Workspace: Scientific Workflows and Applications for multiple Environments

Matt Bolger, Paul Cleary, Lachlan Hetherton, Chris Rucinski, David Thomas and Damien Watkins
CSIRO, Clayton, Victoria, Australia
Email: firstname.lastname@csiro.au

CHALLENGE
Developing a scientific software platform that caters for cross-discipline workflows means meeting the unique requirements of different scientific domains. In addition, there are the complexities and constant evolution of technologies in the underlying computing domain. Scientific software platforms (somewhat similar to a Scientific Workflow System) attempt to abstract many of the underlying computing technologies. For example many such platforms can allow a scientist to conduct their research and execute their workflows identically on any one of many different operating systems. Few platforms, however, support developing scientific workflows that can both execute on and interact with the entire spectrum of computing infrastructure, ranging from a mobile device to cloud systems. Small footprint devices are great for gathering data in the field but lack the power and storage to run complex computations and simulations. Large cluster or cloud systems can handle huge datasets and massive computations but are often housed in large purpose built facilities the size of a warehouse, making them impractical for fieldwork, or for rapid prototyping. For example, a device such as a radio telescope can be deployed in the field and provide large computational and storage capacity, but its use is very domain specific. The challenge is to create a scientific software platform that allows scientists from different domains to simply and intuitively build complicated, multi-disciplinary workflows from suites of individual operations, running on and/or interacting with many different platforms and devices.

REQUIREMENTS
Scientific software platforms exist to abstract the underlying computational hardware and software technology stack to allow scientists to focus on their scientific domain - as opposed to the computing implementation domain. These platforms can range from scientific programming languages, such as the R Programming Language, to software libraries and frameworks, such as Python’s NumPy and SciPy libraries, through to scientific workflow systems, such as Galaxy, Kepler and Taverna. These scientific software frameworks often provide additional benefits such as operating system independence, visualisation, concurrency, provenance and so on. A platform that can support both 1) interaction with and 2) application development for everything from small mobile devices up to cloud-based systems provides yet another necessary feather in a scientist’s cap.

Many scientific workflows often start with data collection from a device in the field. For example, this may involve the capture of imagery from devices ranging from cameras mounted in quadcopters or smartphones, all the way up to expensive and domain specific devices such as telescopes and satellites. The data may then require pre-processing, which could take place on an individual workstation, or alternatively in the case of large datasets, a processing algorithm may be first developed and verified on a single workstation using a subset of the data and then later run against the entire dataset on a cluster or cloud system. Finally the results may need to be used in the field, for example visualised on a mobile device – as such the workflow has come full circle.

WORKSPACE
Workspace is a scientific workflow platform developed at the CSIRO. Workspace is built on top of the Qt Framework and optionally incorporates a number of other scientific libraries such as NetCDF, GDAL and OpenCV to name but a few. The Qt framework provides support for deploying applications to Linux, Mac, Windows, iOS, Android and Windows Mobile. Workspace provides support for building operations on the platform most suited to the execution environment required, as well as support for utilising a number of different platforms in a single scientific workflow. Additionally, Workspace facilitates the development of custom applications built on top of scientific workflows. The following sections feature examples of the different platforms that can be utilised in a Workspace workflow and Workspace-based applications.
USE CASE: MULTIPLE EXECUTION PLATFORMS

As mentioned previously, Workspace, Workspace workflows and Workspace-based applications, can be run on Linux, Mac and Windows operating systems. This cross-platform desktop deployment has been Workspace’s core focus since it was initially developed. This has also permitted Workspace to run on many cluster and cloud systems, as these often provide such operating system images. In the last year this focus has been expanded with the team developing support for deployment of Workspace-based applications to mobile Android and iOS devices as well as a web deployment solution stack. This goal of even more deployment options for research code reflects the evolution of computing environments used by many researchers. One such example is Spark. Spark is a wildfire simulation framework for researchers and experts in the disaster resilience field developed by the Spark development team at CSIRO, and built on the Workspace platform. The team includes Bushfire Behaviour scientists, Computational Modelling scientists and Software Engineers. The Spark bushfire application can be run on desktops, cloud systems and can also be integrated with web server technologies.

![Figure 1: Spark, a Workspace-based application, running on an AWS instance and website](image1)

USE CASE: IMAGE CAPTURE IN UNUSAL ENVIRONMENTS

The Markerless Motion Capture project deals with capturing images from Olympic standard swimmers during training and uses CSIRO’s Image Analysis software and OpenCV, integrated within Workspace, to identify parts of the swimmer’s body. These are overlaid with a digital representation of the swimmer to construct a model of their action in water for use in numerical simulations for human performance modelling.

