19th International Congress on Modelling and Simulation, Perth, Australia, 12–16 December 2011
http://mssanz.org.au/modsim2011

RS-YABI: A workflow system for Remote Sensing Processing in AusCover

Z. Wang [a], E. King [b], G. Smith [b], M. Bellgard [c], M. Broomhall [d], H. Chedzey [d], P. Fearns [d], R. Garcia [d], A. Hunter [c], M. Lynch [d] and D. Shibeci [c]

[a] *CSIRO Marine and Atmospheric Research, Canberra, Australia*
[b] *CSIRO Marine and Atmospheric Research, Hobart, Australia*
[c] *Centre for Comparative Genomics, Murdoch University, Perth, Australia*
[d] *Remote Sensing and Satellite Research Group, Curtin University, Perth, Australia*
Email: ziyuan.wang@csiro.au

Abstract: Earth observation from space involves accessing, sharing, and processing huge volumes of data collected from satellite sensors. It is crucial for users to easily analyze and process the datasets, which may require complex computation on very high volume datasets, particularly for large-scale spatio-temporal analysis. Generation of data products, access to, analysis and visualization of these large datasets is often perceived as a highly technical and daunting set of complex steps. In this context, a workflow application can help users to manage and process large volumes of satellite data and execute scientific experiments on distributed resources. This paper reports our work on utilizing a workflow engine (RS-YABI) that works with data from Moderate Resolution Imaging Spectroradiometer (MODIS) sensors and enables researchers to process datasets using High Performance Computing (HPC) resources. The benefits of this approach include abstracting the complexity of the underlying processing environment and easy processing of raw satellite data to generate derived data products in standard formats.

We start with providing an overview of AusCover, a facility of the Terrestrial Ecosystem Research Network (TERN), as the application context of our work. It provides a national expert network and a data delivery service for provision of Australian biophysical remote sensing data time-series, continental-scale map products, and selected high-resolution datasets over Terrestrial sites. AusCover supports a nationally consistent approach to the delivery and calibration/validation of key current and future core satellite-derived datasets.

Some background of workflows and remote sensing data processing is described next. In our context, using a workflow for processing remote sensing data offers several advantages, such as (a) the ability to design an operational process by leveraging existing application modules, (d) utilizing distributed resources to increase throughput or reduce execution costs, (c) obtaining specific processing capabilities as required by users, and (d) hiding technical complexity behind a straightforward user interface.

It is followed by the description of YABI, a workflow engine developed at Murdoch University, Australia, originally for use in a bio-informatics context supporting linear workflows similar to those used in remote sensing processing. It provides a flexible mechanism for joining independent executables through wrappers that can be implemented in virtually any language. Examples of ongoing work on the development of wrapping scripts for customizing specific remote sensing operations, orchestrating multiple modules and simplifying user input choices are described.

One of YABI's key features is its abstraction of the backend HPC resources such as file stores and execution engines which makes it ideal for use in a distributed data system such as that developed by AusCover. YABI has been deployed as RS-YABI (for Remote Sensing) on HPC resources in the National Computational Infrastructure (NCI). The RS-YABI instance is deployed on a Virtual Machine (VM), which provides a self-contained environment independent of the HPC system. Multiple workflows for processing MODIS sensor data have already been successfully developed. We also present two brief case studies to demonstrate the applicability of RS-YABI in practical application areas such as dust detection, surface temperatures (land and ocean) monitoring, and smoke plume detection. We conclude the paper with a list of future research directions.

Keywords: *Remote Sensing (RS), workflow engine, AusCover, High Performance Computing (HPC)*

1. INTRODUCTION

Remote sensing (Lillesand et al., 2004) has always been at the forefront of automated data acquisition and has traditionally produced data sets that are large, in both data storage requirements and computational demand senses, relative to the computing capacity available to process them. This has been a long-standing obstacle to the use of remote sensing data. Typical responses to this problem have been to work with data sets reduced in either or both time and space (extent and resolution), or by averaging the data. All of these approaches restrict the exploitation of the densely sampled character of the data. This is particularly apparent as modern analysis approaches, such as data assimilation, develop the capacity to use more and more data directly, and the scope of modeling efforts extend to continental and global scales. The restriction in scope of data processing and storage also leads to duplication of processing systems as they are replicated to work on different subsets of the same sensor data set. Moreover, as the user community is becoming increasingly specialized, it is more difficult to maintain the additional expertise required to undertake the complex processing that remote sensing data often demands.

