Australian Government
Bureau of Meteorology

An adaptable framework for development and real time production of experimental sub-seasonal to seasonal forecast products

Catherine de Burgh-Day, Morwenna Griffiths, Hailin Yan, Griffith Young, Debbie Hudson, Oscar Alves

April 2020

Bureau Research Report - 042

An adaptable framework for development and real time production of experimental sub-seasonal to seasonal forecast products

Catherine de Burgh-Day, Debbie Hudson, Oscar Alves, Morwenna Griffiths, Hailin Yan, Griffith Young

Bureau Research Report No. 042

April 2020

National Library of Australia Cataloguing-in-Publication entry

Authors: Catherine de Burgh-Day, Morwenna Griffiths, Hailin Yan, Griffith Young, Debbie Hudson, Oscar Alves

Title: An adaptable framework for development and real time production of experimental sub-seasonal to seasonal forecast products

ISBN: 978-1-925738-15-5

Series: Bureau Research Report – BRR042

Enquiries should be addressed to:

Catherine de Burgh-Day

Bureau of Meteorology
GPO Box 1289, Melbourne
Victoria 3001, Australia

Catherine.deBurgh-Day@bom.gov.au

Copyright and Disclaimer

The Bureau of Meteorology advise that the information contained in this publication comprises general statements based on scientific research. The reader is advised and needs to be aware that such information may be incomplete or unable to be used in any specific situation. No reliance or actions must therefore be made on that information without seeking prior expert professional, scientific and technical advice. To the extent permitted by law and the Bureau of Meteorology (including each of its employees and consultants) excludes all liability to any person for any consequences, including but not limited to all losses, damages, costs, expenses and any other compensation, arising directly or indirectly from using this publication (in part or in whole) and any information or material contained in it.

Contents

List of Tables

Tables from Appendix A

Tables from Appendix C

Tables from Appendix E

List of Figures

Figures from Appendix D

Figures from Appendix E

EXECUTIVE SUMMARY

The Bureau of Meteorology's multi-week and seasonal prediction system, ACCESS-S1, was made operational in August 2018 (Bureau of Meteorology, 2018). Forecast outputs from this system are delivered to the Climate Services, Water Forecasting Services, and Science to Services within the Bureau, as well as many other stakeholders in Australia and overseas.

A new post-processing pipeline has been developed to add value to the forecasts produced by ACCESS-S1. This pipeline has been designed to be modular, robust and flexible. The early stages of the pipeline have been made operational (Bureau of Meteorology, 2019), and later stages of the pipeline are due to be implemented in operations by 2021. A research version of this pipeline is also used to rapidly prototype and trial new forecast products, both for internal use and with external stakeholders. The pipeline has been designed such that new trial products can be added quickly and with minimal technical expertise and produced in a near-realtime setting from then on. In this capacity the pipeline underpins several major research projects including The North Australia Climate Program, Rural R&D for Profit: Forewarned is Forearmed, Ocean Climate Metrics for New Zealand, and informs the narrative of the operational seasonal climate outlook service.

1. INTRODUCTION

When developing processing pipelines in a research setting, it can be a struggle to maintain a balance between flexibility, stability, and ease of use. This challenge is amplified where these pipelines are running in realtime, are providing outputs to external parties, and where they are processing large volumes of data.

A significant component of the recent work undertaken by the authors of this report is the development, skill assessment, and trialling of new forecast products to support decision making on multi-week to seasonal timescales. A proportion of the products trialled are transitioned to operational systems for either use by paying customers, or general dissemination to the public.

Prior to this, they are often trialled with collaborators and stakeholders in a research setting. At this stage there is less need for continuity of service, but more need for flexibility in making design and scientific changes to the products, as well as adding new trial products relatively quickly.

Here we present a framework for developing a robust, versioned, and flexible post-processing and plotting pipeline for generating experimental forecast products using real time model data as an input.

In this report we first discuss the motivation for, and design constraints of the pipeline, static files and realtime data. We then give a high-level overview of the architecture of the pipeline, before describing each stage of the pipeline in order. Finally, we discuss remaining challenges and future improvements, and provide some concluding remarks.

2. MOTIVATION AND DESIGN REQUIREMENTS

The pipeline was developed for generating products from the Australian Bureau of Meteorology seasonal prediction system, ACCESS-S1 (Hudson et al. 2017). This post-processing and plotting pipeline was required to rapidly process and visualise model outputs for both monitoring model performance in real-time, and for servicing a number of research projects and collaborations.

Constraints on this framework were that it must require relatively little effort to set up and maintain, and each individual product must be modular and resilient to failures in other products. New experimental product “modules” would be added to the pipeline by several researchers and developers, both within the same group, and in other groups within the Bureau. Contributors to the pipeline have widely varying levels of programming expertise and although the use of Python is encouraged, there are instances of other programming languages that have been used. The framework for the pipeline therefore had to be agnostic about both the language used for each module, and to some degree, the quality of the code.

Since products generated by the pipeline were developed through a number of projects, it was also necessary to be able to transition separate components or stages of the pipeline to operations without needing to operationalize the entire system. This further reinforced the requirement that the pipeline be modular.

As mentioned above, it was not a requirement that the research pipeline had perfect continuity of service. However, the goal was to achieve the minimal number of product outages without requiring dedicated support.

An additional challenge for the pipeline framework was that there was minimal infrastructural support available. This meant that different stages of the pipeline had to be able to be spread across multiple systems (e.g. the Bureau's development supercomputer, Bureau research Virtual Machines, the National Computational Infrastructure (NCI) supercomputer, etc.) that met infrastructure needs, since dictating infrastructure requirements on one system was not an option. These systems did not share service accounts, so it was also necessary for components of the pipeline to be able to operate under a number of user or service accounts.

Prior to the upgrade to ACCESS-S1, the Bureau used the same operational model POAMA (Predictive Ocean Atmosphere Model for Australia; Alves et. al, 2003; Wang et al, 2008) for seasonal prediction for more than 10 years. Development of the post-processing and plotting pipeline for POAMA was incremental and occurred over much of this period. Many now-standard products were developed during the lifetime of the system, and the volume and complexity of the data generated by POAMA was much less than that of ACCESS-S1. It is expected that upgrades to ACCESS-S will occur much more frequently, and many existing products will need to be ported to each new model as upgrades occur. Re-implementing the post-processing and plotting pipeline for each model upgrade would be a costly exercise, and so a final constraint on the pipeline developed for ACCESS-S1 was that it was largely model- and system-agnostic. This ensured that it could continue to be used through subsequent model upgrades and system changes with minimal development costs.

There is a need for versioning of large static files associated with the pipeline: Forecast products on multi-week to seasonal timescales need to be bias-corrected or calibrated and are often presented with reference to some past climatological period (in the case of ACCESS-S1, the standard climatological period is 1990 to 2012). In order to do this, a family of climatological files must be generated. These files are based on a set of retrospective forecasts over the period (referred to as a hindcast). Observations for that climatological period are also required for calibration or for referencing in a particular product.

As the set of products being generated increases in number, and as products change scientifically, these climatologies often need to change as well to remain a valid comparison. To ensure reproducibility of products it is necessary to keep each version of the climatologies that has been used in the pipeline. The family of climatologies required is large on disk (~400 Gb) and so traditional versioning techniques (e.g. fully-copied versioned sets of the entire climatological set on disk, or change-tracking version control using Git) are not suitable for versioning them.

3. PIPELINE DATA

The pipeline has two main streams of input data; operational realtime forecast data from ACCESS-S1, and static data. In the following sections the realtime and static inputs to the pipeline are described.

3.1 Realtime raw ACCESS-S1 data

The pipeline ingests realtime data from the operational ACCESS-S1 model, which runs daily. Data from each model run comes in two sets; a 22-member “weekly” ensemble, which has daily time resolution and lead times to 6 weeks (42 days), and an 11-member “seasonal” ensemble, which has daily time resolution and lead times to 7 months (217 days). This gives a total of 33 members for multi-week timescales and 11 members for seasonal timescales. These forecast data are global and are on the model native resolution (~60 km for the atmosphere and ~25 km for the ocean) and are available for a wide range of model variables. For more information about the Bureau's operational seasonal forecast system, ACCESS-S1, see Hudson et al. (2017).

