Abstract

The Virtual Geophysics Laboratory (VGL, http://vgl.auscope.org) provides a flexible, web based environment where researchers can browse data and use a variety of scientific software packaged into "tool kits" that run in the Cloud. Both data and tool kits are published by multiple researchers and registered with the VGL infrastructure forming a data and application marketplace. The VGL provides the basic work flow of Discovery and Access to the disparate data sources and a Library for tool kits and scripting to drive the scientific codes. Computation is then performed on the Research or Commercial Clouds. Provenance information is collected throughout the work flow and can be published alongside the results allowing for experiment comparison and sharing with other researchers.

VGL's "mix and match" approach to data, computational resources and scientific codes, enables a dynamic approach to scientific collaboration. VGL allows scientists to publish their specific contribution, be it data, code, compute or work flow, to a "marketplace" for science workflows. Other scientists can choose the pieces that suit them best to assemble an experiment. The coarse grain workflow of the VGL framework combined with the flexibility of the scripting library and computational toolkits allows for significant customisation and sharing amongst the community.

The concept of a Scientific Code Marketplace (SCM), a outcome from the VGL project, but having diverse application beyond the VGL target audience, is outlined within this paper. The paper will detail the concept, architectural elements for such a system and potential benefits from its application

Key words:

Virtual Laboratories, Cloud Computing,

1.Introduction

A virtual laboratory has 3 phases: firstly the data selection, secondly the selecting and tuning of the processing algorithms, and thirdly submitting the job to the most appropriate computational facility and monitoring. A provenance workflow is automatically captured as part of the metadata. The entire infrastructure developed provides geoscientists with an integrated environment that provides seamless access to distributed data libraries and loosely couples these data to a variety of processing tools. In other words, a virtual laboratory is in essence a broker that uses interfaces compliant with OGC/ISO standards to enable resources from physically distributed systems to communicate with one another in an online virtual environment. The laboratory can link to a variety of compute resources than span from petascale HPC systems to private and commercial clouds and to local desktop. The user accesses the laboratory through an intuitive user-centred interface that enables real-time seamless linkage between all components. A provenance workflow is automatically generated in the background by capturing information on all inputs to the processing chain, including details about the user and their organization. The resultant metadata record is compliant with the ISO 19115 metadata standard and enables a team to not only keep track of jobs they have submitted, it also enables spatial displays of who ran what, where and when.

The benefit of the provenance workflow is that all products produced are transparent. As the metadata record captures all input files, including any changes to processing algorithms, time of extraction of the data from the databases, who ran what and where, the components of workflow can be reused by others wanting to run similar workflows. The greatest benefit is that all procedures used are accessible, verifiable and can be used by other investigators to test the results and hence there can be credibility of products produced.
A further benefit of transparent processing is that by openly sharing our work we create a platform that supports further innovation in the areas of scientific research, particularly if the data is accessible through open access mechanisms and any tools developed are open source.

Over the last five decades geoscientists from Australian state and federal agencies have collected and assembled around 3 Petabytes of geoscience data sets under public funding. As a consequence of technological progress, data is now being acquired at exponential rates and in higher resolution than ever before. Effective use of these big data sets challenges the storage and computational infrastructure of most organisations. The Virtual Geophysics Laboratory (VGL) is a scientific workflow portal addresses some of the resulting issues by providing Australian geophysicists with access to a Web 2.0 or Rich Internet Application (RIA) based integrated environment that exploits eResearch tools and Cloud computing technology, and promotes collaboration between the user community.

VGL simplifies and automates large portions of what were previously manually intensive scientific workflow processes, allowing scientists to focus on the natural science problems, rather than computer science and IT. A number of geophysical processing codes are incorporated to support multiple workflows. For example a gravity inversion can be performed by combining the Escript/Finley codes (from the University of Queensland - https://launchpad.net/escript-finley/) with the gravity data registered in VGL. Likewise, tectonic processes can also be modelled by combining the Underworld code (from Monash University - http://www.underworldproject.org/) with one of the various 3D models available to VGL.

Cloud services provide scalable and cost effective compute resources. VGL is built on top of mature standards-compliant information services, many deployed using the Spatial Information Services Stack (SISS – http://siss.auscope.org), which provides direct access to geophysical data. A large number of data sets from Geoscience Australia assist users in data discovery. GeoNetwork (http://geonetwork-opensource.org/) provides a metadata catalogue to store workflow results for future use, discovery and provenance tracking.

VGL has been developed in collaboration with the research community using incremental software development practices and open source tools.

While developed to provide the geophysics research community with a sustainable platform and scalable infrastructure; VGL has also developed a number of concepts, patterns and generic components of which have been reused for cases beyond geophysics, including natural hazards, satellite processing and other areas requiring spatial data discovery and processing.