AusCover is a facility of the Terrestrial Ecosystem Research Network (TERN). It provides a national expert network and a data delivery service for provision of Australian biophysical remote sensing data time-series, continental-scale map products, and selected high-resolution datasets over Terrestrial sites. AusCover supports a nationally consistent approach to the delivery and calibration/validation of key current and future core satellite-derived datasets. The primary goal is to assist in the production of ecosystem science data products designed specifically for Australian conditions. Issues related to the processing of remote sensing data are particularly severe in AusCover where the aim is to deliver products to a user community of ecologists who may not have previous experience with remote sensing data. A means of simplifying the processing and reducing the duplication in storage and processing systems for remote sensing data would therefore be a welcome development.

An opportunity to achieve such a development is possible due to the eResearch initiatives currently underway within Australia under the Platforms for Collaboration umbrella of the National Collaborative Research Infrastructure Strategy (NCRIS), together with the research domain consolidation of data management in the related TERN and Integrated Marine Observing System (IMOS). In this context, to consolidate both the storage and processing of key remote sensing time series data sets with National coverage, we have chosen to use the National Compute Infrastructure (NCI)[1] as a platform.

In this paper, we describe the deployment of a workflow system (RS-YABI), for managing the processing of remote sensing data and facilitating web-based control of customized remote sensing products by end-users. The main contributions of this paper are to describe:

- The approach to incorporate remote-sensing processing chains in a workflow paradigm.
- The deployment of an easy-to-use workflow service framework in an HPC infrastructure.
- The demonstration of the use of the workflow system in application contexts.

The rest of the paper is structured as follows. Section 2 provides a brief background on workflows and remote sensing data processing. It is followed by the description of RS-YABI, a workflow system for remote sensing data processing. Section 4 reports the current status of our work, in particular, the deployment of RS-YABI on HPC resources at the NCI. Case studies on the use of RS-YABI are reported in Section 5. Finally, the paper is concluded in Section 6 with a list of future research directions.

2. WORKFLOWS AND REMOTE SENSING DATA PROCESSING

Workflows are sequences of related steps or operations and have been a topic of research in computer science for many years (Georgakopoulos et al., 1995; Van Der Aalst and Van Hee, 2004; Russell et al., 2005). Scientific workflows are concerned with the automation of a scientific process in which tasks are structured based on their control and data dependencies. The typical remote sensing processing chain is a straightforward sequence of steps with no branching or looping. Moreover, the processing of large multi-scene remote sensing data sets is inherently parallelisable as each data granule is generally independent of all others in both space and time. Therefore, processing multiple granules can be distributed across multiple processors very naturally by using a scientific workflow to replicate processing. This approach is well suited to processing on the modern grid environments and cluster based processing systems.

[1] NCI is a computational facility in Canberra, Australia, which is developing capabilities in data intensive computing. Further information can be found in the website: http://www.nci.org.au

Using a workflow for processing of remote sensing data offers several advantages such as (a) ability to design an operational process by leveraging existing application modules, (b) utilizing distributed resources to increase throughput or reduce execution costs, (c) obtaining specific processing capabilities as required by users, and (d) hiding technical complexity behind a straightforward user interface. Figure 1 shows three simple linear workflows typical in the processing of MODIS remote sensing data using the SeaWIFS Data Analysis System (SeaDAS). In the first workflow (orange), raw MODIS data is selected, and then it is processed using SeaDAS. This processing stage involves several computational steps comprising the processing of the data from Level 0 to Level 1B. The second workflow (shown in green) illustrates the processing of L1B data to Normalised Difference Vegetation Index (NDVI) for terrestrial applications. The blue workflow shows processing for a marine product. Every step of each of these workflows can be, and normally are computed separately. Normally an end-user would have to execute each step separately. By encoding multiple steps into workflows, the number of required user operations is dramatically reduced. Notice that the first workflow is also required to produce the intermediate product (L1B), which is necessary for both terrestrial and marine products. In this way the first workflow needs to be set up only once and can be reused, thus efficiently reducing the duplication.