3.2 Static Files

Static datasets are used in all stages of the pipeline, including processing and for adding to visualisations. The static files include both model hindcast data and observational data.

A "standard research set" (which is not specific to the pipeline) of hindcast data serves as the basis of the static data used by the pipeline, although much of the static data the pipeline references has had significant additional processing. The further-processed static data produced for the pipeline changes frequently, since new products added to the pipeline often require new static files to reference.

The variety and complexity of these files is therefore significant, and for many file types there is a dependency chain (for example, climatological mean data must be computed before anomalies can be computed). Furthermore, many of these files are computationally expensive to create, so a full set may not be able to be created at once on a given machine. To enable a larger number of users to easily add to the pipeline static files and to recreate existing files, the scripts used to generate these files have been put into a Cylc framework.

Cylc is a general-purpose workflow engine that was designed for production weather, climate, and environmental forecasting systems (Oliver et al. 2018, Oliver et al. 2019). Cylc is used in conjunction with Rose (https://metomi.github.io/rose/2019.01.2/html/index.html), a toolkit developed at the UK Met Office for writing, editing and running application configurations.

Rose is used to define the combination of variables and file types desired. The start dates for which these files should be computed can also be specified. The Cylc framework will then manage the scheduling of the tasks to create a full set of new static files. This allows new users to (for example) easily generate a given file type for a new variable without needing to modify any scripts.

For details of the ACCESS-S1 hindcast configuration, the processing done as part of the standard research set, and examples of the kinds of pipeline-specific static datasets used by the pipeline, see Appendix A.

As well as model static data, the pipeline references observational data such as AWAP (Australian Water Availability Project; Jones et al. 2009) and Reynolds Optimum Interpolation Sea Surface Temperature V2 (OISST-V2) daily Advanced Very High Resolution Radiometer 0.25° analysis (hereafter ‘Reynolds’; Reynolds and Smith, 1994; Reynolds et al. 2002).

One of the advantages of referencing observational climatologies rather than hindcast climatologies (assuming that the forecasts have already been bias-corrected or calibrated) is that there is no need to use the 23-year hindcast period. Instead, a longer climatological period can be used, reducing noise and improving the representativeness of the climatology. The ability to use a longer period also helps to offset the fact that observational climatologies are a single realisation, in contrast to the model hindcast which is an 11-member ensemble. This means when using observations there is 11 times less data contributing to any analysis compared to if the 11-member model hindcast dataset was used. A tradeoff to using a longer climatological period is that the assumption of a stationary climate is weaker (i.e. the climate change signal may become a consideration).

The processing of the observational datasets is also managed using Cylc and shares many steps with the processing of the hindcast. For details of the processing done for the observational dataset, see Appendix B.

3.3 Versioning of static files and climatology packages

The earlier stages of the pipeline (P1 and P2, see Section 4) change relatively infrequently and require a simpler set of static files compared with the later stages of the pipeline. For each of P1 and P2, the static files are versioned together under one version number, which is incremented when any one of the files is changed.

Both the hindcast and observational pipeline static files and climatologies used in the later stages of the pipeline (P3 and P4, see Section 4) are versioned and stored in packages of methodologically similar files. This is because (1) for these stages of the pipeline the climatologies are complicated and nontrivial to produce, so versioning them helps to isolate any errors, and (2) new regions, stations, and products are added to these stages of the pipeline frequently (on a weekly to monthly basis), and both new climatologies and changes to existing climatologies are often needed to support this. Since multiple products can share a climatology, any changes could impact existing products if the product is not also updated at the same time. Versioning the climatologies helps to control this process.

Each package is versioned separately from the others, and any change to an existing file in a package results in the version number of that package incrementing. When a package version number increments, a new complete version of the package must be created, including any files which are unchanged since the previous version. It is therefore desirable to define the packages such that the files in the package all change together as much as possible, minimising the amount of duplication of files that takes place between package versions. At the same time, the more packages there are, the more complicated it is to administer and maintain the packages.

The number of separate packages that the full set of static files is divided into therefore seeks to find a balance between minimising the amount of duplication between package versions, while also minimising the number of different packages.

A summary of the static data packages at the time of writing is presented in Table 1. Any new climatological files not fitting the descriptions in Table 1 would be put into a new package.

Table 1: A summary of static file packages in use at the time of writing

Package	**Summary**
Ocean raw model spatial	Global spatial (~25 km resolution) ocean data from the ACCESS-S1 Hindcast, with no calibration or bias-correction
Atmosphere raw model spatial	Global spatial (~60 km resolution) atmospheric data from the ACCESS-S1 Hindcast, with no calibration or bias-correction
Atmosphere bias-corrected spatial	Spatial atmospheric data over Australia from the ACCESS-S1 Hindcast, regridded from 60 km to 5 km resolution, and bias-corrected (see Appendix C.2.1) but not calibrated (see Appendix C.2.2)
Atmosphere calibrated spatial	Spatial atmospheric data over Australia from the ACCESS-S1 Hindcast, calibrated using quantile-quantile matching to a 5 km resolution grid
Ocean raw model regions	Ocean data from the ACCESS-S1 Hindcast, with no calibration or bias-correction, areally-averaged over specific sub-regions of the original global domain (e.g. NINO3)
Atmosphere calibrated regions	Atmospheric data from the ACCESS-S1 Hindcast, calibrated using quantile-quantile matching and downscaled from 60 km to 5 km resolution, then areally-averaged over specific sub-regions of Australia (e.g. Queensland)
Atmosphere calibrated stations	Individual grid points drawn from the 5 km calibrated atmospheric data from the ACCESS-S1 Hindcast

Reynolds spatial	Global spatial ocean data from the Reynolds AVHRR-only product (Reynolds and Smith 1994; Reynolds et al. 2002), covering the period 1990-2012, regridded to the ACCESS-S1 ocean grid (~25 km grid)
AWAP spatial	Spatial 5 km resolution atmospheric data from the AWAP product (Jones et al. 2009), covering the period 1990-2012
Reynolds regions	Areally-averaged ocean data from the Reynolds AVHRR-only product (Reynolds and Smith, 1994; Reynolds et al. 2002), covering the period 1990-2012
AWAP regions	Areally-averaged atmospheric data from the AWAP product (Jones et al. 2009), covering the period 1990-2012
AWAP stations	Individual grid points drawn from the AWAP product, covering the period 1990-2012

4. PIPELINE DESIGN AND ARCHITECTURE

An overview diagram of the pipeline architecture is shown in Figure 1. The pipeline is divided into three processing stages (P1, P2, and P3), a visualization stage (P4) and a configurable viewing tool for presenting the trial forecast products to internal users, external users and project stakeholders - Forecast Viewing Tool, or FVT. Throughout the pipeline, standardized file naming conventions are used.

