

Accepted Manuscript

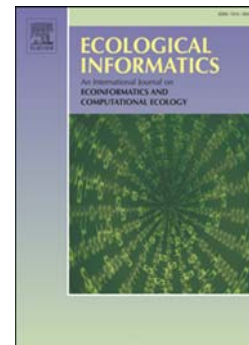
Development of a cloud-based platform for reproducible science: A case study of an IUCN red list of ecosystems assessment

Siddeswara Guru, Ivan C. Hanigan, Hoang Anh Nguyen, Emma Burns, John Stein, Wade Blanchard, David Lindenmayer, Tim Clancy

PII: S1574-9541(16)30118-2
DOI: doi: [10.1016/j.ecoinf.2016.08.003](https://doi.org/10.1016/j.ecoinf.2016.08.003)
Reference: ECOINF 701

To appear in: *Ecological Informatics*

Received date: 24 March 2016
Revised date: 15 August 2016
Accepted date: 20 August 2016



Please cite this article as: Guru, Siddeswara, Hanigan, Ivan C., Nguyen, Hoang Anh, Burns, Emma, Stein, John, Blanchard, Wade, Lindenmayer, David, Clancy, Tim, Development of a cloud-based platform for reproducible science: A case study of an IUCN red list of ecosystems assessment, *Ecological Informatics* (2016), doi: [10.1016/j.ecoinf.2016.08.003](https://doi.org/10.1016/j.ecoinf.2016.08.003)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Development of a cloud-based platform for reproducible science: a case study of an IUCN Red List of Ecosystems Assessment

Siddeswara Guru^{a*}, Ivan C. Hanigan^a, Hoang Anh Nguyen^b, Emma Burns^{c,d}, John Stein^c, Wade Blanchard^c, David Lindenmayer^{c,d}, Tim Clancy^a

^aTerrestrial Ecosystem Research Network, University of Queensland, St Lucia, 4072, Australia

^bResearch Computing Centre, University of Queensland, St Lucia, 4072, Australia

^cFenner School of Environment and Society, The Australian National University, Canberra, ACT 2601, Australia

^dLong Term Ecological Research Network, Terrestrial Ecosystem Research Network, Australia

ARTICLE INFO

Article history:

Received 00 December 00

Received in revised form 00 January 00

Accepted 00 February 00

Keywords:

IUCN Ecosystems Assessment

Kepler workflow

Cloud Computing

Platform as a Service

ABSTRACT

One of the challenges of computational-centric research is to make the research undertaken reproducible in a form that others can repeat and re-use with minimal effort. In addition to the data and tools necessary to re-run analyses, execution environments play crucial roles because of the dependencies of the operating system and software version used. However, some of the challenges of reproducible science can be addressed using appropriate computational tools and cloud computing to provide an execution environment.

Here, we demonstrate the use of a Kepler scientific workflow for reproducible science that is sharable, reusable, and re-executable. These workflows reduce barriers to sharing and will save researchers time when undertaking similar research in the future.

To provide infrastructure that enables reproducible science, we have developed cloud-based Collaborative Environment for Ecosystem Science Research and Analysis (CoESRA) infrastructure to build, execute and share sophisticated computation-centric research. The CoESRA provides users with a storage and computational platform that is accessible from a web-browser in the form of a virtual desktop. Any registered user can access the virtual desktop to build, execute and share the Kepler workflows. This approach will enable computational scientists to share complete workflows in a pre-configured environment so that others can reproduce the computational research with minimal effort.

As a case study, we developed and shared a complete IUCN Red List of Ecosystems Assessment workflow that reproduces the assessments undertaken by Burns et al. (2015) on Mountain Ash forests in the Central Highlands of Victoria, Australia. This workflow provides an opportunity for other researchers and stakeholders to run this assessment with minimal supervision. The workflow also enables researchers to re-evaluate the assessment when additional data becomes available. The assessment can be run in a CoESRA virtual desktop by opening a workflow in a Kepler user interface and pressing a “start” button. The workflow is pre-configured with all the open access datasets and writes results to a pre-configured folder.

Highlights:

- A cloud-based infrastructure was developed to build, execute and share computational-centric research. The infrastructure offers analysis tools as a “Platform as a Service” in a virtual desktop offered as a “Desktop as a Service” at <https://www.coesra.org.au>.
- A repeatable Kepler workflow from CoESRA was built and shared to reproduce and share the IUCN Red list of Ecosystems Assessment for Mountain Ash forests located in the Central Highlands of Victoria, Australia.

© 2016 xxxxxxxx. Hosting by Elsevier B.V. All rights reserved.

1 Introduction

* Corresponding author. Tel.: +61-7-33463125; fax: +0-000-000-0000.

E-mail address: s.guru@uq.edu.au

Peer review under responsibility of xxxxx.

xxxx-xxxx/\$ – see front matter © 2016 xxxxxxxx. Hosting by Elsevier B.V. All rights reserved.

Computation and software tools are used extensively in scientific analyses and the ensuing scientific publications are intended to disseminate data analyses and interpret their results. However, journal publications largely fail to provide sufficient information to generate verifiable knowledge (Donoho et al., 2009). According to Pritsker (2012), it is estimated that only 10%–30% of published studies are repeatable, although Freedman et al. (2015) claim that 51%–89% of all published research is not reproducible, and that one in four data analyses contribute to this irreproducibility. Specifically, data analysis steps are often not explained adequately in the literature for an independent researcher to reproduce the results. To quantify the cost of reproducibility, in one of the experiments, Garjito et al. (2013) estimated that it took 280 hours to reproduce the results of a computational biology paper and concluded that only expert researchers in the domain could do it (Garjito et al., 2013).

Poor reproducibility has generated considerable concern and debate within the scientific community (Yamada and Hall, 2015) (Errington et al., 2014), and among funding agencies (Budget, 2014) (Collins and Lawrence, 2014) and scientific journal publishers (Editorial, 2012) (Jasny et al., 2011). The prevailing concern is that research might lack credibility and can not be trusted, and that poor reproducibility of scientific research precludes building on previous studies and leads to the loss of scientific efficiency (Marcus, 2014).

The seriousness of this problem has caused US National Institutes of Health (NIH) to initiate a plan to enhance reproducibility (Collins and Lawrence, 2014), and it was estimated that US \$28 billion per year is spent on preclinical research that is not reproducible (Freedman et al., 2015). Measures have also been taken to improve reproducibility in other disciplines such as life sciences (Collins and Lawrence, 2014) and psychology (Asendorpf et al., 2013). More generally, there have been calls for all scientific claims to be completely repeatable by independent investigators who ‘address a hypothesis and build evidence for or against it’ (Peng, 2011). However, it is important to note that the exact results need not be computed in a repeatable/replicated study. This is because replication involves probability, and if performed again with a different sample and a new set of measurement errors, some variance between experiments is expected.

To date, concepts such as reproducibility and replicability have been defined in a number of ways. According to Cassey and Blackburn (2006), reproducibility is where the information presented in the study can be used by a third party to obtain the same results. This definition distinguishes reproducibility from replicability, which indicates that ‘a third party must be able to perform a study using identical methodological protocols and analyse the resulting data in an identical manner for independent verification’ (Cassey and Blackburn, 2006). Leek and Peng (2015) define reproducibility as ‘the ability to recompute the results’ and replicability as ‘the chance that another experimenter can independently perform the experiment and achieve consistent results’ (Leek and Peng, 2015). Peng (2011) described reproducibility and replication in a reproducibility spectrum where full replication is the highest form of reproducibility, and suggest that sharing code and data will only lead to partial reproducibility (Peng, 2011). From an experimental science perspective, reproducibility is more about using the information provided by original researchers to generate results, and replicability is where researchers independently recreate the experiment and get similar results.

In essence, reproducibility provides assurance to readers that the claims made in a publication are valid (Mesirov, 2010), and therefore improves the credibility of science. However, the benefits go further because reproducibility will increase scientific advancement through greater transparency, which will enable others to understand the analysis steps and the assumptions and constraints of a given study in more detail. This will limit waste of research funds and human resources because researchers will be better able to build on previous work.

