools. Azure's on-demand virtual machine pricing as of January 2014 can be found in Figure 1.

Figure 1: Microsoft Azure Virtual Machine Pricing

Microsoft Azure Virtual Machine Pricing					
Service Level	CPU Cores	RAM	Local Non-OS Storage	Price/Hour (CAD)	Annual Equivalent (CAD)
A1	1	1.75 GB	224 GB	$0.022	$193
A2	2	3.5 GB	489 GB	$0.095	$832
A3	4	7 GB	999 GB	$0.190	$1,664
A4	8	14 GB	2,039 GB	$0.380	$3,329
A5	2	14 GB	489 GB	$0.422	$3,697
A6	5	28 GB	999 GB	$0.844	$7,393
A7	8	56 GB	2,039 GB	$1.688	$14,787

We find purchase costs for equivalent computers at about 2/3rd of the annual equivalent costs (an A2-level machine would cost about $550 and an A7-level machine would cost about $10,000 to purchase). In corporate IT application, lifetime operating costs are typically estimated as at least equal to hardware costs (Morey and Roopa 2010) and the useful lifespan as 2 to 3 years. This suggests that cloud computing will have a lower total cost of ownership than an internal data center if computing utilization is below 68% (2-year lifespan) or 45% (3-year lifespan). We suspect utilizations are lower than this for most practitioner (and researcher) projects, particularly over the entirety of hardware lifespan.

Of course, the real advantage to cloud computing is that properly parallelized code can be deployed to a far greater number of computers than normally available to practitioners. An analysis that would take 100 days to run on a single machine could be run on 100 machines over one day. We expect this multiple-order-of-magnitude increase in the computing horsepower available to practitioners will drive major advances in geoscience over the coming years.

Creating a Cloud Computing Platform

In this section we discuss our experience integrating cloud storage with a task-based computational engine.

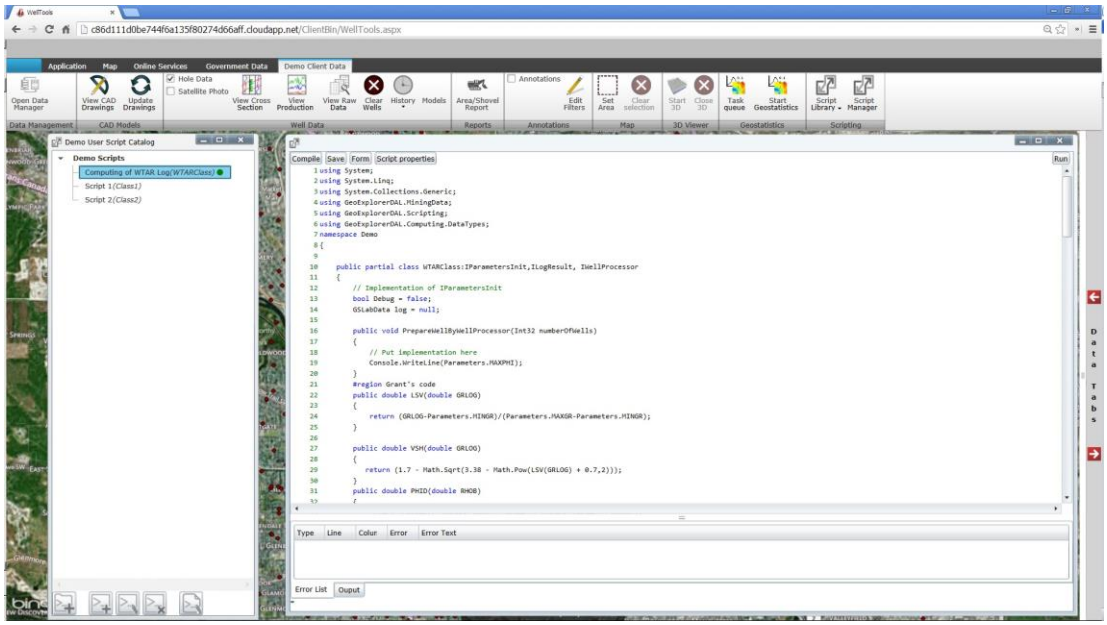
Data-intense geological modeling is an ideal application for cloud computing because of the data density and high computational requirements. This requires all data-intense calculations to be performed server-side. It should be noted there will likely still be requirement to access data from the client side for visualization, selection, data verification and other user-focused tasks. With large underlying data sets this can be a non-trivial task (i.e. allowing users to select from a list of millions of data elements without needing to download the entire element list). We found existing programming best-practices can deal with this problem if it is addressed as part of the design stage.

In our primary field of interest, geostatistics, much of the practitioner and research code is written in Fortran, including hundreds of projects worth of code from University of Alberta's Centre for Computational Geostatistics. This posed a challenge as Fortran is a legacy language largely unsupported by modern cloud platforms and the structure of the existing code was not conducive to functional integration. We pursued undertaking a major code conversion to a modern programming language, but deemed that solution cost-prohibitive. The most efficient solution was to slightly modify and compile existing Fortran code to command-line-callable executable programs. These programs can be installed on cloud servers and virtual machines and be employed within modern programming languages.

