IPCC Standard Output from the CSIRO Mk3.0 Climate System Model

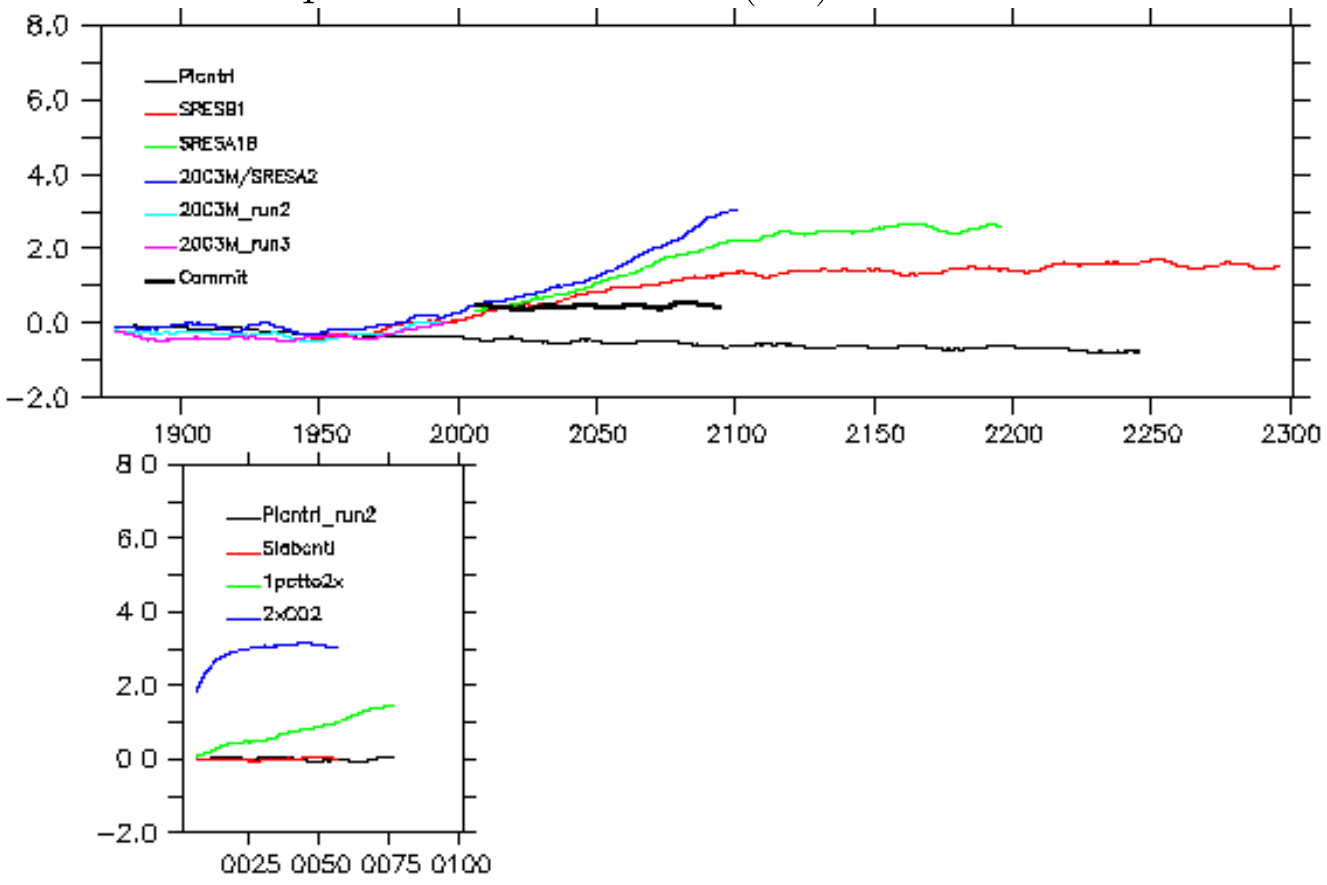
Mark A. Collier, Martin R. Dix, Anthony C. Hirst, Harvey L. Davies, Tracey I. Elliott, Hal B. Gordon, Polly J.L. Morgan†, Siobhan P. O'Farrell, Leon D. Rotstayn and Ian G. Watterson

CSIRO Marine and Atmospheric Research Paper 008

May 2008

†CSIRO High Performance Scientific Computing, Melbourne, Victoria

Globally averaged surface monthly air temperature anomalies (°C)



IPCC Standard Output from the CSIRO Mk3.0 Climate System Model

Mark A. Collier, Martin R. Dix, Anthony C. Hirst, Harvey L. Davies, Tracey I. Elliott, Hal B. Gordon, Polly J.L. Morgan†, Siobhan P. O'Farrell, Leon D. Rotstayn and Ian G. Watterson

CSIRO Marine and Atmospheric Research Paper 008

May 2008

†CSIRO High Performance Scientific Computing, Melbourne, Victoria

National Library of Australia Cataloguing-in-Publication entry

Author: Collier, Mark A.

Title: IPCC standard output from the CSIRO mk3.0 climate system model [electronic resource] / Mark A. Collier.

ISBN: 9780643068971 (pdf.)

Series: CSIRO Marine and Atmospheric Research paper ; 008.

Notes: Includes index.

Subjects: Climatic changes–Detection.
Atmospheric circulation–Computer simulation.
Ocean circulation–Computer simulation.
Ocean-atmosphere interaction–Computer simulation.
Climatology.

Other Authors/Contributors:
CSIRO. Marine and Atmospheric Research.

Dewey Number: 551.524

Enquiries should be addressed to:

Mark A. Collier
CSIRO Marine and Atmospheric Research
Private Bag No. 1, Aspendale, Victoria 3195, Australia
Telephone: +61 3 9239 4422
Facsimile: +61 3 9239 4444
Email: mark.collier@csiro.au

Distribution list

Project Manager	
On-line approval to publish	
Client	
Authors	
Other CSIRO Staff	
National Library	
State Library	
CMAR Library as pdf (Meredith Hepburn)	
CMAR Web Manager as pdf (Diana Reale)	

Important Notice

Use of this Report

The use of this Report is subject to the terms on which it was prepared by CSIRO. In particular, the Report may only be used for the following purposes.