To improve reproducibility in science, several journals, including the Public Library of Science (<https://www.plos.org>) and Vadose Zone Journal (<http://vzj.geoscienceworld.org/>), now require authors of scientific papers to deposit their data in public repositories (Skaggs et al., 2015). This is the first step towards enabling reproducibility, but may not be enough to repeat the published research. The Journal of Biostatistics has a more comprehensive policy to promote reproducibility, requiring that any article published with data or a source code is given a “D” or “C” kite-mark, respectively, and an article that meets these reproducibility requirements will be given an “R” (Peng, 2011). One of the major challenges with this approach is that the onus is on the journal to ensure that papers are reproducible using the code and data provided by the authors. If analysis steps are complex, repeating the simulation could be a difficult and time-consuming task for journal editors and reviewers. Hence, although we agree with Peng (2011) that reproducibility should become the minimum standard for assessing the value of scientific claims, to achieve this goal consistently, the onus should be on the proponent of the research to make the data analysis and synthesis steps reproducible. Accordingly, researchers will need to be provided with appropriate resources, incentives, and assistance to make their data and code accessible for reproduction. In certain instances, it may not be possible to reproduce the results even with the required source code and data because computational environments to execute the source code may not be available—and/or easy to recreate. Hence, a major barrier to reproducible research is the ‘lack of an integrated infrastructure for distributing reproducible research to others’ (Peng, 2011).

In this paper, we have taken a holistic approach to reproducibility, where a complete computational analysis processes is available in an executable platform, irrespective of the complexity of the scientific research. The executable platform is accessible to independent researchers in a computation environment so that analyses can be repeated with minimal effort by other researchers. This makes reproducibility a non-tedious task for other researchers and where appropriate, could assist the original researchers in the longer-term to repeat the computational processes at different time-intervals. We therefore consider computational research as reproducible when computational experiment e that was developed at time t on a hardware infrastructure h using data d is repeated at time t' on hardware h' using data d' that is similar to d (Freire et al., 2012). Thus, we built infrastructure in which computational research r that was developed at time t on a hardware and software infrastructure h using data d was reproduced at time t' on the same hardware and software infrastructure h using the same data d . This infrastructure is accessible in a very easily consumable manner so that users don't require any special tools to access them. In addition, the analysis tools and datasets used in the original computation are conserved and accessible.

The present manuscript describes the development of the integrated cloud-based infrastructure that allows building and distribution of reproducible computational-centric research. To demonstrate the value of the approach, we repeated an International Union for Conservation of Nature (IUCN) Red List of Ecosystems assessments study conducted by Burns et al. (2015), and reproduced published results and made the computational processes of the assessment accessible to other researchers. The IUCN Red List of Ecosystems (<http://www.iucnredlistofecosystems.org>) provides a framework to assess the vulnerability of ecosystems and is akin to the IUCN Red List of Threatened Species (<http://www.iucnredlist.org/>).

Our paper is structured as follows: section 2 further identifies the requirements and challenges of reproducible research, and the benefits of using a scientific workflow and cloud computing infrastructure; section 3 describes the design, development and operation of the present Collaborative Environment for Ecosystem Science Analysis and Synthesis; section 4 provides a brief overview of the IUCN Red List of Ecosystems Assessment

conducted by Burns et al. (2015), and the implementation of this assessment as a Kepler scientific workflow; section 5 concludes the paper and is followed by a statement of necessary future work in section 6.

2 Building Blocks of Reproducible Computational Research

With the advancement of computational science, almost all analyses are performed using software tools. Therefore, the possibility of making research reproducible is quite high. However, most complex analyses involve (1) multiple datasets, (2) various algorithms and analytical tools and (3) high performance computing and/or (4) distributed computing infrastructure. There are therefore complexities and challenges to verifying scientific claims, and to providing the computational infrastructure to enable this. To address these challenges, researchers have recommended best practices for performing computational science (Sandve et al., 2013; Stodden and Miguez, 2014) (Wilson et al., 2014) (Nosek et al., 2015) (Kenall et al., 2015), and although we endorse these recommendations, we believe that the importance of the execution environment for successfully repeating any published analysis is not given the prominence it deserves. For published research to be reproducible, the following building blocks should be available:

- Data: all datasets used in the analysis must be available in an accessible digital format.
- Analysis procedures and methods: a detailed description of all the steps involved in the research and programs for analysis steps including clear interdependencies between the process and source code, and all data harmonisation and refinement routines.
- Infrastructure: this includes hardware, high performance computing, virtual machines with sufficient resources and software infrastructure (operating system, accurate version of tools used, dependent software) and software configuration (Santana-Perez and Perez-Hernandez, 2015).

2.1 Reproducibility Challenges

Reproducing any complex analysis is not straightforward for various reasons, even if the original creators make the data, software source code and tools publicly available. This is because of difficulties such as (1) version mismatch of software libraries, (2) incompatibility of different versions of the software and (3) use of licensed proprietary software in the original analysis and subsequent inability to buy licenses for the required software. The time required to address these issues may deter researchers from engaging in reproducing research. In some instances, the skill sets needed to establish execution environments and the ability to configure various analysis steps for execution may also impede a researcher's ability to make their research reproducible. In addition, it is not always easy or cost effective to access specialised computing infrastructure, such as high performance computing, when required for repeat data analyses. Therefore, for successful reproducibility of any computational analysis, a complete computational environment with access to all datasets and analytical processes is needed as an executable resource. This is more useful than disjoint source code or datasets that are made publically available in the hope that independent researchers have the time and resources to repeat the computation steps and reproduce the results.

Appropriate tools to build and repeat computational analyses would certainly facilitate reproducible science. Among these, the scientific workflow tool provides an opportunity to make complete computational research a single executable workflow that can be reused and re-executed. In addition, cloud computing provides a flexible approach to provision of computing and storage infrastructure as a virtual environment for computational research. Specifically, the use of scientific workflows to build computational research, and the ability to host these virtual experiments on a public cloud platform provides a unique opportunity to create an integrated infrastructure that can be shared and used to re-execute analysis and synthesis steps with minimal cost.

In this paper, we have taken an approach that provides integrated infrastructure to create and host computational research using a scientific workflow, and offer these tools on a sharable cloud infrastructure. Initially, we provide an overview of scientific workflow and cloud computing and then describe the overall approach to building an integrated infrastructure for reproducible research.

2.2 Scientific workflow

A scientific workflow is a systematic organisation of well-defined tasks/activities to achieve outcomes (Talia, 2013). These tasks may include retrieving data from an online data repository, feeding data into specific algorithms for analysis, visualising and storing results for further evaluation and assessment. Visually, scientific workflows provide a simple way of providing a high-level abstract view of a scientific computation, while hiding the underlying details. Workflows can integrate existing software routines, scripts, services and various datasets into 'compositions' that implement computational research. Moreover, scientific workflows can be used to preserve the complete computational research as an artefact of input data and analysis processes, and make it an executable entity. This will also enable the scientific community to examine and study the execution of any piece of research. Hence, workflows can play a major role in sharing and reusing of research methods. However, computational-resources are required for workflow execution and these are not always readily available to researchers.

Several workflow management systems are used in scientific disciplines, including Taverna (Oinn et al., 2006), Kepler (Altintas et al., 2004), Galaxy (Cock et al., 2013) and VisTrails (L. Bavoil et al., 2005). However, simply creating and sharing workflows is not enough to make computational research reproducible. For example, a study of the Taverna workflows published on myExperiment.org showed that nearly 80% of tested workflows failed to execute, and 50% of these failed due to lack of access to third party resources (e.g. web services and databases) (Jun et al., 2012). Therefore, if workflows are to be used as a credible reproducible and reusable computational environment, they need to be delivered as an executable platform with all the open access datasets linked to the analysis routines and dependent software pre-configured for smooth execution with sufficient computational resources. This approach will make reproducibility using a workflow as simple as accessing an execution platform, selecting the workflow for execution, and pressing the 'execute' button in that workflow environment.

2.3 Cloud Computing

Cloud computing enables easy access to computation resources, thus providing executable environments that can be serviced via the Internet. This allows computation and storage resources to be provided on-demand in an 'Infrastructure as a Service' model. Specifically, platforms (i.e. computing

environments and software suites) can be built on cloud compute and storage infrastructure and made available to users as a service. This is called a ‘Platform as a Service’ (Schlicher et al., 2015) and it provides an execution platform for users and removes the need to set up the execution environment to run data analyses. The entire platform can be provided as a virtual desktop accessible via a web browser, thereby enabling users to access a complete desktop environment with storage from their web browser without requiring the download of customised software.