A key learning from the VGL project was the interest that was generated not only with the end-user research community that simply wanted to use the VGL for what it was (a Geophysical data processing workflow engine) but from scientists that either develop computational codes for the geosciences (and other domains for that matter) and researchers wanting to "share" or publish the way they do things (their workflow) to a greater audience. For this reason, the team has explored the notion of a Scientific Code Marketplace (SCM) for the cloud and virtual laboratories so that computational codes, code snippets or processes can be more easily shared and published to research audiences.

The remainder of this paper will focus on the lessons learnt, drivers and intended design for such a mechanism as the SCM

2. Material and Methods

Architecture for Virtual Laboratories that incorporate a data processing workflow is detailed in Figure 1. It highlights that the architecture requires multiple components to fulfill a complete scientific workflow. This paper does not go detail each component as these are well documented and researched elsewhere. The focus of this paper will be on how the various user types contribute their "science" to the virtual laboratory.

Virtual Laboratory Example System Architecture

Figure 1: Architecture for Data/Computational Virtual Laboratory

A Scientific Code Marketplace (or SCM) requires to support various users, in particularly, the Science Code Developer and the Research Scientist (or Science Workflow Publisher). It is the intent of the SCM to endorse good software engineering practice, standardisation and leverage emerging patterns in software development.
Science code developers use case

Science Code developers (SC users) are researchers that develop computational software codes that provide computational simulations of particular phenomena. Examples of which are …[do I reference eScript or other again – don’t really want to replug Underworld though!!???] These users would like a mechanism to contribute their code more easily to VL environments, as currently it is a manual process with no standardisation.

The SCM defines key elements to standardise the way science code is contributed to a VL environment.

1) science codes need to be managed by a version control system (preferably GitHub, however SVN, cvs, googleCode would suffice)

2) mechanism must be in place to “build” codes. The preferred mechanism for this within the Cloud environment is Puppet [ref]. Puppet can be used to provision a “virtual machine” within the Cloud environment, “checkout” code from a managed repository and finally build/deploy the code on the machine.

3) Image must allow VL to use it (VL permissions to be set to allow VL access)

4) Newly created Cloud image must be registered within or discoverable by the VLs registry

5) At least one “input script”/example input file is needed so that the VL can run the science code – this too needs to be registered in the VL registry and in an accessible repository

Once the above are satisfied, the newly available code can be utilised by a VL. The VL will be able to discover and initiate a VL image in the Cloud. The “Script Builder” component within the VL will acquire the “input script” from the registry automatically and expose it within the VL.

<<diagram here on how the science code mechanism will work – activity diagram>>

Science workflow publishers

Scientists and researchers would like a standardised manner to “publish” their code contributions to an environment that makes them executable by others. This compliments the existing abilities to publish papers and make research data available via the web. The ability to “share” workflows or “how I do my science” is a key element to providing greater insight into how results (the outcomes in scientific papers) were come about – workflow provenance

Assumptions:

- code snippets/templates/example input files to run Scientific Codes are in managed code repositories (ie: Github, svn, cvs etc)

- Science Code (see user 1 requirements) needed to run contributions

- Snippet is script-based, input ascii text file, any submission that does not need compilation

Requirements:
- Code snippet must be registered within the SCM – this includes metadata about the code and a system could obtain it

- System must "check-out" code from response

- Virtual Lab template to be automatically generated for the new code snippet

Script Builder

The “Script Builder” within a VL will require the functionality to consume the offerings from a this user. This component will need to be able to query the VL registry for code workflows and expose them within its environment. This could simply be as a text viewer or a more feature rich viewer/form layout. The component will need to link the workflow with the desired science code VM in so that it can run within the correct environment.

Science End-users

3. Results

The intent of the SCM is to simplify the sharing and publishing of scientific codes and workflows within a cloud environment. The SCM architecture provides the following two elements:

- necessary mechanisms for science code developers to make their science code available to a cloud and high performance computing environment by outlining key architectural components necessary to publish the code to the cloud.

- mechanism for Researchers (end users) to share and publish their workflows in a web environment (Virtual Laboratory) - the workflows can then be shared or published to be used by others - essentially researchers are now able "how" they can to their conclusions through executable code (in addition to tradition science publishing - papers and data provision)

The SCM key driver is for better sharing of scientific contributions through executable web-accessible environments and can be paralleled to that of mobile device application stores (ie: Google Play Store, Apple iTunes/appstore)

The SCM for the cloud and virtual laboratories will provide the standardisation necessary to “publish” and sharescientific code contributions just as data is shared and discoverable and accessible.

SCM as an independent piece of cloud infrastructure, will be able to be utilised by virtual laboratories that can communicate

4. Conclusions

5. Recommendations
Appendices

References

-- RyanFraser - 07 Jan 2014

-- RyanFraser - 20 Aug 2014