- this Report may be copied for distribution within the Client's organisation;
- the information in this Report may be used by the entity for which it was prepared ("the Client"), or by the Client's contractors and agents, for the Client's internal business operations (but not licensing to third parties);
- extracts of the Report distributed for these purposes must clearly note that the extract is part of a larger Report prepared by CSIRO for the Client.

The Report must not be used as a means of endorsement without the prior written consent of CSIRO.

The name, trade mark or logo of CSIRO must not be used without the prior written consent of CSIRO.

Contents

List of Figures

List of Tables

1 Abstract

This document details the monthly (1620 years) and daily (420 years) CSIRO Mk3.0 Climate System Model output that has been recently submitted to the collecting body of the Intergovernmental Panel on Climate Change (IPCC) for Working Group 1. Twelve experiments (three control and nine climate change) have been performed and satisfy the IPPC Standard Output data requirements of the Program for Climate Model Diagnosis and Intercomparison (PCMDI) archive. This report describes the raw CSIRO Mk3.0 output that have been used in forming the standard output parameters, and for which of the standard IPCC parameters this were not possible. This new set of data is expected to be conveniently available to research applications in the Australian and international community for many years to come. A new standard of involvement by the CSIRO climate modelling community has been realised.

Several key Climate Indices, namely the Nino3.4, Southern Oscillation and the North Atlantic Oscillation, have been derived for all Mk3.0 model experiments using the IPCC standard output parameters, and are described in this report. IPCC standard parameters for the fluxes of heat, moisture and momentum for the 20th century ensemble average are presented and compared to the equivalent ERA40 reanalysis parameters. Proper representation of these air-sea coupling parameters are vital in getting the atmosphere and ocean climate accurate, particularly when there is no flux adjustment as in the case of the Mk3.0.

A complete list of software written in Tcl-Nap is included in the appendix for researchers using the Mk3.0 model framework who endeavour to make their output conform with this internationally accepted standard. Examples of executing Tcl-Nap functions to produce desired results are included. Locations of where the data is locally stored are provided for those with access to the CSIRO mass data store system.

2 Background

The World Meteorological Organisation and the United Nations Environmental Program established the Intergovernmental Panel on Climate Change (IPCC) to examine and report on climate change science. The Program for Climate Model Diagnosis and Intercomparison (PCMDI) have been archiving Climate System Model (CSM) output to aid the Working Group 1 (WG1) component of the recent IPCC Fourth Assessment Report (AR4). Using the archived parameters, WG1 of the IPCC examine, in particular, atmosphere, land surface, ocean and sea ice components of the climate system.

It was a contractual requirement of the Climate Modelling Group at CSIRO Atmospheric Research to participate in the IPCC AR4 data request, with a number of specific experiments conducted beyond our key contractual requirements joined up with those that matched the IPCC activity, to meet the mandatory experimental requirements of being a provider of data to the IPCC activity.

Nevertheless the benefits of being successful participant in IPCC are two-fold. Firstly CSIRO earns credibility as a climate modelling group of international standing. Secondly, it aids our own model development as we can objectively see how the CSIRO climate model compares with others around the world, by directly comparing results. Model developers can then focus their attention on areas where the climate model performs inadequately which is extremely useful as our computing and human resources are limited.

Groups contributing to IPCC must meet stringent data standards, in particular the netCDF Climate and Forecast (CF) Metadata Conventions. It has been recommended that groups intending to participate use the CMOR[1] package to enforce conformity of processed raw model output, however, we have utilised the Nap extension of the Tcl scripting language, reffered to as Tcl-Nap hereafter *Davies* (2007). This is because, firstly, we have significant expertise and opportunity to extend this already powerful scripting language. Tcl-Nap can perform several functions of which the CMOR package is not able to do (i.e. we have a complete processing environment to perform all of the necessary processing on the raw model files). Finally, we have guaranteed conformity by applying CMOR to a broad selection of our own processed files to ensure the files conform with the way CMOR interprets the implementation of the CF standard.

Important links are available from the man PCMDI web page, `http://www-pcmdi.llnl.gov`, which have been extensively consulted in the preparation of the processing software described in this report, and the preparation of the report itself. The most imporant are the ones pointing to the precise definition of output parameters and a list of model simulation specifications. Another vital link includes detailed documentation on all the participating models, as well as essential information that researchers should access in performing accurate and meaningful analysis. Further details on downloading the full range of model results from the IPCC Data Portal, `https://esg.llnl.gov:8443/index.jsp`, or the CSIRO Mk3.0 model results directly of the CSIRO dataserver are in Section 5.

[1] `http://www-pcmdi.llnl.gov/software/cmor/cmor_readme.php`

3 CSIRO Mk3.0 Climate System Model

The CSIRO Mk3.0 Climate System Model (hereafter CSIRO Mk3.0), has been documented in *Gordon et al.* (2002). Additional details pertaining to the CMIP[2] experiments (which in the document are described by the experiment names PIcntrl and 1%to2×, see Table 1) are provided in *Collier et al.* (2004). A brief summary will be provided here.

The Mk3.0 model is based on the Mk2 model as described in *Gordon and O'Farrell* (1997), *O'Farrell* (1998) and *Hirst et al.* (2000), with major numerical, computational and paramaeterisation improvements. Importantly, the Mk3.0 model is run without any flux adjustments and suffers from only a moderate amount of climate drift. Readers are again pointed to the model documenation summary, which includes pertinent Mk3.0 model specifications at `http://www-pcmdi.llnl.gov/ipcc/model_documentation/CSIRO-Mk3.0.pdf`.