3 Collaborative Environment for Ecosystem Science Research and Analysis (CoESRA)

We propose the platform Collaborative Environment for Ecosystem Science Analysis and Synthesis (CoESRA) as an integrated infrastructure to improve access to computing and storage resources with appropriate tools for building, executing and sharing repeatable workflow-based computational research. Users can access the infrastructure as a virtual desktop that comprises the Kepler workflow environment and support programming languages such as R, Python and Java. The virtual desktop also facilitates distributed computing using Nimrod/G (Abramson et al., 2008) and has the ability to capture and store workflow execution history using provenance. CoESRA is built on a research cloud platform offered by the National eResearch Collaboration Tools and Resources (NeCTAR) (www.nectar.org.au) and Research Data Service (RDS) (www.rds.edu.au).

In CoESRA, Kepler is used for workflow development and execution environment, and a workflow in Kepler comprises various ‘actors’ connected through channels, which enable communication between actors (Altintas et al., 2004). Actors are the building blocks of a Kepler workflow that represent complex analyses into series of interdependent processing steps. Kepler has a large pool of reusable actors that perform specific tasks across disparate execution environments. The actors can be programmed in R, Python, Java and Matlab programming languages (Ludäscher et al., 2006). Initially, Kepler was developed for ecology applications. Hence, a large number of components (actors) can be re-used in biodiversity, ecology and environmental science.

3.1 CoESRA System Functionalities

Computational platforms of CoESRA are virtual desktops that are accessible from a web browser. Each virtual desktop runs an identical software stack to build, execute and share workflow-based scientific analysis and synthesis activities. The overall CoESRA system performs the following core functionalities:

- User registration: all users using CoESRA need to register with the system. This enables the system to create user accounts for the virtual desktop;
- Virtual desktop management: this includes creation of virtual desktops with pre-configured software suites, and assignment of a working virtual desktop to a user when a request is made;
- System access: CoESRA virtual desktops are accessible via the web portal, which allows users to register and access a virtual desktop.

3.2 CoESRA System Design, Development and Operation

The high-level architecture of the CoESRA system is shown in Figure 1. The system is designed so that all components interact with one another independently. The CoESRA system comprises a number of virtual desktops that are managed by Pool Manager; all the functionalities of Pool Manager are invoked using the Pool API. The Lightweight Directory Access Protocol (LDAP) (Zeilenga, 2006) is used to create and store user information for system internal use. The DaaS Admin service interacts with the web portal to provide users with access to virtual desktops. Detailed explanations of all components of the system architecture are described below with an explanation of the system’s core functionalities.

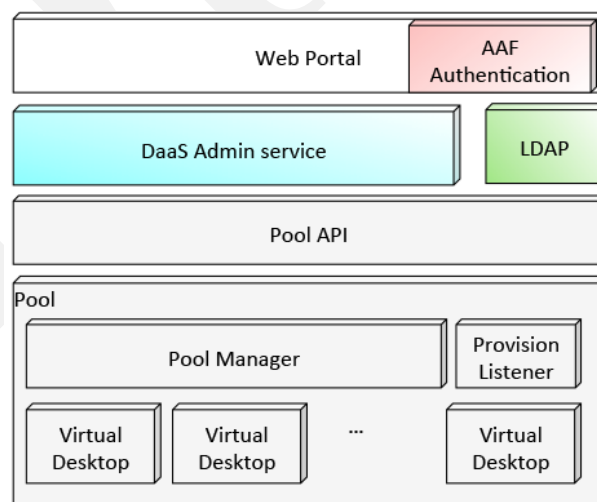


Figure 1: High-level CoESRA system Architecture with major components

3.2.1 User registration

Users are authenticated using Australian Access Federation (AAF, 2016) identity management; however, users from institutions that do not support Australian Access Federation will need to contact the CoESRA administrator to be provided with a guest account. Details of the process for obtaining a guest account are available at <https://www.coesra.org.au/#/faq>, and can be permanent or temporary depending on user preference. Once a user registers to

a system, the user account creation process creates a user entry in LDAP and a home folder on the network file system server (cloud-based storage), and updates user information in the user database. Subsequently, a machine-generated email is sent to the user confirming the successful completion of the registration process.

3.2.2 Virtual Desktop Management

An essential ingredient of the CoESRA system is access to a virtual desktop offered as a “Desktop as a Service”. The two software components, Pool Manager and DaaS Admin Service accomplish virtual desktop management functionalities. The Pool Manager runs as a daemon process in the background and performs the core functionality of checking the number of free virtual desktops in the pool and then provides a virtual desktop if the number is less than the threshold. This provision happens by (1) creating a virtual machine (VM), and (2) deploying customised tools and software (platform). A virtual machine is created using a Boto programming interface (Garnaat, 2009) and the system configuration tool Puppet is used to deploy all tools and packages (Loope, 2011). Although users are allowed to do anything within their user space, only the system administrator can update the system tools. This is to ensure the consistency of all the desktops in the system. Figure 2 shows the components of the virtual desktop. A custom build “Application desktop service” is also installed in each of the virtual desktops. This service is programmed to assess the state of the virtual desktop before allocating it to any user.

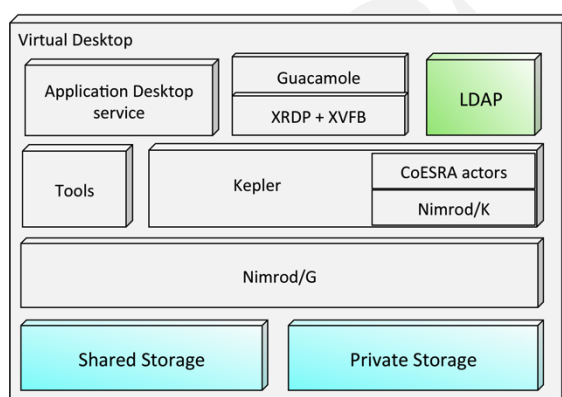


Figure 2: different components of the CoESRA virtual desktop

The virtual desktop launching process occurs asynchronously and is primarily based on the number of free desktops available in the pool. Thus, to ensure quick responses to user requests, virtual desktops are always ready for use and are not generated on-demand. In the CoESRA system, to manage the cloud resource efficiently, a minimum and maximum limit is set for the number of virtual desktops available for users. Thus, the system always has the minimum number of virtual desktops, and if the number of free virtual desktops (not in use) is below the set threshold, the Pool Manager will start the process of creating new virtual desktops. For each virtual desktop successfully created, a domain name is assigned and added to the pool, and virtual desktops are found in one of three states as follows: (1) Ready, the virtual desktop is provided and ready for use; (2) Busy, the virtual desktop is in use; and (3) Error, the virtual desktop is not working appropriately. The Pool Manager checks the status of the virtual desktop at regular intervals and deletes instances that are in the error state and starts the process of creating new instances if free virtual desktops are below the lower threshold limit.

The detailed working architecture of the CoESRA system is shown in Figure 3 and includes communications between services. The DaaS admin service (Figure 3) administers the virtual desktop pool. Following a user request, a virtual desktop access is initiated through a web portal and the DaaS admin service receives notification with the prospective user’s Australian Access Federation credentials. Subsequently, the DaaS admin service searches for LDAP credentials of the corresponding user, and checks the pool to find any free virtual desktops. If there are free virtual desktops available, the first available in the list is selected and an assignment request is sent to the Application desktop service (Figure 3) of that virtual desktop. Prior to assigning a virtual desktop to a user, the service checks access to the network file system server, enables and checks access to a user home folder, checks for all services that are required to effectively run a virtual desktop, and finally authenticates the user with the internal LDAP. Once all the checks are performed satisfactorily, a virtual desktop domain name is sent as a response to the DaaS admin service request. A web browser then opens the virtual desktop of a respective domain name and makes it available for the user. If any one of the checks fails, the DaaS admin service selects the next virtual desktop in the free pool and repeats the process. If there are no free virtual desktops in the pool, the DaaS admin service will pass an error message to the user stating that there are no free virtual desktops available.

3.2.3 CoESRA System Access

The CoESRA is accessible at <https://www.coesra.org.au>. The system works best in the Google Chrome web browser but will provide sub-optimal performance in Mozilla Firefox and Internet Explorer. The virtual desktop has access to two separate public shared folders to share datasets and workflows, respectively. Thus, users can publically share workflows and corresponding data and can contact CoESRA system administrators to publish workflows as part of the platform. The Kepler environment also records provenance information about the workflow execution so that user can always access the provenance database and re-run the precise instance of a previous simulation. Users can also store private data and workflows for later use in private home folders.