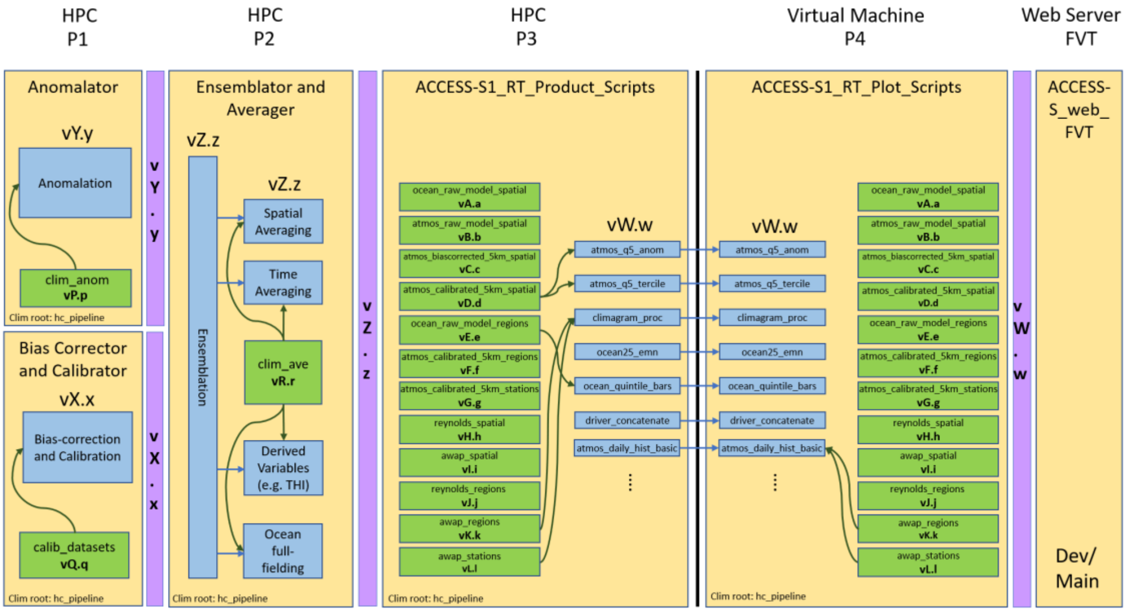


Figure 1: An overview of the architecture of the pipeline, including the five stages of the pipeline, the machines they ran on at the time of writing (i.e. HPC, virtual machines etc.), examples of the main modules and files, and the versioning methodology. Code modules are shown in blue, and static file packages are shown in green. Interfaces between stages of the pipeline where version numbers can be changed independently are shown in purple. Version numbers are indicated with the pattern v?.?, where each independently-versioned component is given a different letter in the diagram to highlight their independence. For example, The

Anomalator code (vY.y), is versioned independently of the Anomalator static files (vP.p). Outputs of the Anomalator suite are written to a directory that includes the version number of the Anomalator code (vY.y). In contrast, the P3 code is versioned in sync with the P4 code (vW.w), and the output files on disk are placed in a directory with the same version number.

The pipeline is deployed across two systems; with the numerical processing of model outputs occurring in an HPC environment and the visualization and display of the processed data occurring on a much smaller Virtual Machine.

Each successive stage of the pipeline changes more frequently than the previous stage. Much of the post-processing of the raw model outputs is confined to the first stage of the pipeline, called P1. P2 includes averaging spatially and temporally, as well as the generation of several derived variables (e.g. daily mean temperatures calculated from maximum and minimum temperature) which are used by multiple downstream products. P3 is dedicated to the creation of the data underpinning forecast products (e.g. probabilities of above median temperature over Australia), and P4 to the visualization of these products. There is a tight coupling between the task in P3 that generates a given product and the task in P4 that visualizes it. All stages of the pipeline are versioned: P1 and P2 are versioned independently, whereas P3 and P4, because they are so tightly coupled, are versioned together.

In the following sections, the purpose and design of each stage of the pipeline is summarized.

4.1 P1: Anomalation, and Bias-correction & Calibration modules

This first step of the pipeline contains low-level post-processing operations which are more computationally expensive. This includes computing anomaly fields, as well as generating higher resolution atmospheric bias-corrected and calibrated fields.

This part of the pipeline is intended to change less frequently, and it is assumed that a smaller, more technical cohort will contribute to its development. It is divided into two independent modules which are run using the Cylc scheduler. The tasks executed by Cylc are implemented in a mixture of Python, Bash and C. The inclusion of C at this stage is by necessity; the calibration step is very computationally intensive and would not be suited to a Python implementation.

P1 ingests raw data from the operational (real-time) forecasts and uses both model hindcast files and observational data for anomaly generation and bias-correction/calibration. Note that since this stage of the pipeline changes much less frequently, these files are kept together under a single version number which increment each time any static file is changed.

As can be seen from Figure 1, the "Anomalation" and the "Bias-correction and Calibration" modules are independent of each other. They do not exchange any data, are versioned independently, and use separate packages of static data (although these static datasets may have some files in common). Any change in either of these (and thus an increment in the version number) will trigger an increment of the version number of all downstream components (i.e. P2, P3 and P4).

A more detailed description of the actions performed by the "Anomalation" and the "Bias-correction and Calibration" modules is provided in Appendix C.

The P1 component of the pipeline is also running in Bureau Operations (Bureau of Meteorology, 2019).

4.2 P2: "Ensemblation", "Averaging" and high-level processing modules

This stage of the pipeline includes "Ensemblation", averaging tasks (dubbed the "Averaging" component), and other high-level processing.

The Ensemblation component constructs 99-member time-lagged ensembles from the outputs of P1 for the multi-week and seasonal products. This step results in a decrease in lead time (to 5 weeks and 6 months for the multi-week and seasonal ensembles respectively) due to the time-lagging and matching to the hindcast for calibration and anomalation.

The next stage of the pipeline (P3) has a requirement that each product generated be independent from any other product (the reason for this is explained in Section 4.3), which means that several products using the same derived variable will duplicate the process of calculating it. To avoid this, high-level processing such as calculation of derived variables which are used in multiple products are calculated once in P2 and then passed to P3. Examples of such variables are full-field bias corrected Sea Surface Temperature (SST) data, and the Temperature Humidity Index (THI; Thom 1959).

Finally, the "Averaging" component performs averaging and extraction steps (including on the derived variables), namely:

1. Time averaging to produce weekly, fortnightly, four-weekly, monthly and seasonal means,
2. Spatial averaging to produce region data and indices, and
3. Extraction of stations (i.e. extraction of single-grid points).

A more detailed description of the processes in P2 is provided in Appendix D.

The ensemblation part of the P2 component of the pipeline is also running in Bureau Operations (Bureau of Meteorology, 2019).

4.3 P3: Modular derived product generation

P3 produces a large number of products from the 99-member multi-week and seasonal ensembles. It ingests data directly from the versioned output directory of P2 and does not directly use any data from P1.

The products generated by P3 are intended for both internal monitoring purposes and as experimental forecast products for trialling with external stakeholders through research projects. Processing and visualization are separated in the pipeline; P3 only produces data files, which are then visualized in P4. P4 in turn does not include any data processing.

P3 consists of a configuration file, a directory of module scripts for computing the data for each product, a runlist indicating which modules to execute, a runner script which executes the modules in the runlist, and a cyclic runner task.

The cyclic runner task is used as a work-around because cron jobs are not permitted on the Bureau’s development supercomputer where P3 is run. It is queued in the supercomputer scheduler PBS, with the instruction to wait until a specific time to execute. At this time it starts the runner script, before re-submitting itself to the PBS queue for the next day.

Work is currently underway to replace this with a Cylc runner framework that runs daily, and uses the runlist to automatically spawn Cylc tasks for each product in the pipeline. Since these tasks are automatically generated, it is expected that there will be little additional work for contributors to the pipeline due to this change.

The configuration file contains high level information which each module in the runlist can draw on, such as:

- Environment setup and variables,
- Version information. Specifically, the version of P2 from which to get data from, the version of P3, and the version of each static file package to use,
- High level bash functions for interacting with the 99-member ensemble,
- A common versioned data input location (i.e. the P2 input data location),
- A common versioned data output location (versioned according to the P3 version),
- The static file package locations (versioned paths)

Each product is independent of the others and can be easily “plugged into” the pipeline, without needing to be integrated into a complex dataflow. This means that no module in P3 can use data generated by any other module in P3, even if this results in the same data being generated multiple times by different modules (as stated earlier, when this does occur, that calculation is often moved to P2). Each module in P3 has a “product name”, and the name of the script that executes the module must contain this product name. The data produced by the module must be written into a subdirectory with the product name, within the common output location. The output file itself must also contain this product name, along with other information which ensures the file is unique, such as the time resolution of the data (daily, weekly, fortnightly, monthly or seasonal), the start date of the forecast, the variable the file contains, the spatial domain the data covers, etc.

If an error occurs in a module in P3, the runner script will continue to execute the remaining scripts in the runlist. In an operational setting this would not be desirable behaviour, however in a research setting this ensures that as many products as possible are created each day, and the responsibility for a given module failing can be left with the developer of that module.

The design of P3 ensures it is scalable and can contain modules with a wide range of quality and robustness without it compromising the pipeline as a whole.