The horizontal resolution of the Mk3.0 atmospheric model is spectral T63 (approximately 1.875° latitude × 1.875° longitude) with 18 vertical levels (hybrid sigma-pressure vertical coordinate). The atmospheric model has been upgraded to include a comprehensive cloud microphysical parameterisation (*Rotstayn et al.* (2000)), and the convection parameterisation is based on that used in the Hadley Centre model (*Gregory et al.* (1990)). This convection parameterisation has been linked to the cloud microphysics scheme via the detrainment of liquid and frozen water at the cloud top. Atmospheric moisture advection (vapor, liquid, and frozen) is carried out by the semi-Lagrangian method (*McGregor* (1993)). A simple treatment of the direct radiative effect of sulfate, which entails a perturbation of the surface albedo (Mitchell et al. 1995), is included in the model. The land surface scheme (six layers of moisture and temperature) with a vegetation canopy (*Kowalczyk et al.* (1991, 1994)), includes a three-layer snow model. Multiple soil (9) and vegetation (12) types are included. The model incorporates a dynamical-thermodynamic polar ice model that includes a variable fraction of leads (*O'Farrell* (1998)).

The Mk3.0 ocean model is based upon the Modular Ocean Model version 2.2 (MOM2.2) of the Geophysical Fluid Dynamics Laboratory (GFDL) model. The oceanic component has horizontal resolution matching that of the atmospheric model's grid in the east-west direction, and twice that in the north-south direction. This represents a grid spacing of approximately 0.9375° latitude × 1.875° longitude (approximately; latitude is on a Gaussian grid). For every atmospheric grid point there are two ocean points in the meridional direction, which allows for the atmospheric model and ocean model subcomponents to have matching land-sea masks. There are 31 levels in the vertical, with the spacing of the levels gradually increasing with depth, from 10 m at the surface to 400 m at depth. The ocean model includes a parameterisation of mixing of tracers based on the formulation of *Griffies et al.* (1998)) and *Griffies* (1998), and improved vertical mixing in the tropical Pacific (*Wilson* (2000)).

4 CSIRO Mk3.0 Climate System Model Experiments

The simulations performed for the IPCC AR4 using the Mk3.0 model are listed in Table 1 together with the locally used Mk3.0 run identifier and the ensemble member identifier. Further details of the simulation are presented in Tables 2- 4.

As model solutions are machine dependant, Table 2 describes which NEC supercomputer architecture each of the CSIRO Mk3.0 experiments were performed at the CSIRO High Performance Scientific Computing facilities. This information is important to numerical modellers who wish to re-run portions of completed experiments, to replace missing or bad data, or who may use the information to clarify a numerical solution.

4.1 Model forcings

`Here Martin will write a section on model forcings including a CO2 Emsssion summary plot.`

[2]Coupled Model Intercomparison Experiment `http://www-pcmdi.llnl.gov/projects/cmip`

IPCC Experiment	Mk3.0 Run Identifier	Description	Ensemble
PIcntrl	uk1	pre-industrial control experiment	run1
PIcntrl	ct3	pre-industrial control experiment	run2
20C3M	m20	climate of the 20^{th} century experiment	run1
Commit	s20	committed climate change experiment	run1
SRESA2	m20	SRES A2 experiment	run1
SRESA1B	a1b	720 ppm stabilisation experiment	run1
SRESB1	st1	500 ppm stabilisation experiment	run1
1%to2x	cm3	1% per year CO_2 increase experiment (to doubling)	run1
Slabcntl	qv3	slab ocean control experiment	run1
2×CO2	qv4	2×CO_2 equilibrium experiment	run1
20C3M	n20	climate of the 20^{th} century experiment	run2
20C3M	o20	climate of the 20^{th} century experiment	run3

Table 1: IPCC and Mk3.0 experiment names.

Mk3.0 Experiment	Machine	Model Year	Directory Location of Experiment	
Name	Architecture	Overlap	Year Range[3]	
a1b	SX6	none	~hir020/a1b	
cm3	SX5	none	~col414/cec006/CMIP	
ct3	SX5	none	~col414/cec006/CMIP	
m20	SX5	none	~col414/cec006/M20	
n20	SX6	none	~hir020/n20	
o20	SX6	none	~hir020/o20	
s20	SX6	none	~col414/cec006/S20	
st1	SX5/6	386-393	~col414/cec006/STAB1 $\leq$ 385	~col414/STAB1_CHECK $>$ 385
uk1	SX5	none	~gor079/C_Mk3a.f0.dir $\leq$ 260	~col414/cec006/Mk3.0 $>$ 260
qv3	SX6	none	~dix043/qv3	
qv4	SX6	none	~hir020/qv4	

Table 2: Machine architecture and location on cherax of the raw model output.