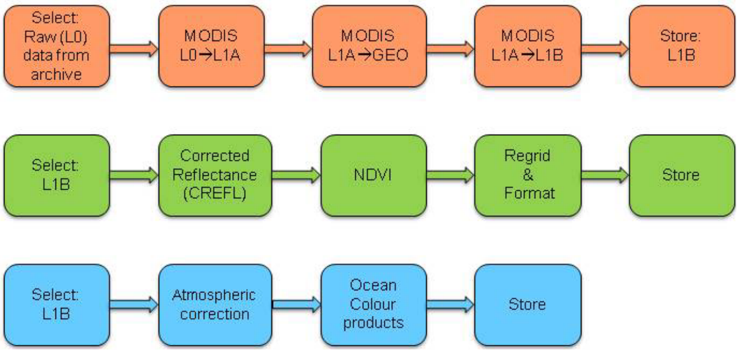


Figure 1. Illustration of remote sensing processing workflows

Similarly, the automation of workflows within a generic workflow framework enables re-use of much of the control logic and data handling. The framework thus provides a platform or environment within which application specific tasks, such as specialized remote sensing processing, can be combined with more general steps like statistical computation, spatial masking or visualization. Moreover, it can offer pre-defined templates for future workflow components. In the next section, we report the development of such a workflow service framework.

3. WORKFLOW ENGINE FOR REMOTE SENSING DATA PROCESSING

There are many workflow tools that can be used to implement a generic workflow service framework, including Kepler (Ludäscher et al., 2006), Pegasus (Deelman et al., 2005), and Taverna (Oinn et al., 2004). They are used in different application areas to perform a reliably repeatable sequence of operations. Many of these systems exist to support workflows in particular domains. They manage tasks such as automatic routing, partially automated processing and integration between different functional software applications and hardware systems that contribute to the value-addition process underlying the workflow. There are also a number of related solutions that are focused on processing of remote sensing data. Some of them target imagery data (Biesemans et al., 2007; Zheng et al., 2005), while others are more specific to complex computing process such as remote sensing quantitative retrieval (Ai et al., 2009), collaborative science workflow (Wilson et al., 2009), or remote sensing application execution (Luo et al., 2006). Among the existing systems, one of the most promising is the YABI system developed at the Centre for Comparative Genomics at Murdoch University, Australia (Hunter et al., 2011). It was developed for use in a bio-informatics context and supports linear workflows similar to those used in remote sensing processing. YABI provides a flexible mechanism for joining independent executables through a task wrapping mechanism that can be implemented in virtually any language. YABI has a number of features that make it attractive for AusCover:

- Abstraction of the processing backend to support different HPC and processing environments. This separates the workflow engine from the backend so that multiple different execution engines can be used in different steps of the workflow.
- A similar abstraction of storage backend (via scp, gridftp etc) so that workflow data can be fetched and stored from/on different remote servers.
- A very simple to use web-based user interface, designed for application users rather than computer scientists, with a capacity to record workflow configurations and parameters and archive for re-use.
- A developing capacity to be run entirely in batch mode making it suitable for the operation of canned workflows in operational (e.g. data-ingest) environments.

- A strong, well-supported and active local development team, engaged with the Australian HPC environment and ARCS, NeAT and the Australian research community.