Another advantage of the pipeline being completely modular and agnostic about the quality of the code being executed is that the learning curve to add new modules is shallow, making it possible for researchers with little programming experience to add new experimental products quickly. This enables the rapid transition of experimental work into a routinely-running realtime setting, exposing the results to a wider community within the Bureau of Meteorology, and also to any relevant external parties.

Some modules within P3 rely on the static data packages described in Section 3.3, which are versioned. The version numbers of the static data packages are specified in the P3 configuration file, which is referenced by all modules in P3. This means that all of the modules refer to one common set of static file versions, and incrementing the version of a static file package in the configuration file will affect all of the P3 modules. Enforcing this rule reduces the complexity of the static file dependencies of the pipeline, and hence the workload associated with maintaining it.

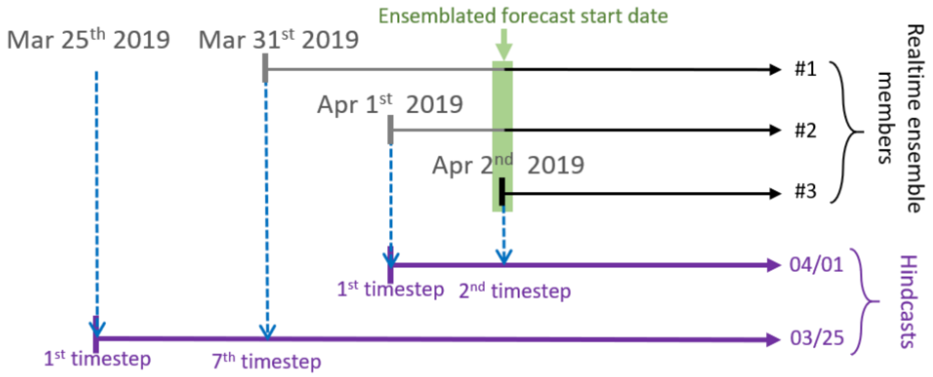


Figure 2: An illustration of the matching methodology used to reference each ensemble member to a hindcast climatology. Each ensemble member is referenced to the hindcast climatology with the closest prior start date. In this example, the ensemblated forecast for April 2nd uses one ensemble member from April 2nd, one from April 1st and one from March 31st. The ensemble members from the 1st and 2nd of April are matched to the 1st and 2nd timestep of the hindcast climatology starting on the 1st of April respectively, while the ensemble member from the 31st of March is matched to the 7th timestep of the hindcast climatology starting on the 25th of March.

Wherever modules in P3 reference hindcast static data packages, the closest prior hindcast climatology start date (dd/mm) is used and the dates are matched with the realtime forecast dates along lead time. The hindcast dates must be matched to the true start dates of each ensemble member (i.e. the date the ensemble member was actually run from, prior to the ensemblation step), meaning different ensemble members in the time-lag ensemble may reference different hindcast start dates. An example of the matching methodology is illustrated in Figure 2.

In 2020 some parts of the P3 component of the pipeline will be transitioned to also be running in Bureau operations, and will include these modules as required to support operational products and services.

4.4 P4: Modular derived product visualisation

P4 generates plots of the data produced by P3. The plots produced by P4 fall into three categories:

1) Plots of climate driver indices and other information of scientific interest and for monitoring purposes. These products are primarily intended for internal use, for example, contributing to the narrative and understanding behind the operational seasonal outlook.

2) Experimental products that have been scientifically validated. These are used by external stakeholders as part of forecast trial studies within a project.

3) Experimental products that involve new methods of visualising information and novel analysis of the forecast ensemble that are un-validated and un-verified. They are highly experimental and are shared with external stakeholders through projects for feedback on the usefulness of the type of information conveyed in order to guide the progression or cessation of development.

In each of these cases, a watermark is added to the products to indicate that they are experimental, non-operational products and are not intended for dissemination outside of current research project groups. Since these are non-operational products, there is no requirement for continuity of service.

P4 shares many design features with P3:

- P4 consists of a configuration file, a directory of module scripts for plotting each product, a runlist indicating which modules to execute, and a runner script which executes the modules in the runlist. The configuration file contains the same types of information as in P3.

- Each product is independent of the others and can be easily “plugged into” the pipeline, without needing to be integrated into a complex dataflow.

- Each module in P4 has a “product name”. The same script and output file naming rules apply to P4 as to P3.

- If an error occurs in a module in P4, the runner script will continue to execute the remaining scripts in the runlist.

As with P3, this means that the learning curve to add products to P4 is relatively shallow. This is especially true if the user has already added the product to P3, since the design of P3 and P4 is very similar.

One point of difference between P3 and P4 is the manner in which they are run each day. P3 uses a cyclic runner script and runs in an HPC environment (Terra), whereas P4 uses cron jobs and runs on a virtual machine.

The P3 and P4 stages of the pipeline (together referred to as P3P4) are tightly coupled. It is a requirement of this part of the pipeline that there be a 1:1 mapping between modules in the two stages. That means that a product in P3 cannot produce files for multiple products in P4, and a product in P4 cannot use files from multiple products in P3. An additional constraint on P3 and P4 is that they must version together, so that a given version of P4 can only ingest data from the same version of P3, where outputs from P3 are written to a versioned location on disk. If any data is missing, P4 will continue to run, with only the modules with missing data failing. It is this resilience that is the motivation behind the 1:1 mapping between P3 and P4; this way one module failing in P3 can only result in one module failing in P4.

The independent, modular nature of each product in P3P4, combined with the fact that there is no need to ensure perfect continuity of service, significantly reduces the workload associated with maintaining the pipeline.

Note that while P3 and P4 are tightly coupled, they do not need to execute on the same machine; P4 includes functionality to routinely fetch data from P3 on a different machine. This means that the more computationally intensive P3 can be run on a more powerful machine, while the smaller P4 can run on a less powerful machine.

As with P3, some modules within P4 rely on the static data packages described in Section 3. The same rules for static files apply to P4 as with P3:

- The packages are kept in versioned directories on disk
- The version of each package used by a given version of P4 is specified in the configuration file. All modules that use a particular package must always use the same version of that package.

4.5 Forecast Viewing Tool: Dissemination of products to External Projects

A set of webpages (called Forecast Viewing Tools; FVTs) are used to provide access to the plots created by P4. It is a requirement of the FVT pages that they are only used to display data from P4.

There are multiple instances of the FVT, each of which displays a subset of the full set of products available from P4. The full set of products are shown on an FVT that is for internal use only. There is an FVT for each relevant research project and each FVT page is password protected, with page-specific passwords for external users, and a common internal master password.

Because each webpage can be configured to show a particular subset of the total outputs of P4, it is possible to finely control which products are shared within each project and each set of stakeholders. Additionally, the simple design on the control files underpinning the FVT pages makes it easy to quickly modify the content of an existing page or generate a new page. This enables both new and established products to be shared with potential project partners and collaborators on an extremely short timeframe, increasing engagement and interest in further work with the Bureau.

5. CHALLENGES AND FUTURE IMPROVEMENTS

There are a few challenges associated with maintaining the pipeline. It is under constant development and new products are added on an ongoing basis, as demanded by existing projects. Maintaining a reliable realtime feed of the large number of products in the pipeline with this rapid rate of development is a challenge; the pipeline produces over 60,000 plots per day and this number is expected to grow with time.

Another challenge arising from this rapid growth is the increasing computational load. Because the pipeline is modular and uses various queuing systems it is scalable, although it can still exceed the capabilities of the systems it is running on. This is most evident for the P1 and P4 components of the pipeline. P1 is the most computationally expensive component. There are 8 model variables to be calibrated by 8 jobs run in parallel, each of which uses 288 processors. Another 8 jobs perform bias-correction for the same variables, each using 144 processors. This is a significant load for the Bureau’s development supercomputer, although fortunately this is the component of the pipeline which is expected to grow the least with time. While P4 is the least computationally expensive component, it is run on a virtual machine with limited CPUs and memory and is expected to grow (together with P3) much faster than P1 or P2. An obvious remedy for this issue is to move P4 onto a more powerful system.

It is important to note that this rapid growth in the volume of data processed by the pipeline, and the number of plots produced, suggests that it has been successful in meeting its intended purpose and is in demand.