To manage efficient use of cloud resources, virtual desktops are allocated to users for 48 hours only. Six hours before the expiry of their allocation, users will receive an email notification stating that the virtual desktop allocation will expire in six hours and that the user has the option to request an extension by replying to the email. Following expiry of access to the virtual desktop, users can log on to the CoESRA again and access another free virtual desktop from the pool.

A workflow publishing functionality is offered from the Kepler environment to share workflows on the Research Data Australia portal (www.researchdata.ands.org.au) and in myExperiment.org. Detailed instructions on how to share workflows in these portals are provided in FAQs (<https://www.coesra.org.au/#/faq>). The sharing process will enable workflows to be discoverable from the Australian national data and tools portal, and from myExperiment.org.

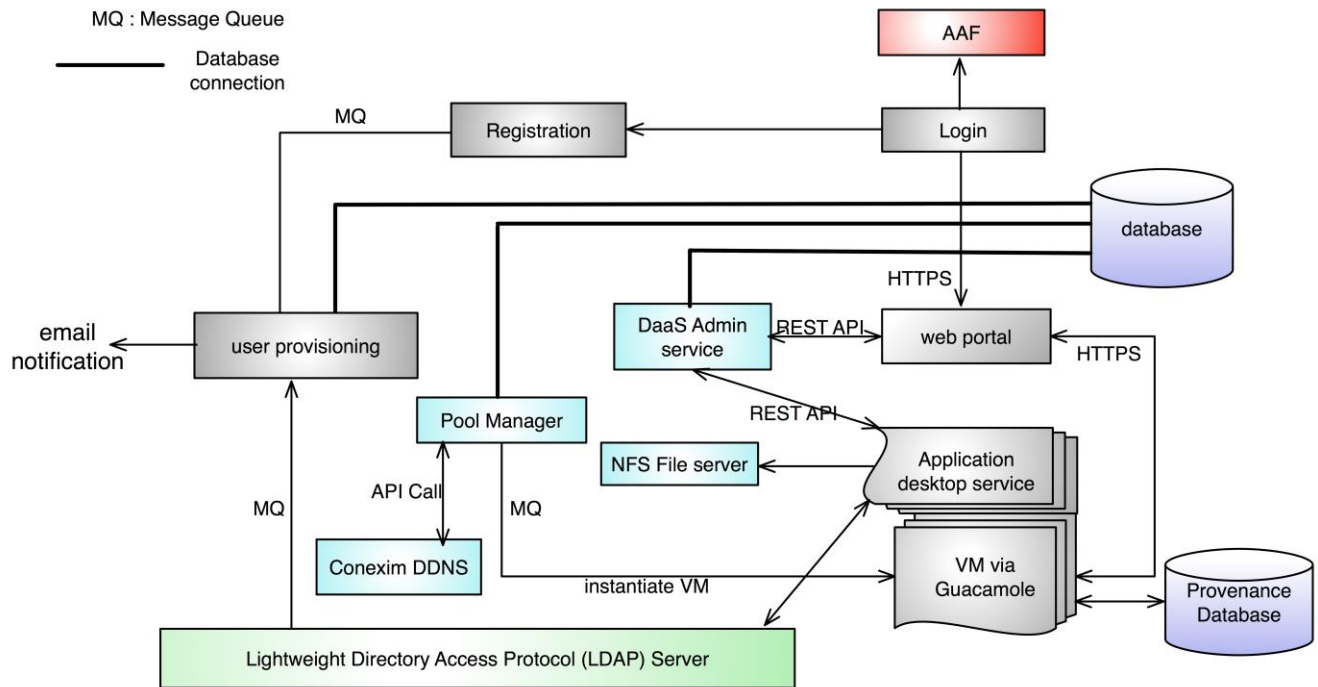


Figure 3: Detailed working architecture of CoESRA

4 IUCN Red List of Ecosystems Assessment

The International Union for Conservation of Nature (IUCN) has developed a Red List of Ecosystems (RLE) framework to methodically assess the status of an ecosystem based on characteristic biota and ecological processes (Keith et al., 2013) (Rodríguez et al., 2011). These frameworks improve the ability to monitor and manage biodiversity at local, national and global levels (Rodríguez et al., 2015). Under the IUCN RLE, the ecosystem is assessed for the risk of collapse based on criteria A–E (Keith et al., 2013) as follows: “A) rates of decline in ecosystem distribution; B) restricted geographic distribution with continuing declines or threats; C) rates of environmental (abiotic) degradation; and D) rates of disruption to biotic processes”. The fifth criteria (E) quantifies the risk of ecosystem collapse. The four criteria (A–D) have three sub-criteria for assessment and outcomes of these criteria range from Least Concern (LC) to Critically Endangered (CR). The overall evaluation is determined according to the most severe ranking of the five criteria (Rodríguez et al., 2015).

Ecosystems evolve over time and are dynamic in nature. Moreover, ecosystem change can be dramatic or gradual depending on the drivers and processes (natural and/or human-induced) causing change, and the time frames and scales within which change is occurring. Severe or repeated disturbance events can cause ‘ecosystem collapse’, leading to loss of the core functions and biota that characterise the system. Alternatively, recovery follows disturbances, but the pace and form of this recovery varies depending on the intensity and type of disturbance. Monitoring and assessing ecosystem ‘health’ is therefore challenging and necessarily dynamic. Ecosystems need to be monitored repeatedly and therefore ecosystem assessments need to be repeated as new information becomes available or as the underlying modelling frameworks to inform them are adjusted (e.g. revised climate modelling). Re-evaluation is more difficult if the processes and underlying data of the previous evaluations are not conserved. That is, it is more difficult and time-consuming to rebuild the assessment processes every time a new evaluation needs to be conducted. It is also important for the evaluation processes to be transparent and repeatable so that stakeholders can critique and debate the assessment and its implications for ecosystem management.

4.1 Case Study: Mountain Ash Forest IUCN Assessment

The Mountain Ash forest is located in the Central Highlands of Victoria, Australia. The forest is home to the world’s tallest flowering tree (Ashton, 1975), and is highly valued for its biodiversity, timber production, and water. This area is also the primary catchment for water-provision to Melbourne (Viggers et al., 2013), and the Mountain Ash forest is also one of the most carbon-dense forests in the world (Keith et al., 2009). However, severe repeated wildfire and widespread logging have impacted the forest ecosystem (Lindenmayer et al., 2011). Hence, old growth forest within the area has diminished considerably in comparison with its historical extent. Due to the unique biodiversity of the Mountain Ash forest and its usefulness, there is a strong need to assess the ecosystem and conserve its ecosystem services for long-term social and economic interests.

4.2 The Original Ecosystem Risk Assessment

Burns et al. (2015) conducted the Mountain Ash forest ecosystem risk assessment using the IUCN Red List of Ecosystems framework described in (Keith et al., 2013) and (Rodriguez et al., 2011). According to their assessment, there is a 92% to 99% chance that the ecosystem will collapse by 2067. The original assessment was undertaken by a multi-disciplinary team comprising field ecologists, a statistician, a Geographic Information System (GIS) expert, and a IUCN RLE subject matter expert, and data analyses were performed mostly using Esri ArcGIS (ArcGIS, 2016) and R programs. Most analysis steps to assess criteria and sub-criteria were disjointed and completed in a manual interactive mode where the statistician and the GIS expert were intimately involved in running software tools with input data and executing R programs. The two sub-criteria in criterion C were not assessed due to data deficiency.

There were several challenges to reproducibility of the complete assessment as a workflow-based computational research outcome. First, informaticians creating workflows needed to understand how the analysis was performed. For example, detailed instructions were gathered on how Esri ArcGIS software was used to perform some of the analysis. However, because ArcGIS is licenced software, it is not possible to use it to reproduce results if a researcher doesn't have a valid licence. Therefore, as part of the automation process, R programs were written to perform the analysis steps conducted in ArcGIS.

Second, all the primary data required in the assessment needed to be acquired from different sources and refined for use in the assessment. For example, a data access agreement was required to gain access to the primary datasets from VicForests, CSIRO and the Australian National University (ANU). Forest datasets are owned by VicForests, whereas the climate variable data is owned by the prestigious Australian Government research organisation CSIRO. A researcher based at the Australian National University (ANU) owns the long-term observational field-data of the hollow-bearing trees in the Mountain Ash forest. Overall, the simulation uses close to 4 GB of primary datasets of different file types and formats, and it is generally difficult to share such large datasets and expect others to understand and use them in complex analyses.