3.1. Architecture of YABI

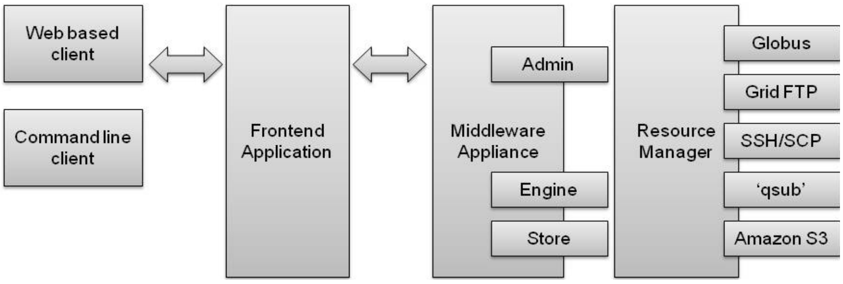


Figure 2. YABI system architecture

Figure 2 shows an overview of the YABI system architecture. There are four main components: the client, the front-end application, the middleware appliance, and the Resource manager. The client is typically a web browser, although a command line client is also under development. The Yabi front-end application is a Python web application running under Apache 2. The middleware appliance is intended to be run on an internal network that is not exposed to the Internet, though this is not an absolute requirement. It has a web based application to allow system administrators to manage the appliance and also exposes a REST style HTTP interface to the front-end application. The resource manager is a backend server daemon written in Stackless Python and makes use of the Twisted Python networking stack. It runs as a dedicated non-root user and is not intended to be network accessible by users. The resource manager is responsible for the communication with individual data and compute resources.

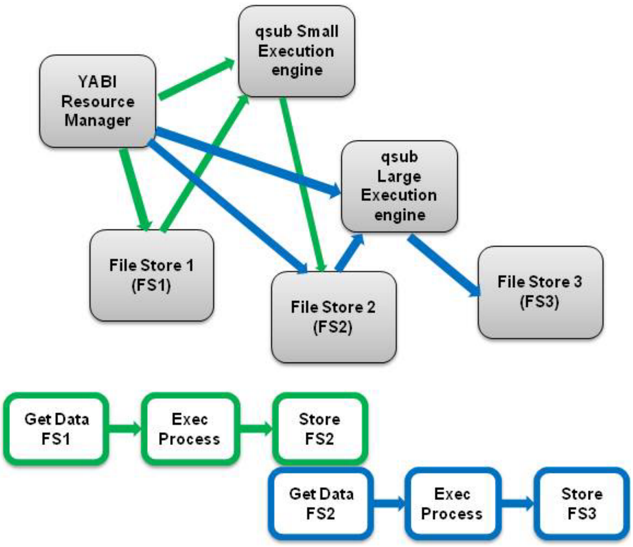


Figure 3. The YABI backend works by controlling remote storage and execution resources. In this example two tasks (green and blue) are run in sequence by YABI using two different execution engines and three file stores.

The strong abstraction of the compute and file store resources within YABI makes for a very flexible implementation because there is no requirement that all workflow tasks, or data need reside on a single machine. YABI orchestrates processing in response to a specified workflow by controlling execution and storage resources remotely via standard network and HPC protocols (Figure 3). It also means the machine upon which YABI itself runs need not be particularly powerful, since all the data intensive I/O work and computation can take place elsewhere.

3.2. RS-YABI

YABI utilizes a simple mechanism for wrapping or encapsulating existing tasks. It was originally designed to undertake processing of genetic data by separating tasks into separate executables, between which data are passed using file stores. YABI models individual tasks as execution processes requiring a set of arguments to control them. Any self-contained executable can be easily made to fit this model by wrapping it in a script that provides:

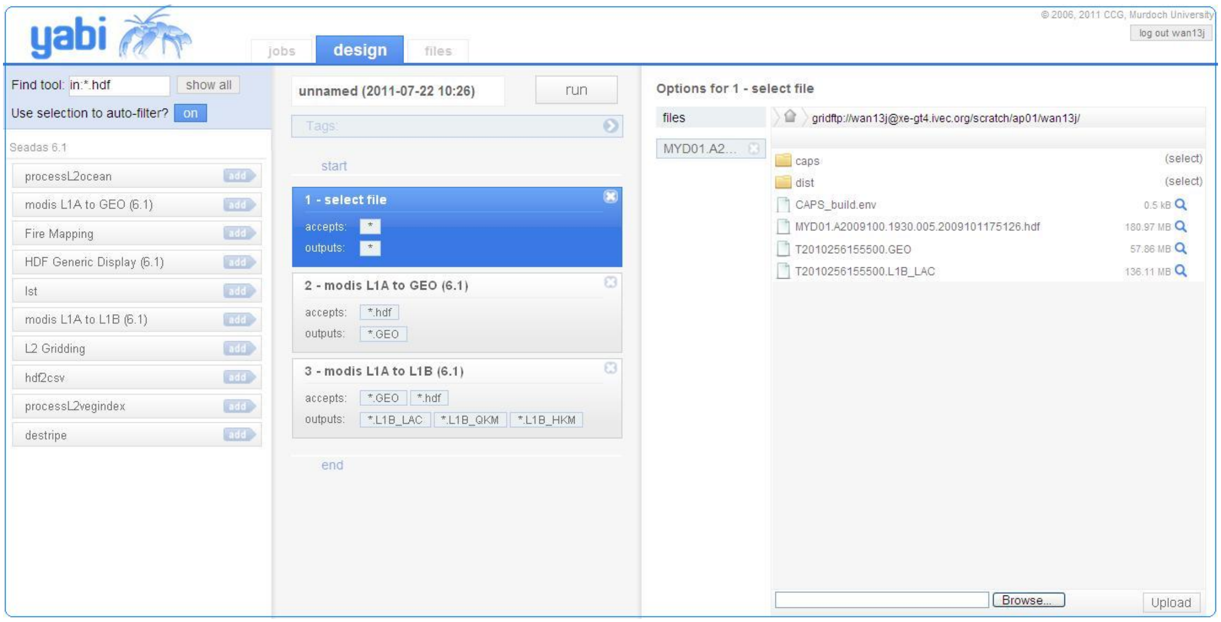
- the translation between the YABI model and the arguments required by the wrapped task;
- execution of the task and signaling of exceptions; and
- handling of outputs by naming and placing them as YABI requires.