Another major challenge in ensuring continuity of service from the pipeline is tracking changes in the large static files it references. While these files are versioned in climatology packages, this does not capture the details of the changes which have taken place between different versions. At the time of writing, these changes are recorded in accompanying documentation which must be kept current. A framework in which changes between climatology package versions were automatically tracked would be preferable. Note however that since the code used to create the static files is under version control, the Git information which ties a climatology package version to the version of the code used to produce it is currently added to the metadata of the static files.

Although the pipeline was designed to be as accommodating as possible of different levels of programming expertise, this remains a challenge. Nonetheless, the pipeline is still a significantly more accessible pathway to realtime production than the available alternatives. Additionally, as the software underpinning the pipeline becomes more mature and robust, the ease with which new products can be added should increase.

Development of the pipeline is ongoing, and work is already underway for the next major upgrade. There are two major avenues for further improvement of the pipeline:

- Transitioning the P3 and/or P4 components of the pipeline to a scheduler such as Cylc, in place of the cron jobs and a bash script runner currently in use. This should simplify the structure of the pipeline and improve mechanisms for triggering between pipeline stages. Work is underway to move P3 to Cylc, but there are no current plans to do so with P4.

- Better versioning and change tracking of the static datasets (as mentioned above). One possible avenue for better change tracking of these files is Git-LFS (https://git-lfs.github.com/). This would enable a more rigorous approach to the version control of the large files associated with the pipeline.

As mentioned previously, the first of these improvements is planned to be implemented within the next 12 months, and it is likely the second improvement will be incorporated either at the same time or immediately afterwards. Several smaller changes can also be implemented to significantly improve robustness and user-friendliness. These include:

- Automated deployment via a workflow manager such as Jenkins or Gitlab CI,

- Automated testing and validation of the codebase using unit and integration tests, and a workflow manager such as Jenkins or Gitlab CI,

- Further standardization of software standards, stylistic conventions and architectures,

- A more unified monitoring framework for the pipeline. Currently a collection of monitoring tools are in place such as web pages summarising plot availability on the FVT, static file availability on disk, etc. However, they vary stylistically, are not in a common location, and in some cases must be run manually.

These improvements will be implemented over time.

6. CONCLUSIONS

Here we have presented an in-depth description of the post-processing pipeline for the Australian Bureau of Meteorology Seasonal Prediction System ACCESS-S1. This pipeline is designed to be modular, scalable, and developer-friendly. We have also described the processing and versioning of the static data used by the pipeline.

Development of this pipeline was extremely rapid, with work commencing in early 2018, and the first version of the pipeline made available to external users by September 2018. An upgrade to a fully-versioned pipeline (including versioning of the static files), with routine automated deployment of the development version was completed in April 2019. The pipeline is now producing over 60,000 plots in the main version and the same number in the development version. As of May 2019, there are two FVT pages providing plots to internal groups, and seven pages providing data to external projects, including one international collaboration.

There is significant interest in obtaining access to the products produced by the pipeline from other external stakeholders and potential project partners, suggesting that the pipeline and FVT are attracting additional collaborators and co-funders to the Bureau. The flexibility, and ability to rapidly develop and add new products to meet multiple project needs, plays a large part in the strong interest in the pipeline.

ACKNOWLEDGEMENTS

We would like to thank all the team members who have been involved with the design and development of the ACCESS-S1 post-processing pipeline and have provided feedback. We would also like to thank all those who have used the pipeline to trial new forecast products. Special thanks to Harley Dallafiore, Andrew Marshall, Paul Gregory, Avijeet Ramchurn and Claire Spillman.

REFERENCES

Alves, O., Wang, G., Zhong, A., Smith, N., Tseitkin, F., Warren, G., Schiller, A., Godfrey, S. and Meyers, G., 2003. POAMA: Bureau of Meteorology operational coupled model seasonal forecast system. Science for drought: Proceedings of the National Drought Forum, Brisbane, Apr 2003, R. Stone and I. Partridge, Eds. Department of Primary Industries, 49-56. Available from DPI Publications, Department of Primary Industries, GPO Box 46, Brisbane, Qld 4001, Australia.

Bureau of Meteorology, 2018. Operational Implementation of ACCESS-S1 Seasonal Prediction System. Bureau National Operations Centre Operations Bulletin Number 120.

Bureau of Meteorology, 2019. Operational Implementation of ACCESS-S1 Forecast Post-Processing. Bureau National Operations Centre Operations Bulletin Number 124.

Dee, D. P. and co-authors, 2011. The ERA-Interim reanalysis: configuration and performance of the data assimilation system. Quart. J. Roy. Meteorol. Soc. 137, 553–597. doi: https://doi.org/10.1002/qj.828

Frost, A. J., Ramchurn, A., and Smith, A., 2018. The Australian Landscape Water Balance model (AWRA-L v6). Technical Description of the Australian Water Resources Assessment Landscape model version 6. Bureau of Meteorology Technical Report.

Hudson, D., Marshall, A. G., Yin, Y., Alves, O., Hendon, H. H., 2013. Improving Intraseasonal Prediction with a New Ensemble Generation Strategy. Mon. Wea. Rev. 141: 4429–4449. doi: https://doi.org/10.1175/MWR-D-13-00059.1

Hudson, D., Alves, O., Hendon, H. H., Lim, E., Liu, G., Luo, J-J., MacLachlan, C., Marshall, A. G., Shi, L., Wang, G., Wedd, R., Young, G., Zhao, M., Zhou, X. 2017. ACCESS-S1: The new Bureau of Meteorology multi-week to seasonal prediction system. Journal of Southern Hemisphere Earth Systems Science. 67: 132-159. doi: 10.22499/3.6703.001

Jones, D.A., Wang, W. and Fawcett, R., 2009. High-quality spatial climate data-sets for Australia. Australian Meteorological and Oceanographic Journal, 58(4), p.233. doi: 10.1.1.222.6311

Nixon-Smith, W.F., 1972. The forecasting of chill risk ratings for new born lambs and off-shears sheep by the use of a cooling factor derived from synoptic data. Commonwealth Bureau of Meteorology.

Oliver, H., Shin, M. and Sanders, O., 2018. Cylc: A workflow engine for cycling systems. Journal of Open Source Software, 3(27), p.737. doi: 10.21105/joss.00737

Oliver, H., Shin, M., Matthews, D., Sanders, O., Bartholomew, S., Clark, A., Fitzpatrick, B., van Haren, R., Hut, R. and Drost, N., 2019. Workflow Automation for Cycling Systems. Computing in Science & Engineering, 21(4), pp.7-21. doi: 10.1109/MCSE.2019.2906593

Reynolds, R. W., Smith, T.M. 1994. Improved global sea surface temperature analyses. Journal of Climate 7: 929–948. doi: https://doi.org/10.1175/1520-0442(1994)007<0929:IGSSTA>2.0.CO;2

Reynolds, R. W., Rayner, N. A., Smith, T. M., Stokes, D. C., Wang, W. 2002. An improved in situ and satellite SST analysis for climate. Journal of Climate 15: 1609–1625. doi: 10.1175/1520-0442(2002)015%3C1609:AIISAS%3E2.0.CO;2

Thom, E.C., 1959. The discomfort index. Weatherwise, 12(2), pp.57-61.

Wang, G. Alves, O., Hudson, D., Hendon, H., Liu, G. and Tseitkin, F., 2008. SST skill assessment from the new POAMA-1.5 system, BMRC Res. Lett. No. 8, 2-6. www.cawcr.gov.au/publications/BMRC_archive/resear chletter/reslett_08.pdf

APPENDIX A: PROCESSED HINDCAST FILES AND HINDCAST CLIMATOLOGIES

A 23-year hindcast set has been created for the ACCESS-S1 system. This is a set of retrospective forecasts, with 4 start dates per month (1st, 9th, 17th and 25th of each month) and each year from 1990 to 2012. For each start date an 11-member ensemble is run and simulates to a lead time of 217 days (about 7 months) with daily resolution. This results in 23x12x4x11 model runs. This dataset is referred to as the model hindcast.