A design goal for our testing platform was the ability to create, test and run new code independent of modifying the core application. Our original plan was to write the code in a specialized development environment on the desktop and build an interface to transfer this code to the cloud-based platform. However, we were able to create a client interface that allows for coding creation and execution, with access to a development environment and our full data object model, all within the web-based client as seen in

Figure 2. This significantly simplifies our ultimate model testing process.

Figure 2: Scripting Interface



Data Storage and Access

In this section we discuss developments in data storage and access technologies and implications for high-performance big data technical computing.

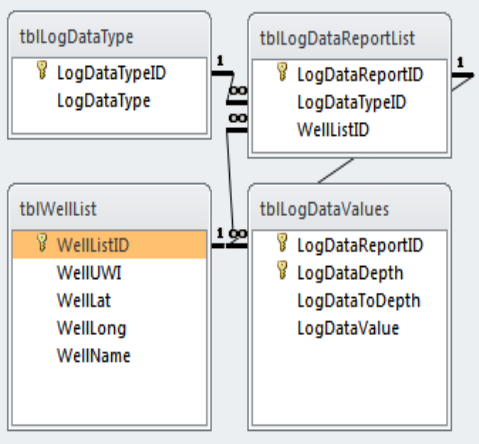
E&P data is traditionally hosted in relational databases broken over many normalized tables such as the Public Petroleum Data Model (PPDM). Such databases are often called SQL databases because a

structured query language is used to reassemble the data in many related tables into a useful form. Because proper relationship design breaks up data into as many tables as required to avoid any duplicate data elements, complexity and of table structure and of queries required to extract data can be quite high.

Figure 3 shows the query language required to extract the value of specific lab test from a specific depth of a specific well in a simple four-table relational database model of lab values. This complexity results in reduced performance for increasingly large data sets.

Figure 3: Sample Table Structure and Query

```
SELECT tblLogDataValues.LogDataValue
FROM tblWellList INNER JOIN (tblLogDataType
INNER JOIN (tblLogDataReportList INNER JOIN
tblLogDataValues ON
tblLogDataReportList.[LogDataReportID] =
tblLogDataValues.[LogDataReportID]) ON
tblLogDataType.[LogDataTypeID] =
tblLogDataReportList.[LogDataTypeID]) ON
tblWellList.WellListID =
tblLogDataReportList.WellListID
WHERE
(((tblWellList.WellUWI)="1AA092509213W400")
AND ((tblLogDataType.LogDataType)="Gamma") AND
((tblLogDataValues.LogDataDepth)=115));
```



Our original intent was to use a traditional relational database for data storage. However, despite optimization efforts it became apparent that there would be performance issues related to accessing data as we scaled up the database size and that a technical solution was required to remove the bottleneck.

There is a new family of database technologies commonly referred to as Not Only SQL (NoSQL) databases that have been developed to meet the performance requirements of Big Data. Of particular relevance to geological modeling are document-oriented NoSQL databases such as MongoDB and CouchDB. These databases store data in a hierarchical format that in can be often resembles the natural structure of geological or production data and are far easier to implement in an object-relational mapping framework. A sample of a MongoDB document storing well lab data can be found in Figure 4

A transition of lab data from MS SQL to Mongo DB increased search and retrieval performance by an order of magnitude and enabled us to remove a major performance bottleneck to meeting our testing goals.

Figure 4: MongoDB Document

```
{{
 "Well": "1AA092509213W400",
 "Lat": "57.0118",
 "Long": "-111.942",
 "Name": "Bob 32",
 "Type": "Completed",
 "LogData": {
   "Gamma": {
      ["Depth": "125", "Value": "28.760"],["Depth": "120", "Value": "26.365"],
      ["Depth": "115", "Value": "24.368"],["Depth": "110", "Value": "31.256"],
      ["Depth": "105", "Value": "33.125"]
  }}},
 {
 "Well": "1AA123009212W400",
 "Lat": "57.0131",
 "Long": "-111.9338",
 "Name": "Bob 26",
 "Type": "Drilled",
 "LabData": {
   "Neutron": {
      ["Depth": "135", "Value": "11.2"],["Depth": "130", "Value": "8.7"],
      ["Depth": "125", "Value": "6.9"],["Depth": "120", "Value": "7.4"]
}}}}
```

Conclusions

Emerging cloud computing technologies can provide geoscience practitioners and researchers with access to the computing needs required for data- and processor-intensive tasks. We determine that current cloud pricing suggests significant cost savings over conventional models at likely practitioner utilizations. We also make suggestions on how to implement legacy Fortran code into modern cloud projects and on use of emerging database technologies to handle large-scale data.

References

Kumar, M., Manral, D.S. Benerjee, M.K., Karmakar, K., Das, A., Reddy, B.J., Dasgupta, R., Singh, S.N, 2012, High Performance Computing in Geoscience: Promises and Challenges, in Proceedings, 9th Biennial International Conference & Exposition on Petroleum Geophysics, Hyderabad Pakistan, p208

Mell, P., Grance, T., 2011, The NIST Definition of Cloud Computing, Special Publication 800-145 of the National Institute of Standards and Technology of the U.S. Department of Commerce, Gaithersburg MD, p. 2

Morey, T., Roopa, N., 2010, Using Total Cost of Ownership to Determine Optimal PC Refresh Lifecycles, Wipro White Paper for Intel Corporation