Third, the quantitative analysis, which was developed in R programming language, was not written for third party comprehension. Hence, original creator assistance was required for any re-run and to understand the functionalities of the routines. Hence, during the development of the workflow, informaticians worked closely with the geographic information system specialist, the statistician and the subject matter expert from the original study to develop the automated Kepler workflow. Moreover, the source code used to perform quantitative analysis (criteria E) in the original assessment was reused but modularised to meet the needs of the workflow. The analysis and manipulation processes are mapped to the required datasets and each criterion is evaluated separately and then aggregated to provide the overall assessment result.

4.3 Kepler workflow for Mountain Ash forest risk assessment

In each criterion and sub-criterion of the Mountain Ash forest assessment, all computing steps, including data access, refinement, manipulation and processing, were identified and a conceptual workflow was developed. The conceptual workflow detailed in Appendix S1 formed the basis for the development of the executable workflow.

Each step in the conceptual workflow was implemented to develop an executable workflow. During implementation, modular R packages were written to perform processing steps and compute the required calculations for each criterion. To leverage the flexibility of reusing code beyond building the Kepler workflow, we developed the complete assessment using R programming language (Hanigan and Blanchard, 2015). Each criterion (A, B, C, D and E) of the Mountain Ash forest assessment was implemented as a separate R package. The main R package was called "IUCNEcosystemRiskAssessment" (Hanigan and Blanchard, 2015) and was developed to implement the components of any generic risk assessment. R packages for each criterion were also designed specifically for the Mountain Ash forest ecosystem 'IUCNcriteria_X_mountainashforests' (where 'X' is the IUCN framework module A, B, C, D or E). Functions in these packages perform specific tasks related to the assessment. To build the Kepler workflow, actors (reusable components) were created by invoking functions from the packages and interconnecting them to pass parameters (data) for the flow of assessment execution. The workflow was designed to execute the criteria either individually or simultaneously in one sequential run because we used Synchronous Dataflow Director in Kepler, which orders execution of the workflow.

4.4 Components of the Mountain Ash forest assessment workflow

The workflow begins with Criterion D (the disruption of the principal biotic variable) because this result is required as an input to other sub-criteria, such as 'decline in geographic distribution' (Criterion A) and 'restricted geographic distribution' (Criterion B). Criterion D requires the defined ecosystem (Mountain Ash forest) and principal biotic variable. The principle biotic variable used in the assessment was the presence of hollow-bearing trees, and declines in the principle biotic variable were assessed in the following three time periods: D1, over the past 50 years; D2, over the next 50 years and D3, since 1750.

Criterion A determines the change in geographic distribution of the ecosystem over three time periods as follows: Sub-criterion A1 evaluates a change in distribution of Mountain Ash forest for the current period (past 50 years), A2 evaluates future change (a) next 50 years or (b) any 50 years in the present and future and A3 evaluates historical decline (decline since 1750). In the original assessment, this complete analysis was performed using ArcGIS and the GIS expert performed the analytical procedures. In the Kepler workflow, the complete criteria A assessment was re-written in open-source R programming language and was automated, with the input to the processes as the vector data in a shapefile format of the region.

Criterion B requires the restricted nature of the ecosystem's geographic distribution to be assessed as follows: sub-criterion B1 is the extent (a convex hull around the occurrence points) of a) decline, b) threat and c) small numbers of locations. The sub-criterion B2 assesses the area of occupancy and any a) decline, b) threat or c) small number of locations (using 10 km² grid cells). Finally, the sub-criterion B3 assesses if there is a small number of locations and if these are threatened in the short term (i.e. if a landscape scale fire is a critical threat then the burnt and unburnt locations of the largest known fire represents the number of locations). These locations are discrete areas of the ecosystem that would be affected by the most pervasive plausible threat (i.e., wildfire for this assessment). The assessment was completely re-created using open source R programming language to make it widely available and reusable, whereas Esri ArcGIS software was used in the original assessment.

Criterion C assesses environmental degradation of the most important abiotic variables. These are usually temperature and precipitation and other climate-related variables such as drought, and are assessed over three time periods as follows: sub-criterion C1, previous 50 years; sub-criterion C2, next 50 years and sub-criterion C3, since 1750. Due to data deficiencies, criteria C1 and C3 were not included in the original assessment. Hence, these criteria are not evaluated in the workflow either.

Criterion E is an overarching analysis of the impacts on the principal biotic variables on the probability of ecosystem collapse within the next 50–100 years. Most of the source code used in the workflow was re-purposed from the original assessment but was made more modular. All the simulation parameters used in the workflow are exactly the same as in the original assessment. More details about the assessment and rationale behind using the variables are given in the supplementary materials of Burns et al. (2015).

4.5 Workflow Execution

The Mountain Ash forest IUCN assessment is implemented as a user-driven executable workflow that is highly modularised around each criteria/sub-criteria. The workflow enables users to control the execution by selecting criteria to be executed, and takes around 35 minutes to execute all the criteria sequentially and aggregate the results. Figure 4 shows a complete executable Kepler workflow for the Mountain Ash forest IUCN assessment. Users can interact with the workflow to execute individual, multiple or all criteria of the IUCN assessment. All the results are summarised at the end of the execution and are stored in a pre-defined location that can be changed by users.

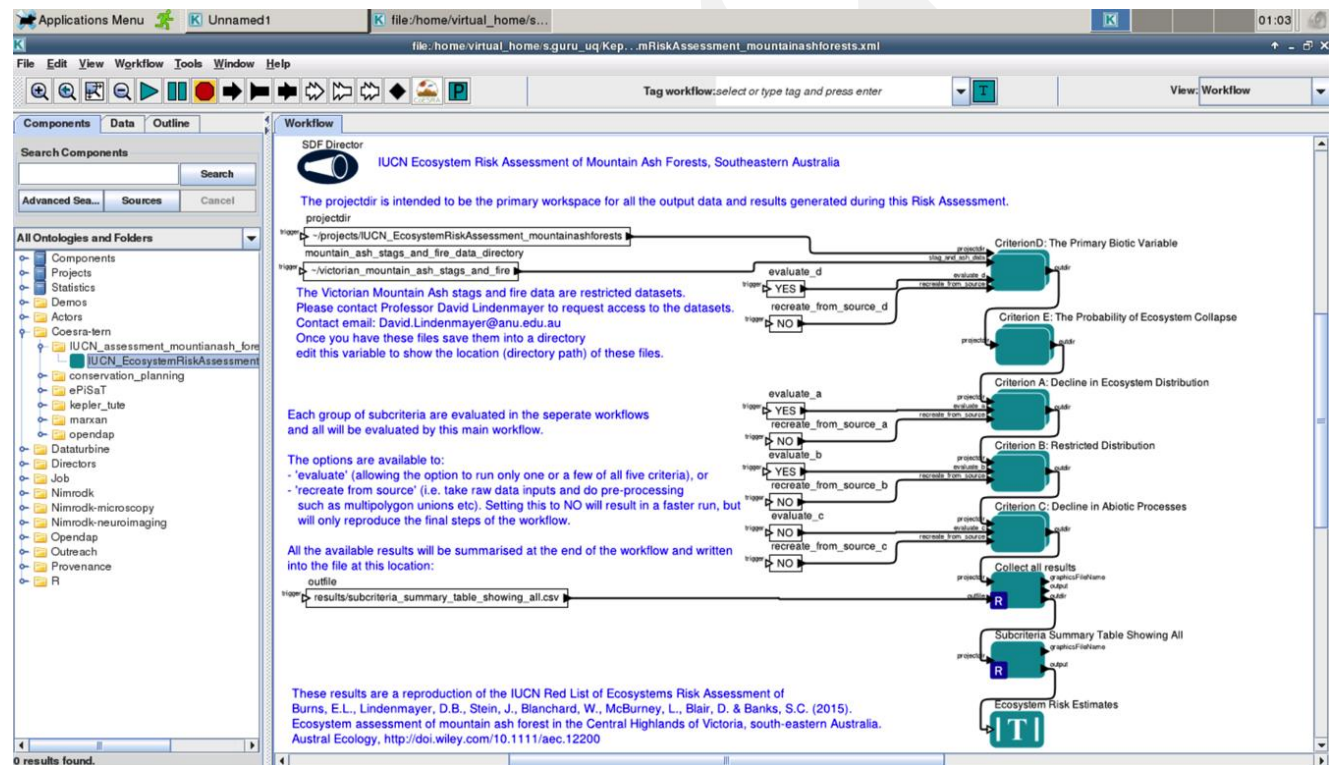


Figure 4: Complete executable Kepler workflow for Mountain Ash forest IUCN Red List of Ecosystems Assessment

4.5.1 Running Workflow from CoESRA virtual desktop

Any registered user of the CoESRA can run the Mountain Ash forest IUCN assessment workflow from the Kepler environment. All the CoESRA sharable workflows are available in a “Coesra-term” folder of the Kepler components menu as shown in Figure 5. The Mountain Ash forest workflow is shared under the folder of “IUCN_assessment_mountainash_forest”. Workflows can be opened in a Kepler workflow viewer and details of the workflow for each of the criteria can also be viewed by opening the composite actors of each criterion. The workflow is pre-configured with all the open access input data files and result output folders and all outputs from the workflow execution are stored in the pre-configured output folder inside the user home folder. However, users can change the output folder based on their requirements.