A subset of the hindcast atmospheric variables are bias-corrected and calibrated using almost the same method as that applied to the realtime data and described in Appendix C2. The only difference is that a "leave-one-out cross-validation method" is used. For example, if the 2nd of February 2010 was being calibrated, all years other than 2010 from both the model and the observations would be used in the calibration.

The standard “research” hindcast dataset, which is not expected to change significantly, includes daily, weekly-averaged and monthly-averaged forecasts.

The daily hindcast data are time-averaged along lead time to produce weekly and monthly mean files. The weekly files are averaged from the first lead time in the file. For example, for a model start date of the 9th of January, the first week would be the 9th - 15th of January, and second week would be the 16th – 22nd of January, etc. The monthly files are based on whole calendar months. For example, for a model start date of the 9th of January, the first month in the file will be February (i.e. ignoring the partial month of January), and the second month will be March, etc.

Ensemble means are computed by averaging the 11 ensemble members for each of the model start dates in the hindcast. For each start day and month within the hindcast, a climatology is then created by averaging across the 23 years to give 12x4 mean climatologies. This is done for the daily, weekly and monthly data.

Additional processing is done on the hindcasts to produce the static files used specifically by the pipeline described here. This includes producing:

- Other climatological files, such as region and station index data, weekly data for each lead time (rather than averaged in serial), percentiles of the hindcasts dataset distribution, etc.

- Further-processed versions of the existing files, for example smoothed versions of the daily data to reduce noise arising from the sample size of the distribution.

Details of these additional climatological files created are described in the following sections.

A.1. Regions and stations

As part of the pipeline, a number of products are generated for spatially averaged regions, and stations (single grid cells). Region and station hindcast data are produced to complement these products. This includes creating region and station files for all years, months, start days and ensemble members in the hindcast set, for particular variables.

All of the following steps are applied to all of the spatial, region and station climatologies, and to the raw, bias-corrected and calibrated data, unless stated otherwise.

A.2. Mean daily climatologies including a leap day

Some products in the pipeline use daily data and refer to daily mean climatologies. It is important that these climatologies take leap days into account, and so a new set of mean climatologies are created that include a leap day.

A.3. Smoothed daily mean climatologies

For a given start date in the hindcast, the climatological mean value for each grid box and each lead day is computed from 11x23 values (11 ensemble members and 23 years). This small sample size results in a climatological mean for daily values which is often too noisy. To mitigate this effect, the daily mean climatologies (with a leap day) are smoothed with a top-hat window. The majority of lead times are smoothed with an 11-day window size, however the first 5 and last 5 lead days are smoothed with a variable top-hat window width to account for there being fewer than 11 days of lead times available around the central lead day. The variable window size applied is summarized in Table A1.

Table A 1: A summary of the smoothing window size applied to the daily mean hindcast climatologies

Lead Day	Mean of lead days
1	1 to 3
2	1 to 3
3	1 to 5
4	1 to 7
5	1 to 9
6 to n-5	11-day top hat centered around the lead day
n-4	n-8 to n
n-3	n-6 to n
n-2	n-4 to n
n-1	n-2 to n
n	n-2 to n

A.4. Rolling weekly and fortnightly means (months and seasons are the same as in the research set)

When referencing the realtime ensemble members to the hindcast, the hindcast date chosen is the closest prior hindcast (dd/mm) to the date the ensemble member was run from, rather than the date of the 99-member super-ensemble. Each ensemble member is then individually matched to its respective hindcast data. For example. for an ensemble member run from March the 3rd, the hindcast data from March the 1st will be used and then the dates are matched, i.e. the first two lead times will be removed so that the first time step in the hindcast is for the 3rd of March. This is true for daily data, weekly data and fortnightly data.

Because of the way the 99-member ensemble is created in P2 (See Section 4.2), the weekly (fortnightly) mean hindcast data used in the pipeline must contain a week (fortnight) starting on each lead day. In other words, the time-averaging done to produce the weeks (fortnights) must be a rolling time-average, rather than a serial one. For example, for a hindcast start date of the 17th of March, the first timestep in the file would be the average of the 17th – 23rd of March, the second timestep in the file would be the average of the 18th – 24th of March, etc.

Since both the monthly (seasonal) realtime data and hindcast data have been clipped to the next whole calendar month, rolling average months are not necessary to ensure that matching of dates can be done.

A.5. Weekly, fortnightly, monthly and seasonal percentiles

A common climatological quantity used in the pipeline is percentile values of the hindcast distribution (i.e. quantiles of the full probability distribution function of the hindcast values for a given day, week, fortnight, month or season of the year).

A range of daily, weekly, fortnightly, monthly and seasonal percentiles are computed for:

- The raw, bias-corrected, and calibrated spatial data
- The raw ocean region and station data
- The calibrated atmosphere region and station data

Additional percentiles may be added for future versions of the pipeline.

A.6. Other climatologies

Additional climatological files are added to the above types as the need arises with the addition of new products to the pipeline. Some additional climatology types added, but not described here in detail are:

- Daily counts within a week, fortnight, month or season: The days in a given period are binned according to a set of defined thresholds for each model variable. The result is a histogram of the number of days within a given range of a variable for that period. This is used to compare to the same histogram computed for the realtime forecast ensemble, and is only done for region and station data.

- Consecutive days exceeding a threshold within a week, fortnight, month or season: For a given period, the number of consecutive days N exceeding a threshold T are calculated as a function of N and T, populating a two-dimensional parameter space. This is used to compare to the same calculation applied to the realtime ensemble, and is only done for the region and station data.

APPENDIX B: OBSERVATIONAL DATA

As well as the model static files described in detail in Appendix A, the pipeline references observational datasets.

At the time of writing, these consisted of AWAP (temperature and precipitation only; Australian Water Availability Project; Jones et al. 2009), and Reynolds Optimum Interpolation Sea Surface Temperature V2 (OISST-V2) daily Advanced Very High Resolution Radiometer 0.25° analysis (hereafter 'Reynolds'; Reynolds and Smith, 1994; Reynolds et al. 2002).

Note that since AWAP data from the period 1990-2012 are used to calibrate the realtime atmosphere data in the pipeline, the realtime atmosphere ensemble distribution can be directly compared to the AWAP climatological distribution.

Processing of the observational datasets shares many steps with the processing of the hindcast, including creation of regions and stations, smoothing of daily climatologies, and calculation of time averages and percentile climatologies. Two key differences are:

1. Observation climatological files contain an entire year and can be considered to have periodic boundary conditions (that is, December 31st at the end of the file is followed by January 1st at the start of the file). This means that the smoothing window used can be consistent across the entire file and does not require the ramp-up and ramp-down used for the hindcast.

2. In order for the historical observational data to be matched to the hindcast data over the whole period 1990 to 2012 for all lead times, some 2013 observational data is also required (since, for example, the hindcast run starting on the 25th of December 2012 will have timesteps extending into 2013).

APPENDIX C: ANOMALATION, BIAS-CORRECTION AND CALIBRATION

C.1. Anomalation

The anomalation suite performs a first-order linear correction for model mean bias and drift. Anomalies are created by subtracting a lead-time dependent ensemble mean hindcast climatological mean from the real-time forecast. The climatology (and bias-correction) is a function of variable, start date, lead time and grid box. The creation of anomalies is done on the daily time resolution realtime data and is applied to each ensemble member independently.

The climatological files used for this anomalation step are "smoothed daily mean climatologies" (described in Section A.3). For a given forecast date, the closest prior hindcast climatology start date (dd/mm) is identified and the dates are matched with the forecast dates to create the anomalies. For example, if the real-time forecast starts on 3rd Jan 2019, then the closest hindcast start date is the 1st Jan. The 1st Jan hindcast climatology is used to bias correct, but the first two days in the climatology file are ignored (i.e., the dates in the files are matched).

C.2. Calibration

This module is used to produce calibrated forecasts for Australia on a 5 grid. The resulting data are "full field" and not anomalies. Within the module, two post-processing methods are independently applied: a) simple bias correction and b) quantile-quantile matching. The variables that are produced by this module are shown in Table C1.