The model enables YABI to establish which parameters are mandatory, what task dependencies exist, and how to connect tasks into a workflow.

The SeaDAS task flow (e.g. Figure 1) is a natural fit to this model. A large part of the work in developing this system for remote sensing use has involved the wrapping of SeaDAS and other related software components. We have written several script wrappers (in Python, Perl or Shell scripting language) to interface with the remote sensing data tools and allow end-users to provide input values. New modules (tasks) can be added to YABI via a web interface in the administration pages, or by importing JSON files that provide detailed descriptions of the inputs and outputs at the same interface. We refer to the resulting system, YABI together with the suite of remote sensing (RS) task modules, as RS-YABI.

Figure 4 shows a screen shot of a simple workflow within the RS-YABI user interface. There are three tabs at the top of the screen labelled jobs, design and files. The "jobs" tab provides access to the results of previously run jobs. Data from intermediate steps may be accessed, and saved jobs may be altered and re-used. The "design" tab, as shown in Figure 4, presents the user with a list of modules on the left hand side, where each module may be selected in turn to compose a workflow in the middle panel of the screen. The workflow in Figure 4 begins with a file selection, followed by generation of GEO and L1B MODIS files, similar to the first workflow illustrated in Figure 1. The "files" tab provides a simple user interface for access to data sets on the network.

The development version of RS-YABI focused on the creation of tools specifically for working with MODIS data and enabled the building of workflows for a range of products (Fearns et al., 2011). The development version was created simply by adding the remote sensing modules to an already-working installation of YABI on iVEC infrastructure at the Murdoch University in Western Australia. The remote sensing data processing applications embedded within RS-YABI include components from several freely available remote sensing data tools other than SeaDAS, such as IMAPP, IPOPP, ms2gt, mrtswath, and DRM.



Find tool: in:*.hdf

unnamed (2011-07-22 10:26)

1 - select file
accepts: *
outputs: *

2 - modis L1A to GEO (6.1)
accepts: *.hdf
outputs: *.GEO

3 - modis L1A to L1B (6.1)
accepts: *.GEO *.hdf
outputs: *.L1B_LAC *.L1B_QKM *.L1B_HKM

MYD01.A2...

gridftp://wan13j@xe-gt4.ivec.org/scratch/ap01/wan13j/

caps (select)
dist (select)
CAPS_build.env 0.5 kB
MYD01.A2009100.1930.005.2009101175126.hdf 180.97 MB
T2010256155500.GEO 57.86 MB
T2010256155500.L1B_LAC 136.11 MB

Figure 4. Screen shot of a workflow set up in RS-YABI GUI

4. DEPLOYMENT OF RS-YABI

The YABI system was designed to be deployable as an "appliance," with required software, middleware and server hardware able to be installed at a site and managed remotely, if required. The appliance can interface with local high performance computing systems and/or submit compute jobs to the HPC infrastructure. The design of appliance has the following benefits over traditional software applications that are installed on top of an operating system:

- *Simplified deployment*: A software appliance encapsulates an application's dependencies in a pre-integrated, self-contained unit. This can dramatically simplify software deployment by freeing users from having to worry about complex technique such as resolving potentially complex OS compatibility issues, library dependencies or undesirable interactions with other applications.