The Mountain Ash forest risk assessment used seven datasets, and with the exception of one dataset, all were directly mapped to the processes in the workflow. The datasets used are as follows:

1. 'bioclim_biomap': one current climatic envelope and nine future scenarios.
2. 'fire100_year': This is the fire history of the Mountain Ash forest as derived from the datasets published by the Victorian Government. The derived dataset is the same as that used in the original study conducted by Burns et al. (2015)
3. 'lastregen': This is the estimated old growth in 1964 and the present, and was created from various data sources for the original study by Burns et al. (2015).
4. 'mash': This is the Mountain Ash shapefile derived from various sources, including EVC_BCS100, SFRIFred2007, LASTLOG25, FMZ100

and fire100_year.

5. 'nvis_victorian_central_highlands': This is reclassified data from the National Vegetation Information System 3.1 (NVIS 3.1) that was cropped to the central highlands region.
6. 'victorian_mountain_ash_fire_matrix': These data were collated to represent fire-history records and mapped old-growth forests. The primary data source for these derived data was the Victorian Government.
7. 'victorian_mountain_ash_stags': These data are Victorian Tall Eucalypt Forest Plot Network observations of the condition of hollow-bearing trees in the Mountain Ash forest in the Central Highlands of Victoria, South-eastern Australia between 1997 and 2011.

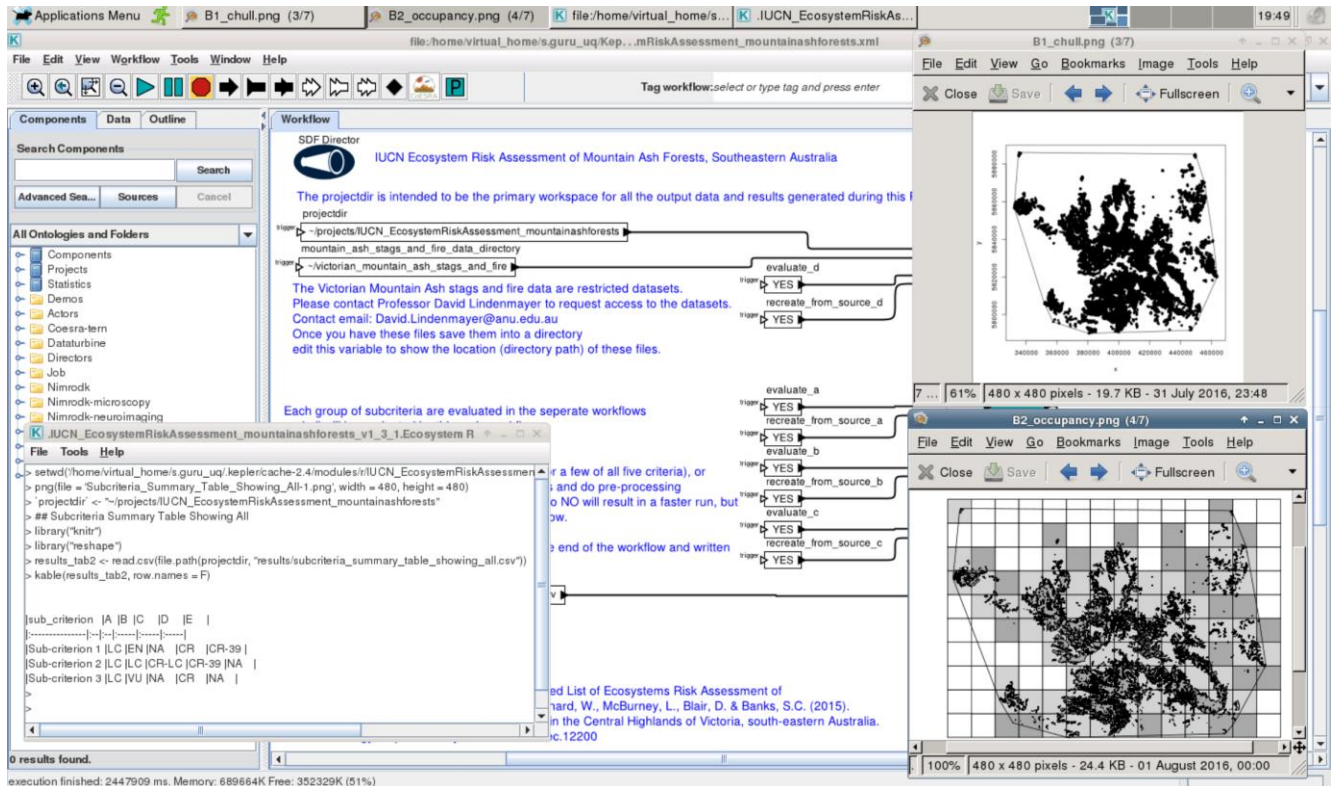


Figure 5: Output of the Mountain Ash forest assessment workflow; Ecosystem collapse criteria B1 and B2 are shown in the right hand side inset map boxes, a summary table of all criteria with available data is shown in the table inset; bottom left.

All the data packages required for execution of the workflow except for the 'victorian_mountain_ash_stags' data are available in a public data share folder in the CoESRA system and are linked to the processes in the workflow. However, the 'victorian_mountain_ash_stags' dataset can be requested through the data portal of the Terrestrial Ecosystem Research Network's Long Term Ecological Research Network Facility (Lindenmayer, 2015). The workflow can be run without restricted data but the results for criterion E are shown as "NA" (Not Available). If users have access to 'victorian_mountain_ash_stags' data from the repository, they should save the file into a folder in the virtual desktop, change the name of the file to 'ash_fire_matrix.csv' and edit the workflow input parameter to the folder path where the data is stored.

Groups of assessment sub-criteria were evaluated sequentially with a separate composite actor, and the main composite actor aggregated the results. The structure of the overall workflow is shown in Figure 5. Because some of the computations are time-consuming, the workflow is developed to run with various options:

1. evaluate: This option allows users to run only one or a few of all five criteria; the workflow will evaluate the criteria if the evaluation is set to "YES", and will skip the evaluation if it is set to "NO";
2. re-create from source (i.e. take raw data inputs and do pre-processing such as multi-polygon unions etc.): Setting this to "NO" results in a faster run, but only some of the steps of the workflow are reproduced.

At the end of the workflow, all available results from the assessment are summarised and written into the file at the location: 'results/subcriteria_summary_table_showing_all.csv' inside a "projectdir" variable parameter of the workflow. The "projectdir" can be configured to any folder of the users choice.

All R Packages and supporting modules used in the workflow are installed into the virtual desktop during the desktop creation phase using the configuration management tool Puppet. However, as a secondary check before every run, the workflow checks for all required third-party packages and install them if they are not available. The provenance of every workflow run is recorded using the Kepler Provenance system. All provenance information is stored in a centralised provenance information database, which is available for users whenever they access the virtual desktop. Provenance enables users to view all of the historical workflows executed by them and can also re-run any previously executed instances of the workflow.

4.5.2 Workflow results

The Kepler workflow reproduced the assessment of the Mountain Ash forest in its entirety as shown in Burns et al. (2015). The workflow also produced all of the intermediate results and reproduced all of the figures from the original published assessment, and all workflow results were matched to those in the literature. Figure 5 shows the sample results of ecosystem collapse criteria B1 and B2 (in the right hand side inset map boxed), with the summary table of all criteria with available data (the table inset bottom left). The workflow is pre-configured to store overall and intermediate results in “/projects/IUCN_EcosystemRiskAssessment_mountainashforests” folder under ‘user home folder’. The results folder also contains all of the calculation results for each of the sub-criteria in separate files with the respective sub-criterion name to make them easily identifiable. Each run of the workflow overwrites the result folder.