Table C 1: Variables valid on the 5 km grid over Australia (produced from both the simple bias-correction and the quantile-quantile matching methods)

Rainfall
Maximum temperature
Minimum temperature
Wind speed
Incoming solar radiation
9 am vapour pressure
3 pm vapour pressure
Evaporation

For both methods, the raw full-field (not anomalies) forecast (and hindcast) data on the native atmospheric grid (N216: ~60 km) are first interpolated to the 5 km AWAP grid using bilinear interpolation. Only model grid-boxes that are 100% land are used in the interpolation and data are extrapolated over the part-land and ocean grid-boxes prior to interpolation.

C.2.1 Bias correction method

The bias correction module first performs the same correction as described in the anomalation section (C.1), but on the 5 km grid over Australia. This module then goes a step further than the anomalation module, to create bias-corrected full-field values. To do this, the 5 km daily time-step forecast anomalies are added to the observed daily climatology, which has been smoothed with an 11-day window. This bias correction dataset is a first-order linear correction for model bias and drift - only systematic biases in the mean of the forecasts have been corrected. The production of this dataset allows comparison with the more sophisticated quantile-quantile matching method and for application and testing of new calibration techniques other than quantile matching.

The variables and respective observed datasets used in this module are the same as those used for the quantile-quantile matching and are described in the following section.

C.2.2 Quantile-quantile matching method

The quantile-quantile (Q-Q) matching technique is described in detail in Bureau of Meteorology (2019). Here we briefly describe the technique. In contrast to the bias correction, described in C.2.1, the Q-Q matching corrects the whole probability distribution function (PDF). If the climatological PDF of the variable in question resembles the observed climatological PDF but is simply shifted in the mean, then a simple bias correction is sufficient. However, often the shape of the model's PDF is different from observed and a more sophisticated calibration is advantageous, particularly when daily data from the forecasts are used to drive application models and getting the correct daily frequency and intensity of, for example, rainfall is important.

The Q-Q matching approach forces the model distribution to be the same as the distribution of the observational data, to produce daily calibrated forecasts over Australia. Where available, the observations used are the 5 km resolution AWAP data (Jones et al. 2009). AWRA-L (Frost et al. 2018) is used as the observations for evaporation. Wind speed is not available from this data set, so daily ERA-Interim (Dee et al. 2011) data are used. These data are interpolated onto the 5 km AWAP grid using bilinear interpolation. Vapour pressure (vprp) is not saved in ACCESS-S1; however, it can be calculated from surface pressure and surface specific humidity, as follows:

$$vprp = -\frac{huss \times R_v \times P_s}{huss \times (R_d - R_v\ ) - R_d}$$

where $huss$ = daily mean specific humidity, P_s = daily mean surface pressure, R_v = 461.7 J kg$_{-1}$ K$_{-1}$ (gas constant for water vapour) and R_d = 287 J kg$_{-1}$ K$_{-1}$ (gas constant for dry air). This effectively gives a value for the daily vapour pressure and is calibrated to the 9am and 3pm AWAP vapour pressure respectively.

The steps required to calibrate a real-time forecast are as follows (each variable, start-date, lead time and location are treated independently):

1) Interpolate the 60 km real-time and hindcast data (1990-2012) for the relevant start date to the 5 km grid using bilinear interpolation;

2) Extract relevant observations for the same date and period (1990-2012) on the same grid;

3) Rank the observations and hindcast data respectively;

4) Calculate the quantile of the real-time forecast datum within the ranked hindcast data (from the step above);

5) Translate the above quantile to the observation quantile to obtain the calibrated value.

6) For example: assume that a 33 °C real-time forecast from the model is at the 98th percentile of the 23 years of hindcast data and assume that the 98th percentile of the observation data is 36 °C. Then 33 °C will be calibrated to 36 °C.

7) If the uncalibrated real-time forecast is above the maximum value in the ranked hindcasts, then the calibrated value is set to: uncalibrated value * (observed max / model hindcast max)

As with the anomalation and bias correction, the Q-Q matching is applied to the daily-timestep data. The closest prior hindcast start date (dd/mm) to the real-time forecast start date is identified and the dates are matched with the forecast dates to perform the calibration. However, for the Q-Q matching, the hindcast data from all 11 ensemble members are used instead of the climatology. The calibration also makes use of the 11-day window (described in detail in Appendix A.3) i.e., the surrounding 5 days (not averaged) are used in the calibration of a particular day (for the hindcasts and observations). This means that for step 3 above, for a given lead-time day and 5 km grid box, the number of observations being ranked is 253 (23 years * 11 days from the window) and the number of hindcast values being ranked is 2783 (23 years * 11 days from the window * 11 ensemble members).

APPENDIX D: ENSEMBLATION, AND PROCESSES IN THE AVERAGING COMPONENT OF P2

The P2 stage of the pipeline includes two components:

1) **Ensemblator:** the construction of weekly and seasonal 99-member time-lagged ensembles.

   For seasonal timescale forecast products, the 99-member ensemble is created from the most recent 9 days of 11-member forecasts (if there are missing forecasts, the products will utilise forecasts going back to a maximum of 15 days). This is hereafter referred to as the seasonal ensemble.

   For multi-week timescale forecast products, the 99-member ensemble is created from the most recent three days of 33-member forecasts (if there are missing forecasts, the products will utilise forecasts going back to a maximum of 5 days). This is hereafter referred to as the multi-week ensemble.

   The forecast start date of the ensemble is given as the date of the most recent ensemble members. When creating forecast products for the forecast start date, the days prior to the "start date" in those ensemble members with an earlier start are ignored. The ensemblator strips off these days and so to a user it appears that all 99-members were initialised on the same day. This makes creating forecast products and manipulating the data much easier for both internal and external users. This ensemblation process is illustrated in the schematic below (Figure D1).

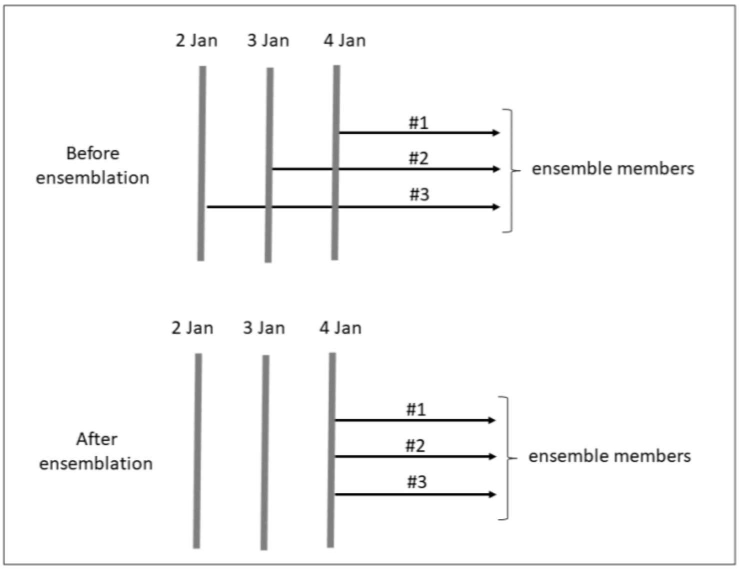


Figure D 1: Schematic illustrating the effect of ensemblation for a hypothetical three-member daily-lagged ensemble. For the creation of the lagged ensemble valid for the 4th January, the ensemblation will remove the 3rd January from Member #2 (which was initialised 1-day prior to #1) and will remove the 2nd and 3rd January from Member #3 (which was initialised 2-days prior). All three members now have daily forecast data that starts on 4th January.

Ensemblation is done for all the variables and data types (e.g., global atmospheric; global SST; 5 km-calibrated; 5 km bias-corrected) that are passed to the ensemblator suite from P1.