![Figure 2: Capturing images and generating computational models using Olympic swimmers.](image2)

USE CASE: VISUALISATION IN IMMERSIVE ENVIRONMENTS

Out of the box, Workspace provides advanced visualisation capabilities such as 3D mesh and volume rendering along with the ability to interactively visualise data during workflow execution. Visualisation software such as Unity and Maya and hardware such as the Oculus Rift and Microsoft Kinect can be connected to a Workspace workflow to provide an immersive experience. In the example below a user is experiencing a virtual bushfire in a Unity-based visualisation and interacting with a visualisation using a Microsoft Kinect.
CONCLUSION

New devices and technologies are constantly emerging from the computing domain - enabling new and innovative ways to get science done. However most scientists do not have the time, inclination or even expertise to incorporate these new technologies into their system. A scientific software platform such as Workspace allows scientists to leverage an increasing array of tools anywhere they may be needed in their workflow and reduces the time and effort needed to incorporate new technologies.

ACKNOWLEDGEMENTS

We would like to acknowledge Craig Scott and Ben Morris, who were previous development team members, for all their contributions. We would like to acknowledge all the scientists whose work is featured in our Use Cases.

REFERENCES


ABOUT THE AUTHOR(S)

Matt Bolger is a senior software engineer in CSIRO’s Computational Modelling and Simulation Group. Matt has a BSc (Computer Science) from the University of Melbourne and started his career working in the computer game industry before joining CSIRO in early 2010. He has worked on commercial software products ranging from children’s games, aircraft and sports simulations to more recent work in industrial processes, disaster modelling and cross-platform workflow frameworks. Matt currently works across a number of research and commercial projects in areas such as CFD, computer vision, 3D rendering, human kinematics and additive manufacturing. Matt’s work involves developing on a wide range of computing environments using a variety of programming languages and frameworks.

Paul Cleary is a Chief Research Scientist at CSIRO and specialises in the development and application of particle based computational methods and the development of software and visualisation tools. These include the DEM and SPH methods which are now used extensively for simulating industrial, biomechanical / biomedical and geophysical / geotechnical flow problems involving combinations of particulates, fluids and bubbles. He is broadly acknowledged as a leading developer of the SPH and DEM numerical methods and is the architect and primary software developer of both the CSIRO DEM and SPH software. He is also the leader of the Workspace development program which is building and deploying a sophisticated general purpose workflow engine that enables efficient inter-connection and interoperability of heterogeneous software pipelines and their migration into commercialisable products.

Lachlan Hetherton is a senior software engineer from the Computational Modelling and Simulation group in CSIRO, specialising in developing visualisation capabilities for Workspace. Specifically, he has implemented Workspace’s 3D scene rendering and grid plotting capabilities, as well as the recently added NetCDF and GeoSpatial plugins. Prior to joining CSIRO in 2009, Lachlan worked as a consultant for Accenture, designing and developing solutions for a diverse range of projects in the telecommunications space.

Chris Rucinski is a software engineer in CSIRO’s Computation Modelling and Simulation Group. He graduated as a computer scientist from Monash University in 2011 and since then has been a developer on the Workspace team. He has developed a number of computational modelling and data analysis tools that are used internally by the Computational Modelling Group and which are also deployed externally in commercial particle-based simulation software packages. Chris collaborates with researchers from multiple flagships to design and build user-friendly software tools relating to mining, metallurgy, bushfires and urban sustainability. Chris is the lead developer on Amicus, AlteTreat and other Workspace-based applications.

David Thomas is a software engineer in CSIRO’s Computational Modelling group. He currently manages and develops commercial software using Workspace, providing user interfaces for complex scientific applications. He has a first-class Bachelor’s (Hons) degree in Mathematics, and a Master’s in Mathematics and Chemical Engineering, both from The University of Manchester in the UK. He is a software developer with over 18 years’ commercial experience, the majority of this time spent working in the CAD industry, working on core functionality used by some of the biggest product vendors such as Autodesk, SolidWorks, NX and CATIA. He has also provided consultancy work for SolidWorks.

Damien Watkins is a Research Team Lead in CSIRO’s Computational Modelling and Simulation group. His team produces a number of software applications, including Workspace which is a scientific workflow engine used across a number of projects within CSIRO. Prior to working at CSIRO, Damien worked for Microsoft/Microsoft Research for ten years. Damien holds a Ph.D. in Software Engineering from Monash University.