- *Improved isolation*: software appliances are typically used to run applications in isolation from one another. If an appliance's security is compromised, or if the appliance crashes, other isolated appliances will not be affected.

We have commenced deployment of an instance of a RS-YABI appliance within a self-contained Virtual Machine (VM) at the NCI infrastructure in Canberra. The abstraction of file and execution backends has enabled a phased approach deployment, allowing it to be undertaken in a way that greatly simplifies the process by breaking it into small steps. Initial testing of the RS-YABI engine and frontend application installed in the VM took place by remotely using the workflow tasks already installed on the Murdoch development system. Once these were demonstrated to be working correctly, we progressively enabled access to the NCI file and compute resources. All the module wrappers created in the development system were straightforwardly moved to the NCI with only a small number of local details, such as pathnames of executables, having to be changed.

5. USE CASES

RS-YABI has begun to be used for several applications, two of which we briefly describe here.

Advection of mineral dust is important for its effects on crops, radiative forcing and public health. For dust originating in remote arid regions, remote sensing is an effective tool for monitoring its movement and understanding its genesis and transport. The MODIS instrument is sensitive to airborne dust and we have incorporated the necessary processing steps as tasks within RS-YABI (Broomhall et al., 2011). By building on the simple workflows for preprocessing MODIS data (described earlier), this has enabled the efficient implementation of a standardized consistent procedure for dust detection and monitoring which can be applied to both the MODIS historical archive as well as contemporary data.

Smoke from environmental fires is frequently present in the Australian atmosphere. Detection and monitoring of smoke plumes is important not only for understanding the potential health impacts on the human population, but also is an indicator of the presence of active fires, especially in the sparsely populated regions of Northern Australia. An algorithm for differentiating smoke plumes from the background scene in MODIS imagery, based on a series of threshold tests, has been encoded as a task in RS-YABI. It has been successfully applied to two events in Western Australia in 2009 and 2011 as described in Chedzey et al. (2011).

These two examples illustrate the ease with which the RS-YABI framework facilitates the implementation of new remote sensing products by providing a framework within which they can be built. Of particular importance is the way that it facilitates the re-use of existing processing steps, and the provision of a simple and easy-to-use interface to control and monitor the processing.

6. CONCLUSION AND FUTURE DIRECTIONS

The YABI workflow engine is a tool that orchestrates workflows by organizing data and controlling the execution of stand-alone tasks by making use of remote storage and execution resources. This makes it a very effective tool for data intensive computing because it can access substantial HPC resources remotely. Furthermore, the wrapping-of-executables task model provides great flexibility because existing applications can be easily incorporated within the YABI framework in their native environment; that is, there is no need to port application software to the same environment in which the YABI engine itself runs. We have added numerous modules that perform remote sensing processing to an existing YABI installation to create RS-YABI, demonstrating the effectiveness of the model.

A virtual laboratory for satellite data processing is being established at the NCI to improve the management and processing of large National remote sensing archives. A production version of an RS-YABI appliance has been installed in a VM at the NCI to support this work. By providing easy user access to pre-defined standard processing workflows, RS-YABI will greatly simplify the access and use of previously difficult to use data sets for a wide range of researchers. In concert with the creation of the National archive, we will progressively add more tools to RS-YABI that provide specific features to support other sensors. Of immediate interest are the long time series (since 1990) of daily NOAA/AVHRR data, and the forthcoming data from the VIIRS sensor which is due for launch in late 2011 and is intended to replace MODIS over the next decade.

A focus of our work over the next six months will be to develop sustainable management practices for the administration of the RS-YABI instance, including the establishment of appropriate conventions for source control and documentation of the remote sensing task modules. Our aim is to establish the system as a tool of choice for routine processing of data and to simplify the development of new products through close interaction with the research user community. We will explore the possible deployment of a dedicated RS-YABI system at the iVEC data-intensive facility in Western Australia where there is an AusCover node and a strong community making use of remote sensing data.