2) **Averaging:** this step includes several processes:

a) Calculation of bias-corrected full-field Sea Surface Temperature (SST) data. The SST anomalies are used to compute a bias-corrected full-field SST for all timescales by adding an observed SST climatology from 1990-2012 which has been smoothed with an 11-day top-hat window. This climatology is versioned so that any changes to the processing methodology can be implemented in a trackable manner, and previous versions of the full field SST can be recovered.

b) Calculation of derived variables. For example, Temperature Humidity Index (THI; Thom 1959; a measure of heat stress in cattle), Sheep Chill Index (Nixon-Smith 1972), T_{mean} (the mean of the maximum and minimum temperature on the same valid date), and T_{air} (the mean of the maximum temperature on day n and minimum temperature day n-1);

c) Time-averaging of all daily 99-member ensembles to weekly, fortnightly, monthly and seasonal averages. The multi-week ensemble is averaged to produce weekly means in serial (i.e. days 1-7, 8-14, etc.), and rolling fortnights (i.e. days 1-14, 8-21, etc.), as well as a 4-week period (i.e. days 1-28). The seasonal ensemble is averaged to produce means to 6 months for full calendar months. For example, a forecast starting on the 18th of November would have December as the first month in the file. Finally, a rolling seasonal mean is created by averaging these monthly means (e.g. DJF, JFM, FMA, and MAM);

d) Calculation of the ensemble means from the 99-member ensembles;

e) Spatial averaging of the daily, weekly, fortnightly, monthly and seasonal data to regional indices, and extraction of stations from the spatial data (for the daily weekly, fortnightly, monthly and seasonal data).

APPENDIX E: FORECAST VIEWING TOOL (FVT) DESIGN

The FVT webpages (e.g. Figure E1) are automatically generated using a set of common Python, JavaScript, and HTML scripts, together with the page-specific lists of products to display. Each webpage displays data according to 9 dimensions shown in Table E1:

- *Class*, *Category*, and *Product*: These are used for categorisation and grouping in a menu on the left-hand side of the webpage. They are set in the products list for the page and can only have one value per product.

- *Variable*, *Domain*, *Forecast Period*, and *Option*: These are also defined in the products list, however in this case several possible values for each can be defined. These values are then presented in selectable drop-down menus on the FVT.

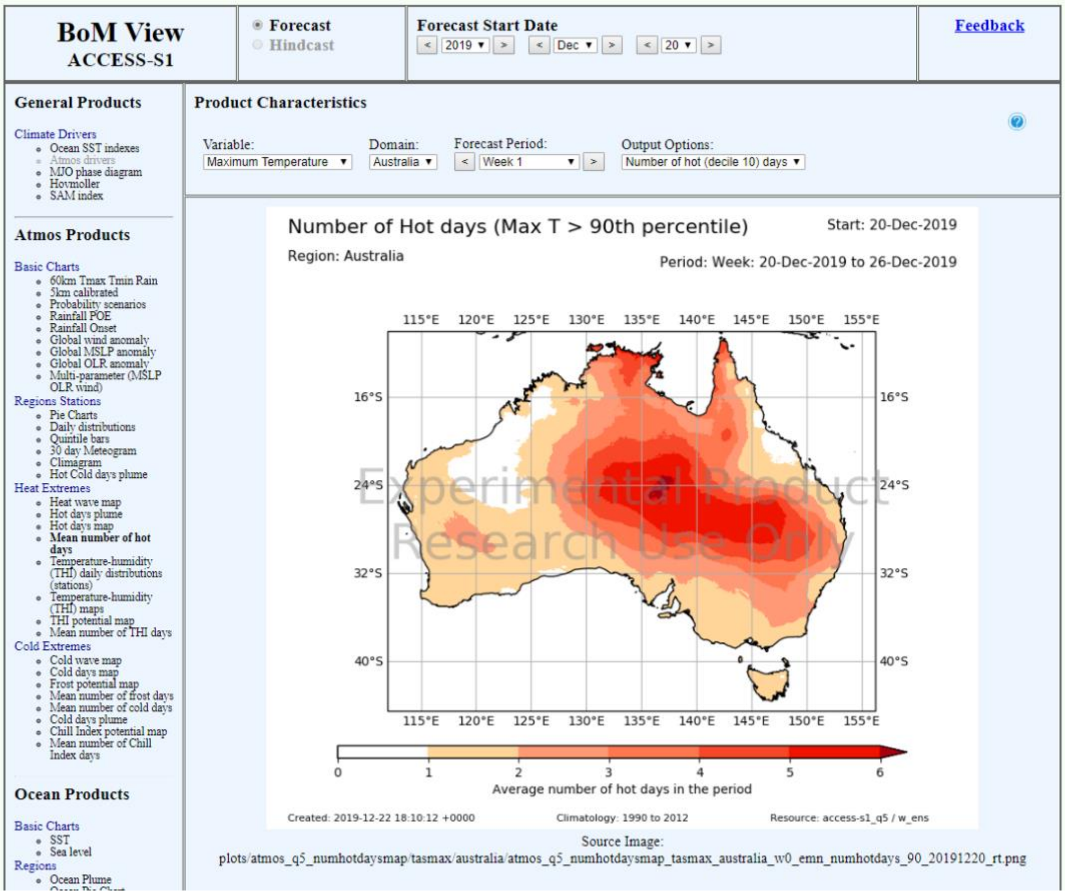
- *Forecast Start Date*, and *Cast*: These are set by the FVT and are not manually configurable. *Forecast Start Date* is selectable and covers every day in a range from some fixed time in the past to a date defined in a file produced by P4. The FVT pages read this file and update the newest Forecast Start Date accordingly. *Cast* describes whether it is hindcast or realtime data being displayed. It is set in the FVT and does not change.

Table E 1: A summary of the dimensions by which products are defined on the FVT

Dimension	**Purpose**
Class	Broad groupings of the categories on the webpage, for example “Atmos Products”.
Category	The category of the product, for example “Basic Charts”.
Product	The name of the product being displayed.
Variable	The variable being displayed, for example maximum temperature, rainfall, etc.
Domain	The spatial domain being displayed. This could be a map of global data, a map of a region, regionally-averaged data, or station data.

Forecast Period	The time-averaging period of the data being displayed. For example for a plot of monthly average temperature data, this would be “monthly’.
Option	“Miscellaneous”: Can contain a wide variety of options, such as ‘anomaly’, probability of top decile’, ‘ensemble mean’, etc.
Forecast Start Date	The date of the data being displayed. The most recent date available is read from a file created by P4 at the end of its most recent run.
Cast	There are only two options; ‘rt’ and ‘hc’. Denotes whether the plot being viewed is of realtime data or hindcast data.

The values defined by the 9 dimensions described above map to segments of the paths and filenames of the files produced by P4. FVT therefore assumes a strict naming structure for the files it displays, and outputs of P4 must conform to the path and filename rules dictated by the FVT dimensions.



BoM View
ACCESS-S1

● Forecast
○ Hindcast

Forecast Start Date: 2019 ▾ Dec ▾ 20 ▾

Product Characteristics

Variable: Maximum Temperature ▾
Domain: Australia ▾
Forecast Period: Week 1 ▾
Output Options: Number of hot (decile 10) days ▾

Number of Hot days (Max T > 90th percentile)
Start: 20-Dec-2019
Region: Australia
Period: Week: 20-Dec-2019 to 26-Dec-2019

Created: 2019-12-22 18:10:12 +0000
Climatology: 1990 to 2012
Resource: access-s1_q5 / w_ens

Source Image:
plots/atmos_q5_numhotdaysmap/tasmax/australia/atmos_q5_numhotdaysmap_tasmax_australia_w0_emn_numhotdays_90_20191220_rt.png

Figure E 1: An example of an FVT page displaying a product. For a given Class (“Atmos Products”), Category (“Heat Extremes”), and Product (“Mean number of hot days”), the Variable (“Maximum Temperature”), Domain ("Australia”), Forecast Period (“Week 1”), and Option (“Number of hot (decile 10) days”) can be selected from drop down menus. The Forecast Start Date is set to the 20th of December 2019, and the Cast is set as “Forecast”.

Features displayed on each page are determined by a group of control files (one per product and one per page) which can contain as few as four lines of information to specify:

- For the webpage control files, what products to display on the webpage;
- For the product control files, the product name, forecast periods, variables, and output options.

Creating a new version of the FVT involves creating a new version of the webpage control file and running a build script. Similarly, adding new products to the pages involves creating a new product control file, adding the product to the desired webpage control files, and running the build script.