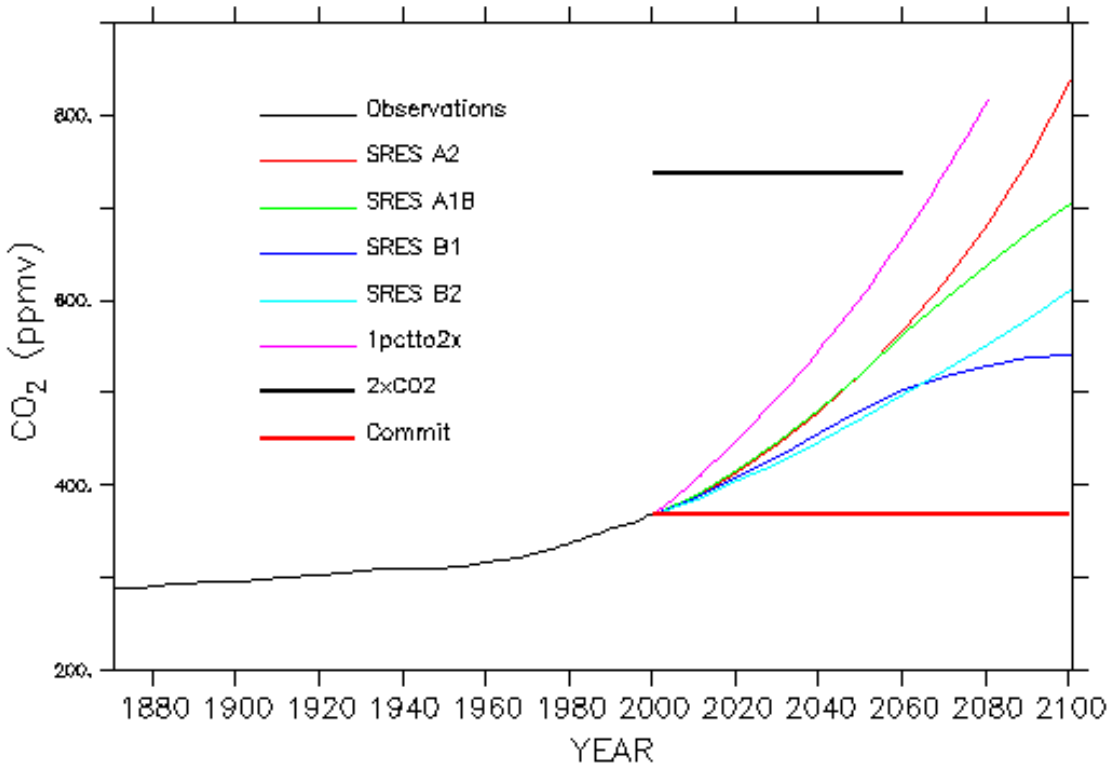


Figure 1. Observed (1871-2000) CO_2 emissions (ppmv) compared with equivalent CO_2 for the SRES, Commitment, 1% compounding to doubling and doubled CO_2 experiments.

To illustrate the performance of the Mk3.0 model to simulate past climates, Figure 1 compares the twentieth century results for three experiments (ensembles) against the the historical surface temperature data set HadCRUT described in *Brohan et al.* (2006). The observational set is provided as an annual time series of area averaged anomalies relative to 1961-1990. For the model results we have adjusted for the control experiment drift by subtracting away from each ensemble member the corresponding part of its series, after smoothing the control experiment time-series with a 30-year running mean so as to avoid an spurious noise. In addition the model monthly anomalies were adjusted so that an average of the 1961-1990 period were again zero, and smoothed with a 12-month running mean to produce an annual time series to compare with the observations. The model members captures the trend and variability of the observations considerably well, including the trend of the 10 year SRESA2 experiment result beyond the year 2000.

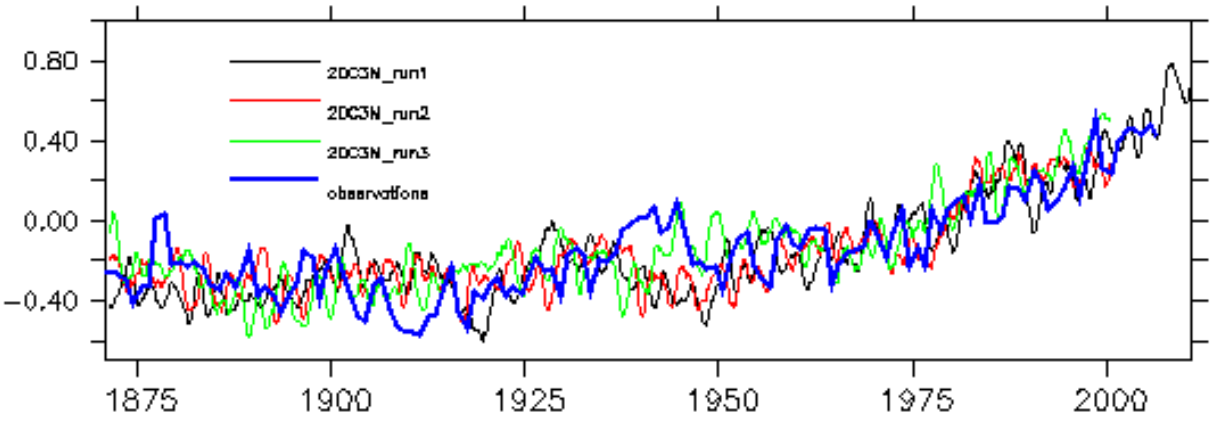


Figure 1: Globally averaged surface annual air temperature anomalies for an ensemble of three 20^{th} century experiments against the observations, relative to the basline climate of years 1961-1990 (°C). See text for details.

To illustrate the sensitivity of the Mk3.0 model to CO_2 forcing when compared to other models, Figure 2 compares the Mk3.0 globally averaged annual surface air temperature anomalies, relative to a climatology of the first 10 years of each experiment. Intrinsically Mk3.0 is near the bottom end of most most models in the range of climate sensitivity to CO_2 forcing. The fairly realistic temperature trend (0.6°C over the 20th Century) shown in Figure 1 in the Mk3.0 is achieved despite the low sensitivity because the indirect aeorosol forcing is ignored. Note that there is an apparent second group of models (BCC-CM1, GISS-ER and FGOALS-g1.0) which occupy a region of the graph with a much lower sensitivity that the majority (only 5 ensembles out of a total of 32) of models. Further judgement and analysis of this behaviour is beyond the scope of this report.

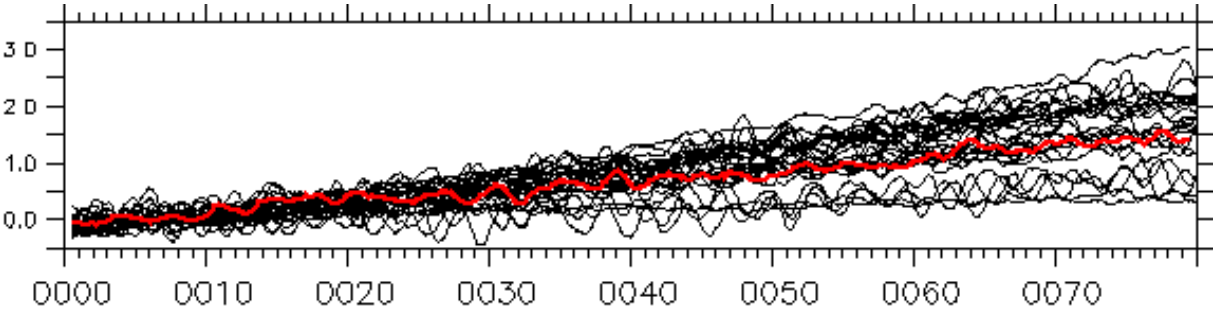


Figure 2: Globally averaged surface annual air temperature anomalies minus climatology formed over first 10 years of each modelling groups (24 in total) IPCC AR4 experiment. Each series has been smoothed with a 12-month running mean to give annual anomalies and shown are all individual ensemble (32 in total), (°C). The CSIRO Mk3.0 ensemble is shown in thick red. See text for details.