ACKNOWLEDGMENTS

We are particularly grateful to Muhammad Atif, a data intensive computing specialist at the NCI, and to the head of the NCI Ben Evans for facilitating and assisting with the installation of the RS-YABI instance.

REFERENCES

Ai, J., Y. Xue, Y. Li, J. Guang, Y. Wang, and L. Bai (2009). A dynamic grid workflow for remote sensing quantitative retrieval service. *Proc. IEEE International Geoscience and Remote Sensing Symposium*, 61-64.

Biesemans, J., S. Sterckx, E. Knaeps, K. Vreys, S. Adriaensen, J. Hooyberghs, K. Meuleman, P. Kempeneers, B. Deronde, and J. Everaerts (2007). Image processing workflows for airborne remote sensing. *Proc. 5th EARSeL Workshop on Imaging Spectroscopy*.

Broomhall, M., H. Chedzey, R. Garcia, M. Lynch, P. Fearns, E. King, Z. Wang, G. Smith, and D. Schibeci (2011). Ensemble dust detection techniques utilising a web-based workflow environment linked to a high performance computing system. *34th International Symposium for Remote Sensing of the Environment (ISRSE),* Ref.416.

Chedzey, H., P. Fearns, M. Broomhall, R. Garcia, M. Lynch, E. King, Z. Wang, G. Smith (2011). Processing smoke plume products from the Moderate Resolution Imaging Spectroradiometer (MODIS) within a workflows environment. *34th International Symposium for Remote Sensing of the Environment (ISRSE),* Ref.416.

Deelman, E., G. Singh, M. H. Su, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, G. B. Berriman, and J. Good (2005). Pegasus: A framework for mapping complex scientific workflows onto distributed systems. *Scientific Programming, 13*(3), 219-237.

Fearns, P., M. Bellgard, M. Broomhall, H. Chedzey, R. Garcia, A. Hunter, et al. (2011). Web-based processing of remote sensing data in a workflows environment. *34th International Symposium for Remote Sensing of the Environment (ISRSE),* Ref. 488.

Georgakopoulos, D., M. Hornick, and A. Sheth (1995). An overview of workflow management: from process modeling to workflow automation infrastructure. *Distributed and parallel Databases, 3*(2), 119-153.

Hunter, A., A. Macgregor, T. Szabo, C. Wellington, and M. Bellgard (2011). Yabi: A sophisticated online research environment for Grid, High Performance and Cloud computing.

Lillesand, T. M., R. W. Kiefer, and J. W. Chipman (2004). *Remote sensing and image interpretation*: John Wiley & Sons Ltd.

Ludäscher, B., I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. Lee, J. Tao, and Y. Zhao (2006). Scientific workflow management and the Kepler system. *Concurrency and Computation: Practice and Experience, 18*(10), 1039-1065.

Luo, Y., Y. Xue, C. Wu, Y. Hu, J. Guo, W. Wan, L. Zheng, G. Cai, S. Zhong, and Z. Wang (2006). A remote sensing application workflow and its implementation in remote sensing service grid node. *Computational Science–ICCS 2006*, 292-299.

Oinn, T., M. Addis, J. Ferris, D. Marvin, M. Greenwood, T. Carver, M. Pocock, A. Wipat, and P. Li (2004). Taverna: A tool for the composition and enactment of bioinformatics workflows. *Bioinformatics, 20*(17).

Russell, N., A.H.M. ter Hofstede, D. Edmond, and W.M.P. van der Aalst (2005). Workflow Data Patterns: Identification, Representation and Tool Support, *LNCS 3716,* 353-368.

Van Der Aalst, W., and K. M. Van Hee (2004). *Workflow management: models, methods, and systems*: MIT press.

Wilson, B. D., R. Ramachandran, and C. Lynnes (2009). Talkoot portals: Discover, tag, share, and reuse collaborative science workflows. *Proc. American Geophysical Union Spring Meeting*.

Zheng, R., H. Jin, Q. Zhang, and Y. Li (2005). Workflow-based remote-sensing image processing application in ImageGrid. *Proc. Sixth International Conference on Parallel and Distributed Computing Applications and Technologies*.