The intention of developing the present workflow is not to validate the research methodologies of the assessment but to reproduce the results generated in the original assessment. From the outcome of the workflow execution, we reproduced the Mountain Ash Forest IUCN Red List of Ecosystems Assessment conducted by Burns et al. (2015). The workflow is conserved in its entirety so that any registered CoESRA user can login, access and run the assessment. The workflow will enable other researchers to understand and learn the complete analysis steps taken by Burns et al. (2015) in their assessment.

5 Conclusion

Herein we report the development of the CoESRA system, which is a collaborative platform for conducting and sharing computational research. The system provides integrated infrastructure where data, analysis tools and computational research can be hosted on a single platform. The CoESRA offers a flexible platform that could be used as a key tool for generating reproducible computational science. Although the reproducibility of computational-centric research is evolving, there are benefits to exploring the use of scientific workflows for implementing analysis components, and to investigating the use of virtual environments to share the research. All these benefits are made possible by opportunistic use of cloud computing. The CoESRA system enables users to share infrastructure, data and analyses as a service, drastically reducing the setup cost of the infrastructure for other users who wish to re-run the virtual experiment. Formation and deployment of the CoESRA platform are flexible and scalable, and any new software can be added to the platform system and any number of virtual desktops can be created and managed. Moreover, because storage and computing infrastructure are completely separated, new virtual machines can be created at any time without losing user data or public dataset collections.

From an Australian Researcher’s perspective, the CoESRA is an easy way to access cloud infrastructure without going through the cloud compute and storage merit allocation process. The CoESRA virtual desktop offers a flexible way to access and use cloud infrastructure at short notice and is a useful resource for short-term users of cloud resources.

We have reproduced the Mountain Ash forest IUCN RLE assessment and present the complete computational process as an executable workflow. The published workflow processes were mapped to all openly accessible data, were configured with all of the relevant parameters used in the original assessment, and reproduced all the results from the original assessment. Hence, the workflow provides an opportunity for researchers to re-run assessments with additional data. Taken together, the present analyses demonstrate how the execution environment is key for research reproducibility by sharing a platform to run complicated computational research analyses. We also conducted another computational study and shared the reproducible workflow for conservation planning using Marxan (Guru et al., 2015).

To date, almost 32 IUCN RLE assessments have been conducted globally (IUCN, 2015), but to the best of our knowledge, none have been presented in a reproducible and repeatable manner. The Mountain Ash forest assessment workflow is the first of its kind that can be reviewed and executed by anybody. This will enable proponents of assessments to better engage with all stakeholders to determine both short and long-term plans for conservation of the Mountain Ash forest ecosystem.

It is important to note that in practice, due to funding and time constraints; there is a trade-off between publishing new research and investing extra effort to make a study reproducible. However, the benefits of making research reproducible that we have described here should be taken into account. For example, the improved transparency in methods is a fundamental principle of science. Furthermore, this will help researchers to develop analytical processes that are easy for a third-party to review, understand, test and potentially even modify for use in new analyses.

6 Future work

To improve the CoESRA system, several new tools will be added to the platform to cater for a wide-range of ecosystem science users. Currently, all user accounts and data folders are stored in the cloud storage in the network file system server. The system could be stabilised by migrating the users account folders into an OpenStack volume storage service (cinder) and by storing data in an object storage service (swift). From the operational perspective, this will enable the creation of virtual machines from various nodes of NeCTAR, and will mitigate the risk of failure for the Queensland node of NeCTAR, where the current system is deployed. The Guacamole, which renders the virtual desktop in a web browser, requires fine-tuning to work efficiently with Firefox and Internet Explorer browsers. Moreover, additional functionalities for copying content and uploading files from local to remote desktops will be offered in the future release. To improve the usability of the system, a workflow repository will be created so that users can search and run computational analyses without even going to the virtual desktop. This will enhance the usability and ease of use of the system to execute reproducible computational research.

Finally, we intend to implement more IUCN assessments and make them available for the broader scientific community to review and repeat as needed. Potentially, the reproducible computational research workflow could be made available during the scientific literature review process, enabling reviewers to check the analytical approaches with minimal effort.

Acknowledgements

The Authors thank Kim Olsson, Shilo Banihit and Matthew Mulholland for their contribution to the project. Special thanks to Victorian Government,

CSIRO and The Australian National University for providing access to the primary data for successful implementation and execution of a workflow. Terrestrial Ecosystem Research Network (TERN) and Australian National Data Service (ANDS) are supported through the Australian Government National Collaborative Research Infrastructure Strategy (NCRIS). The QRISCloud, a Queensland Node of NeCTAR and RDS administered by Queensland Cyber Infrastructure Facility (QCIF) provides the compute and storage infrastructure for CoESRA. The Authors would like to thank the anonymous reviewers for their constructive comments and suggestions that greatly contributed to improving the quality of the paper. The Authors would also like to thank Catherine Stubbs and Timothy Chaston for proof reading the manuscript.

Appendices

Appendix S1: a Conceptual workflow of the Mountain Ash forest IUCN Red List of Ecosystems assessment.