The bulk of the raw Mk3.0 and procesed data (described by this report) is stored on equipment (high speed disk and a tape "silo") attached to the CSIRO owned and operated machine cherax. A machine of this name has been in existence for more than a decade, although the machine type and operating systems have changed throughout its existence. Throughout the production of Mk3.0 IPCC data, cherax was a SGI Altix 3000 NUMA ia64, and as of November 2005 it had 128 1.3Ghz IA-64 processors and 244 Gbytes of memory. The Altix is tightly coupled with the the CSIRO data store and provides significant processing capacity for working with the data store holdings. Further details on these facilities Further details of the faciliies are available at the web

page http://hpsc.csiro.au/.

As a way of describing the relative performance of the CSIRO Mk3.0 model against other models in the IPCC AR4 archive, a Taylor diagram has been generated which compares basic statistics (normalised standard deviation and pattern correlation) formed from monthly model results and observational values for surface air temperature, sea level pressure and precipitation over the period 1948-2000. For a complete description of this method please refer to the paper of *Taylor* (2001). In this case we have used the NCEP reanalysis (Kislter et al., 2001) as our observational set and the first ensemble for each of the IPCC AR4 models. As the sea level pressure calculation is problematic over regions of high topography, we have restricted the calculation to terrain of less than 1000 metres, this accounts for 93 percent of the globe and importantly eliminates spurious sea level pressure values with result over most of the Antarctic continent and Greenland, as well as more significant land areas found in the sub-continent, Americas and South Africa.

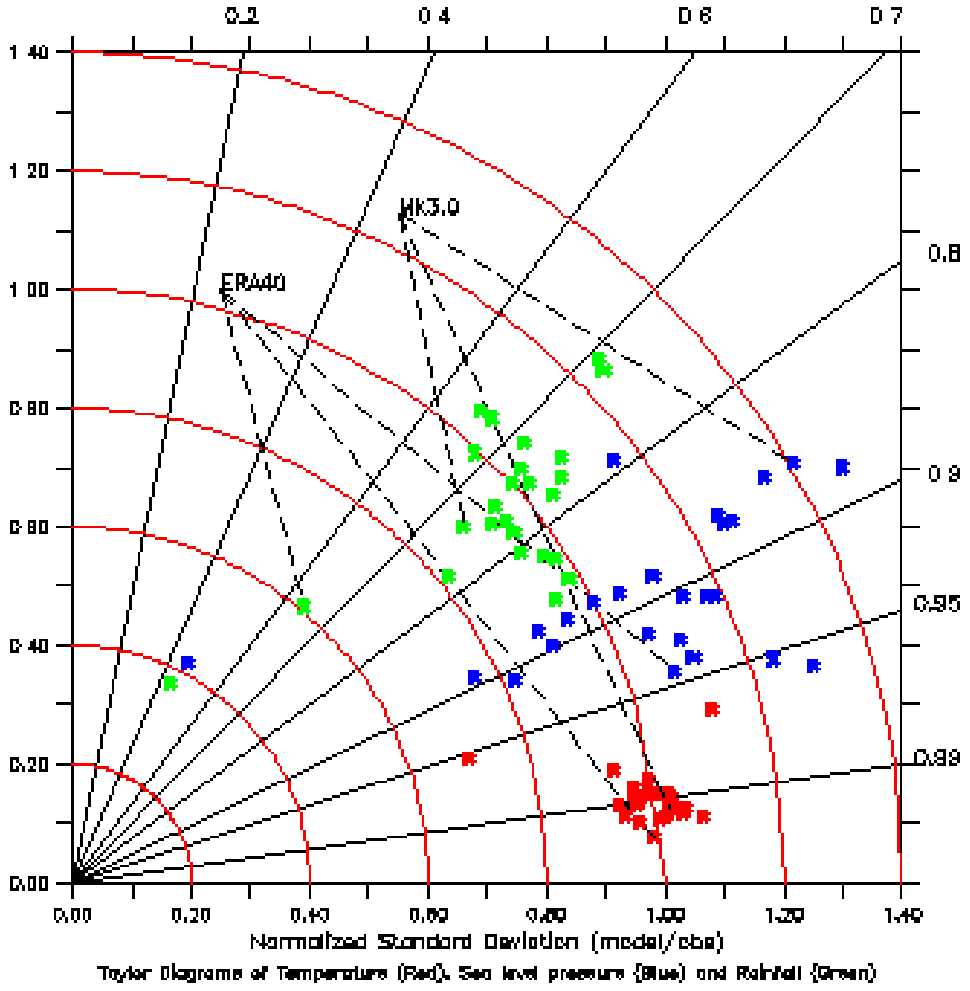


Figure 3: Taylor diagram of model surface air temperature, sea level pressure and precipitation compared against NCEP reanalysis for the years 1948-2000. The first ensemble of all IPCC AR4 models are included in this diagram, and the location of the CSIRO Mk3.0 model is indicated. See text for details.

Mk3.0 IPCC data are located in the following directory structure:

cherax:~IPCC/data/{*model_version*}/{*experiment_name*}/{*table_name*}/{*run_name*}

and netCDF file name structure:

{*parameter_name*}_{*table_name*}_{*year_range*}_{*experiment_name*}_{*model_version*}_{*run_name*}.nc

where

{*parameter_name*} is derived from the Tables 5- 18.

{*table_name*} has the value A1a, A1b, A1c, A2a, A2b, A4, O1a, O1b, O1c or O1d.

{*year_range*} is a single year (e.g. 2100) in the case of 3D daily parameters, a range of years (e.g. 1991-2000) for other time dependant parameters or no value for time invariant parameters (note for the latter the underscore is not necessary and removed, e.g. orog_A1b_20C3M_mk3.0_run1.nc).

{*experiment_name*} is derived from the first column in Table 1.

{*model_version*} is mk3.0[4]

{*run_name*} has the value run1, run2 or run3. From Tables 3 and 4 it can be seen that two control experiments and three 20th century experiments are available. No 1%to4x or AMIP experiments have been performed with the Mk3.0.

A summary of the directory structure holding Mk3.0 IPCC netCDF data is provided in Figure 4. This is also a useful summary of where Tcl-Nap processing scripts and their input data are located. IPCC Experiment names can be referenced from Table 1.

Earlier versions of the *Slabcntl* and *2×CO2* experiments were performed and given the Mk3.0 run identifier of *qv1* and *qv2* respectively. The later experiments of *qv3* and *qv4* are considered superior as they have been integrated for equal periods of time (and a greater number of decades than in *qv1* and *qv2*) and have additional IPCC parameters available for analysis.

It was decided not to allocate resources to the generation of Mk3.0 3-hourly 2D data. The reasons are 1) time constraints 2) the data volume it represents is small when compared to the monthly and daily data totals 3) it would have required a significant effort, and not all of the parameters would have been easily generated because of model constraints 4) very few of the research projects called for 3-hourly data 5) it would have required rerunning of years that had already been completed in some cases.

For most of the requested IPCC experiments, only single ensembles have been performed, and therefore most experiments have data only under sub-directory run1.

Additional "IPCC-like" datasets are available directory through the HPSC mass storage system connected to cherax, amongst other machines, as the CSIRO submission is only a subset of what is available from the raw model output. Not all raw model output has been converted to match the IPCC data standard, and requests for additional data to be made available this way will need to be considered on a case by case basis. The shear volume of Mk3.0 raw data means, both monthly and daily parameters, that not all can be made available to follow the IPCC AR4 data conventions. However, upon request efforts will be made to perform additional processing where practical.

In the following sections, both IPCC and Mk3.0 experiment names will be referenced. Often the Mk3.0 experiment identifiers are of more use to much of the local researchers as they unambiguously describe the experiment and its location on the disk storage.

[4] A future report will be written on mk3.5 experiments and data output.

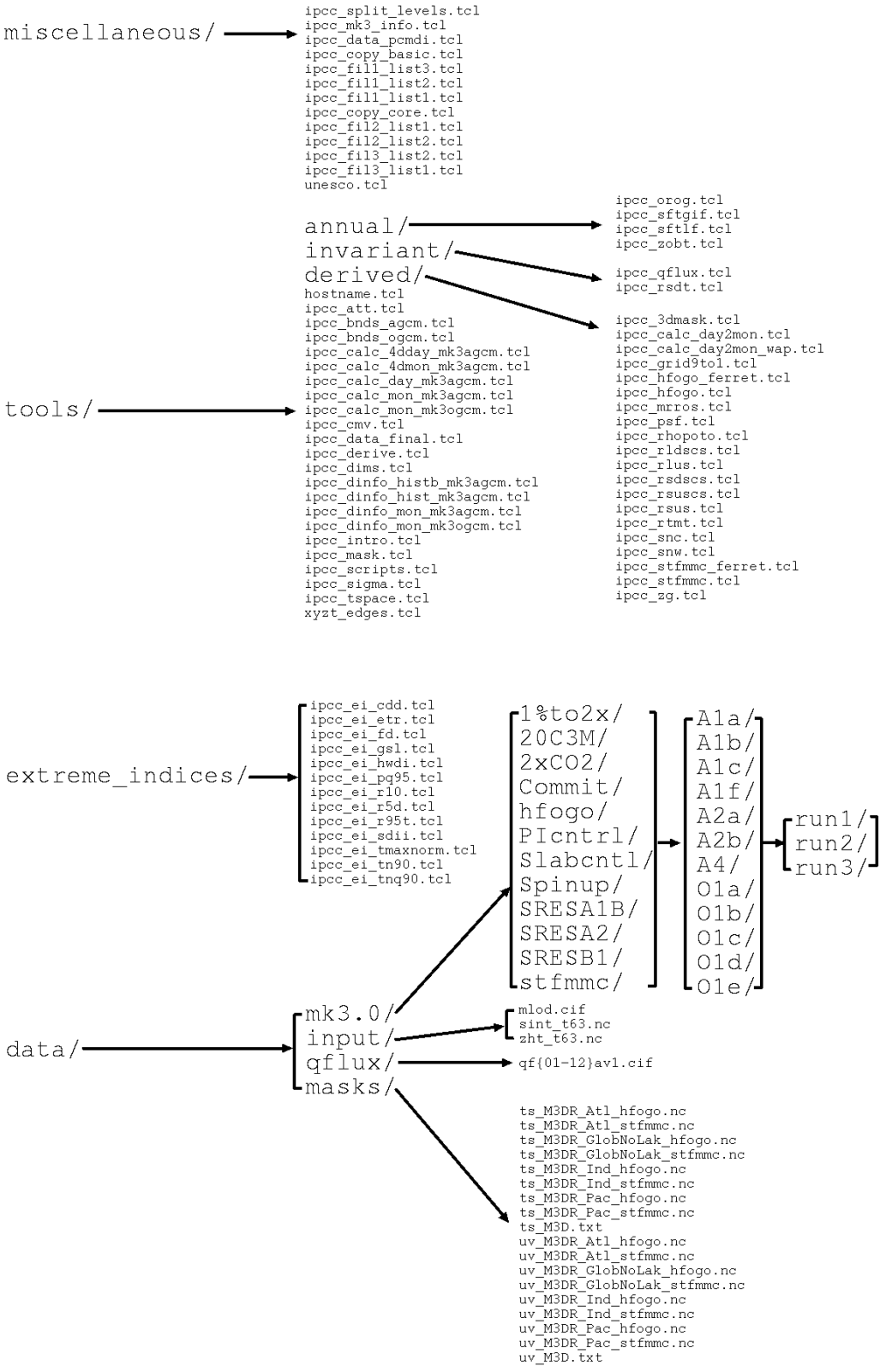


Figure 4: Relevant directories and files in forming IPCC Standard Output from the CSIRO Mk3.0 Climate System Model.