REFERENCES

- AAF, 2016. Australian Access Federation: <https://aaf.edu.au/>.
- Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludascher, B., Mock, S., 2004. Kepler: an extensible system for design and execution of scientific workflows, *Scientific and Statistical Database Management*, 2004. Proceedings. 16th International Conference on, pp. 423-424.
- ArcGIS, 2016. esri ArcGIS: <http://www.esri.com/software/arcgis>
- Asendorpf, J.B., Conner, M., De Fruyt, F., De Houwer, J., Denissen, J.J.A., Fiedler, K., Fiedler, S., Funder, D.C., Kliegl, R., Nosek, B.A., Perugini, M., Roberts, B.W., Schmitt, M., van Aken, M.A.G., Weber, H., Wicherts, J.M., 2013. Recommendations for Increasing Replicability in Psychology. *European Journal of Personality* 27, 108-119.
- Ashton, D.H., 1975. The Seasonal Growth of *Eucalyptus regnans*. *Australian Journal of Biology* 23, 239-252.
- Budget, O.o.M.a., 2014. A Framework for Ongoing and Future National Science Foundation Activities to Improve Reproducibility, Replicability, and Robustness in Funded Research.
- Burns, E.L., Lindenmayer, D.B., Stein, J., Blanchard, W., Mccburney, L., Blair, D., Banks, S.C., 2015. Ecosystem assessment of mountain ash forest in the Central Highlands of Victoria, south-eastern Australia. *Austral Ecology* 40, 386-399.
- Cassey, P., Blackburn, T.M., 2006. Reproducibility and Repeatability in Ecology. *BioScience* 56, 958-959.
- Cock, P.J.A., Grüning, B.A., Paszkiewicz, K., Pritchard, L., 2013. Galaxy tools and workflows for sequence analysis with applications in molecular plant pathology. *PeerJ* 1, e167.
- Collins, F.S., Lawrence, A.T., 2014. Policy: NIH plans to enhance reproducibility, *Nature*. Nature Publishing Group, pp. 612-613.
- Donoho, D.L., Maleki, A., Rahman, I.U., Shahram, M., Stodden, V., 2009. Reproducible Research in Computational Harmonic Analysis. *Computing in Science & Engineering* 11, 8-18.
- Editorial, N., 2012. Must try harder. *Nature* 483, 509-509.
- Errington, T.M., Iorns, E., Gunn, W., Tan, F.E., Lomax, J., Nosek, B.A., 2014. An open investigation of the reproducibility of cancer biology research. *eLife* 3, e04333.
- Freedman, L.P., Cockburn, I.M., Simcoe, T.S., 2015. The Economics of Reproducibility in Preclinical Research. *PLoS Biol* 13, e1002165.
- Freire, J., Bonnet, P., Shasha, D., 2012. Computational reproducibility: state-of-the-art, challenges, and database research opportunities ACM SIGMOD International Conference on Management of Data, Scottsdale, Arizona, USA, pp. 593-596.
- Garijo, D., Kinnings, S., Xie, L., Xie, L., Zhang, Y., Bourne, P.E., Gil, Y., 2013. Quantifying Reproducibility in Computational Biology: The Case of the Tuberculosis Drugome. *PLoS ONE* 8, e80278.
- Garnaat, M.a.o., 2009. Boto. Python interface to Amazon Web Services.
- Guru, S.M., Dwyer, R.G., Watts, M.E., Dinh, M.N., Abramson, D., Nguyen, H.A., Campbell, H.A., Franklin, E.C., Clancy, T., Possingham, H.P., 2015. A Reusable Scientific workflow for conservation planning. MODSIM 2015, Gold Coast.
- Hanigan, I.C., Blanchard, W., 2015. IUCN Red List of Ecosystems Risk Assessment, 1.0 ed, <https://bitbucket.org/coesra/iucnecosystemriskassessment>.
- IUCN, 2015. IUCN Red List of Ecosystems Assessment.
- Jasny, B.R., Chin, G., Chong, L., Vignieri, S., 2011. Again, and Again, and Again, and Again *Science* 334, 1225-1225.
- Jun, Z., Gomez-Perez, J.M., Belhajjame, K., Klyne, G., Garcia-Cuesta, E., Garrido, A., Hettne, K., Roos, M., De Roure, D., Goble, C., 2012. Why workflows break - Understanding and combating decay in Taverna workflows, *E-Science (e-Science)*, 2012 IEEE 8th International Conference on, pp. 1-9.
- Keith, D.A., Rodríguez, J.P., Rodríguez-Clark, K.M., Nicholson, E., Aapala, K., Alonso, A., Asmussen, M., Bachman, S., Basset, A., Barrow, E.G., Benson, J.S., Bishop, M.J., Bonifacio, R., Brooks, T.M., Burgman, M.A., Comer, P., Comín, F.A., Essl, F., Faber-Langendoen, D., Fairweather, P.G., Holdaway, R.J., Jennings, M., Kingsford, R.T., Lester, R.E., Nally, R.M., McCarthy, M.A., Moat, J., Oliveira-Miranda, M.A., Pisanu, P., Poulin, B., Regan, T.J., Riecken, U., Spalding, M.D., Zambrano-Martínez, S., 2013. Scientific Foundations for an IUCN Red List of Ecosystems. *PLoS ONE* 8, e62111.
- Keith, H., Mackey, B.G., Lindenmayer, D.B., 2009. Re-evaluation of forest biomass carbon stocks and lessons from the world's most carbon-dense forests. *Proceedings of the National Academy of Sciences* 106, 11635-11640.
- Kenall, A., Edmunds, S., Goodman, L., Bal, L., Flintoft, L., Shanahan, D., Shipley, T., 2015. Better reporting for better research: a checklist for reproducibility. *GigaScience* 4, 32.
- L. Bavoil, S. P. Callahan, P. J. Crossno, J. Freire, C. E. Scheidegger, C. T. Silva, Vo, H.T., 2005.
- VisTrails: enabling interactive multiple-view visualizations, *VIS 05. IEEE Visualization*, 2005, pp. 135-142.
- Leek, J., Peng, R., 2015. Opinion: Reproducible research can still be wrong: Adopting a prevention approach. *Proc Natl Acad Sci USA* 112, 1645-1646.
- Lindenmayer, D.B., Mccburney, L., Blair, D., 2015. Victorian Tall Eucalypt Forest Plot Network: Stag and Fire Severity Observations in Ash Forest in the Central Highlands of Victoria, South-eastern Australia, 1998–2011, TERN Long Term Ecological Research Network.
- Lindenmayer, D.B., Hobbs, R.J., Likens, G.E., Krebs, C.J., Banks, S.C., 2011. Newly discovered landscape traps produce regime shifts in wet forests. *Proceedings of the National Academy of Sciences* 108, 15887-15891.
- Loope, J., 2011. Managing Infrastructure with Puppet. O'Reilly Media, Inc.
- Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E.A., Tao, J., Zhao, Y., 2006. Scientific workflow management and the Kepler system. *Concurrency and Computation: Practice and Experience* 18, 1039-1065.
- Marcus, E., 2014. Credibility and Reproducibility. *Cell* 159, 965-966.

- Mesirov, J.P., 2010. Accessible Reproducible Research. *Science* 327, 415-416.
- Nosek, B.A., Alter, G., Banks, G.C., Borsboom, D., Bowman, S.D., Breckler, S.J., Buck, S., Chambers, C.D., Chin, G., Christensen, G., Contestabile, M., Dafoe, A., Eich, E., Freese, J., Glennerster, R., Goroff, D., Green, D.P., Hesse, B., Humphreys, M., Ishiyama, J., Karlan, D., Kraut, A., Lupia, A., Mabry, P., Madon, T., Malhotra, N., Mayo-Wilson, E., McNutt, M., Miguel, E., Paluck, E.L., Simonsohn, U., Soderberg, C., Spellman, B.A., Turiitto, J., VandenBos, G., Vazire, S., Wagenmakers, E.J., Wilson, R., Yarkoni, T., 2015. Promoting an open research culture. *Science* 348, 1422-1425.
- Oinn, T., Greenwood, M., Addis, M., Alpdemir, M.N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M.R., Senger, M., Stevens, R., Wipat, A., Wroe, C., 2006. Taverna: lessons in creating a workflow environment for the life sciences. *Concurrency and Computation: Practice and Experience* 18, 1067-1100.
- Peng, R.D., 2011. Reproducible Research in Computational Science. *Science* 334, 1226-1227.
- Pritsker, M., 2012. Studies show only 10% of published science articles are reproducible. What is happening?
- Rodríguez, J.P., Keith, D.A., Rodríguez-Clark, K.M., Murray, N.J., Nicholson, E., Regan, T.J., Miller, R.M., Barrow, E.G., Bland, L.M., Boe, K., Brooks, T.M., Oliveira-Miranda, M.A., Spalding, M., Wit, P., 2015. A practical guide to the application of the IUCN Red List of Ecosystems criteria. *Philosophical Transactions of the Royal Society of London B: Biological Sciences* 370.
- Rodriguez, J.P., Rodriguez-Clark, K.M., Baillie, J.E.M., Ash, N., Benson, J., Boucher, T., Burgess, N.D., Collen, B.E.N., Jennings, M., Keith, D.A., Nicholson, E., Revenga, C., Reyers, B., Rouget, M., Smith, T., Spalding, M., Taber, A., Walpole, M., Zager, I., Zamin, T., 2011. Establishing IUCN Red List Criteria for Threatened Ecosystems
- Establecimiento de Criterios para la Lista Roja de UICN de Ecosistemas Amenazados. *Conservation Biology* 25, 21-29.
- Sandve, G.K., Nekrutenko, A., Taylor, J., Hovig, E., 2013. Ten Simple Rules for Reproducible Computational Research. *PLoS Comput Biol* 9, e1003285.
- Santana-Perez, I., Perez-Hernandez, M.S., 2015. Towards Reproducibility in Scientific Workflows: An Infrastructure-Based Approach. *Scientific Programming* 2015.
- Schlicher, B.G., Kulesz, J.J., Abercrombie, R.K., Kruse, K.L., 2015. A Computing Environment to Support Repeatable Scientific Big Data Experimentation of World-wide Scientific Literature, 15th International Conference on Scientometrics and informatics, Istanbul.
- Skaggs, T.H., Young, M.H., Vrugt, J.A., 2015. Reproducible Research in Vadose Zone Sciences. *Vadose Zone Journal* 14.
- Stodden, V., Miguel, S., 2014. Best Practices for Computational Science: Software Infrastructure and Environments for Reproducible and Extensible Research. *Journal of Open Research Software* 2.
- Talia, D., 2013. Workflows Systems for Science: Concepts and Tools. *ISRN Software Engineering* 2013, 15.
- Viggers, J.L., Weaver, H.J., Lindenmayer, D.B., 2013. Melbourne's Water Catchments: Perspectives on a World Class Water Supply. *CSIRO Publishing, Melbourne*.
- Wilson, G., Aruliah, D.A., Brown, C.T., Chue Hong, N., Davis, M., Guy, R., T., Haddock, S.H.D., Huff, K.D., Mitchell, I.M., Plumbley, M.D., Waugh, B., White, E.P., Wilson, P., 2014. Best Practices for Scientific Computing. *PLoS Biol* 12.
- Yamada, K.M., Hall, A., 2015. Reproducibility and cell biology. *The Journal of Cell Biology* 209, 191-193.
- Zeilenga, K., 2006. Lightweight directory access protocol (LDAP): Technical specification road map.