5 Obtaining CSIRO Mk3.0 IPCC NetCDF Data

5.1 Directly From the CSIRO HPSC

A description of the location of the Mk3.0 IPCC data has been described in Section 4 of this report. For CSIRO users this is perhaps the easiest way of accessing the data for CSIRO users and some partners and much analysis and data visualisation can be performed locally on the HPSC machines.

5.2 OPeNDAP

A OPeNDAP[5] server is being hosted by the HPSC and can be found at the web location http://hpsc.csiro.au/cgi-bin/OPeNDAP/CMAR_mk3/nph-dods/ where a username and password must be entered. These credentials are obtained by sending an e-mail to Martin.Dix@csiro.au with a paragraph specifying the project details and the scope of the usage of the requested data. Once logged on the Mk3.0 experimental data is found under the hyper link "ipccdata" following a directory system analogous to the file system structure described in Section 4.

5.3 PCMDI Data Portal

CSIRO Users may obtain access to the Mk3.0 IPCC data through the PCMDI web page https://esg.llnl.gov:8443/index.jsp once user registration is complete. The web form allows specific data files to be downloaded. To obtain registered parameters for specific climate models, experiments and ensembles must be chosen, and the request submitted through the PCMDI data portal. Later an e-mail is sent to the requestee once the data is made available through their batch processing system. Files can then be downloaded from the list of data file addresses and names in the e-mail.

6 Valid and Processed Year Ranges

Tables 3 and 4 list the IPCC recommended experiment name, valid model years (although a subset is usually only provided to PCMDI), and the corresponding model and calendar years for monthly and daily averaged parameters. The data archived by PCMDI from each participating coupled ocean-atmosphere model is a subset of that model's output. There are a number of parameters for specific model periods which are not archived at PCMDI. Users are advised to contact CSIRO for enquiries about such files, and in particular, with any requests for data or additional information. Note that the underlying ("noleap[6]") Gregorian calendar year that has been arbitrarily allocated to the control experiments uk1 (PIcntrl) and ct3 (PIcntrl) in order to align them with the control experiment Gregorian calendar with the m20 (20c3M 1^{st} ensemble) experiment. Note that experiments which have been initialised from other parts of the uk1 control experiment may have a conflict between their model and corresponding calendars.

[5] Open-source Project for a Network Data Access Protocal, see more about this scientific data networking framekwork at http://www.opendap.org/index.html

[6] All Mk3.0 experiments have been run with a 365 day - no leap year calendar and therefore always 28 days in February)

Experiment name	Mk3.0 Run name	Ensemble	Valid	
			Model Year	Calendar Year
PIcntrl	uk1	run1	1-500(500)	1751-2250
PIcntrl	ct3	run2	121-200(80)	2001-2080
20C3M	m20	run1	121-250(130)	1871-2000
20C3M	n20	run2	131-260(130)	1871-2000
20C3M	o20	run3	141-270(130)	1871-2000
Commit	s20	run1	251-353(103)	2001-2103
SRESA2	m20	run1	251-500(250)	2000-2250
SRESA1B	a1b	run1	251-450(200)	2001-2200
SRESB1	st1	run1	190-551(318)	1939-2300
1%to2×	cm3	run1	121-200(80)	2001-2080
Slabcntl	qv3	run1	1-60(60)	2001-2060
2×CO2	qv4	run1	1-60(60)	2001-2060

Table 3: Valid model and calendar year ranges for Mk3.0 experiments.

Experiment Name	Monthly		Daily	
	Model Year	Calendar Year	Model Year	Calendar Year
PIcntrl	121-500(380)	1871-2250	211-250(40)	1961-2000
PIcntrl	121-200(80)	2001-2080	181-200(20)	2061-2080
20C3M	121-250(130)	1871-2000	211-250(40)	1961-2000
20C3M	131-260(130)	1871-2000	221-260(40)	1961-2000
20C3M	141-270(130)	1871-2000	231-270(40)	1961-2000
Commit	251-350(100)	2001-2100	281-300(20)	2031-2050
			331-350(20)	2081-2100
SRESA2	251-350(100)	2001-2100	296-315(20)	2046-2065
			331-350(20)	2081-2100
SRESA1B	251-450(200)	2001-2200	296-315(20)	2046-2065
			331-350(20)	2081-2100
			431-450(20)	2181-2200
SRESB1	252-551(300)	2001-2300	297-316(20)	2046-2065
			332-351(20)	2081-2100
			432-451(20)	2181-2200
			532-551(20)	2281-2300
1%to2×	121-200(80)	2001-2080	181-200(20)	2061-2080
Slabcntl	41-60(20)	2041-2060	51-60(20)	2001-2020
2×CO2	51-60(20)	2041-2060	51-60(20)	2001-2020

Table 4: Model and calendar year ranges for Mk3.0 experiments for which output was sent to PCMDI.

7 Tables of Standard IPCC Parameters

This section details the individual parameters for which special processing was required. Information here may be of particular use to analysts using the Mk3.0 IPCC data who need to know if any local assumptions, derivations or approximations were made in forming the standard CSIRO Mk3.0 IPCC parameters. As the IPCC AR4 is primarily aimed at comparing results across modelling groups, this local information could prove invaluable when researchers are confronted with differences or similarities in results between the model generated (and observed) parameters.

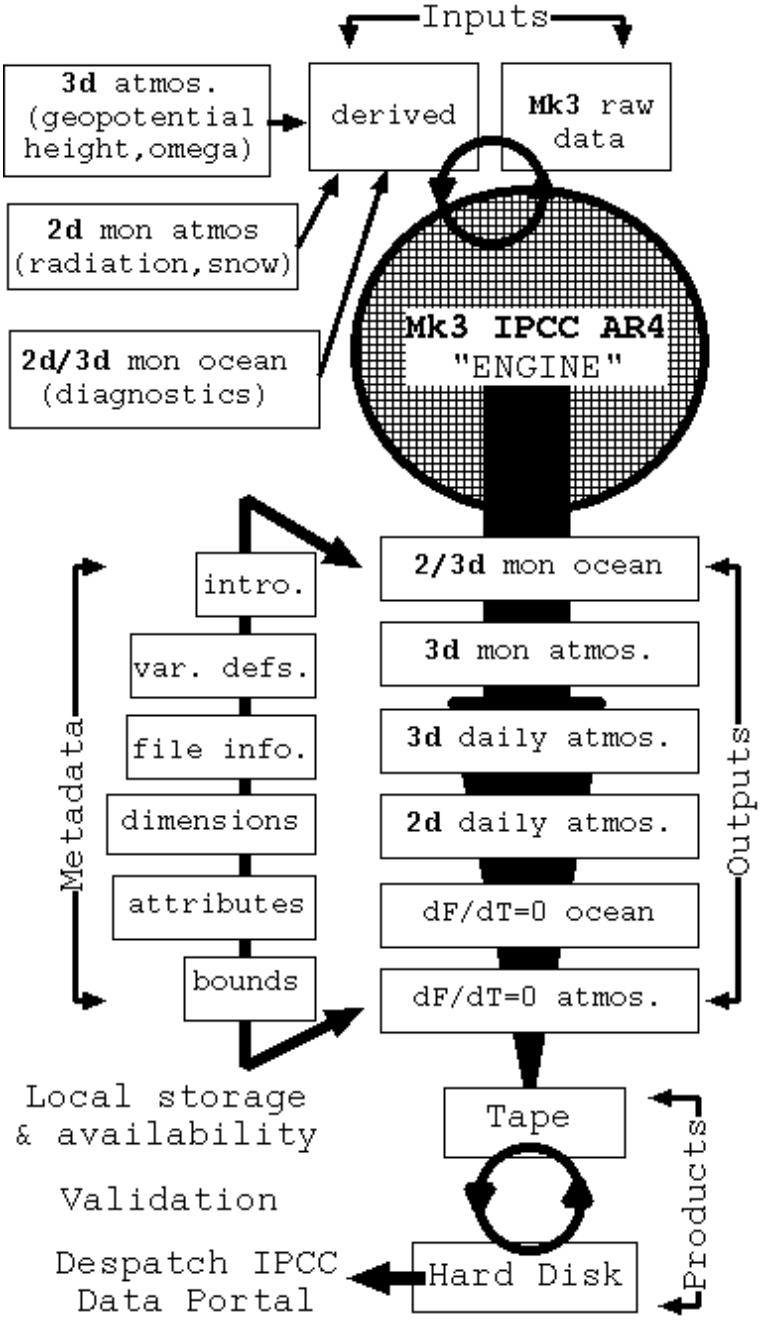


Figure 5: A schematic overview of the Mk3 processing system. See text for details.

As a way of describing the overall processing sytem Figure 5, included here from *Collier et al.* (2007), schematises the basic features of generating and delivering Mk3.0 IPCC output. In general

there are two types of data inputs which are fed into the Mk3 "engine". This engine is a central processing system that determines all requirements of the IPCC parameter request and attaches appropriate metadata to the designated netCDF output. These two inputs are either taken from raw Mk3.0 data or from derived quantities generated themselves from raw Mk3.0 data. There are far more IPCC parameters derived from the raw Mk3.0 data, however, only together are a comprehensive set of high priority IPCC parameters able to be generated. An important feature of the processing system is that the so called engine is kept "running" until it has worked through the entire request. In theory this could mean that a single computer submission could complete a whole task, even the generation of the full Mk3.0 set from raw and derived quantities. However, in practice a single Mk3.0 experiment requires a lot of computing time and in general, for technical computing reasons, numerous restarts of the engine would be required to complete the task. However, this poses no problems for the processing system as it can begin from where it started off from and is entirely "aware" of its previous successes or failures. Another useful characteristic of the processing system is that a part of the engine can be developed as it continues to run. The coding of the processing system for certain parameters in the IPCC standard output tables (see discussion on these later in the Section) were more difficult than for others and it were important that the basic system could continue to work on more easiy generated parameters whilst these technica issues were resolved. This was an extremely important feature as the original submission deadlines required that we got as much easy to genrate data delivered as soon as it were possible, and ultimately allowed the CSIRO model to be incuded in several high profile research artices.

Numerous technical issues associated with generating strict IPCC format data from raw Mk3 model data were overcome by the Mk3 processing system. For some parameters (e.g. mrros) it were necessary to interpolate over land and ocean seperately, to ensure realistic gradients near coastal regions. Some parameters (e.g. surface pressure) it was not just a simple case of interpolating from the reduced to the full grid, an algoritm was required to ensure a realistic representation. Other parameters (e.g. rldscs) were derived from other parameters at a range of resolutions. Care was required to obtain meaningful values at different parts of the globe and seasonally. A chief feature of the Tcl-Nap scripting language for generating the Mk3 IPCC output was its ability to spatially and temporally interpoate data from one grid to another, a missing capability of the CMOR software.

Ultimately we developed the Mk3 processing system by keeping a balance between computational efficiency and ease of its implementation and operational use. During the early Mk3 data submissions the IPCC data submission deadlines were tight and therefore a number of decisions were made as to what parameters we could realistically include in the initial submission and subsequent ones. The processing system works on the simple approach of identifying what files are required, defining their output form and definitions, and to write the file to disk. A detailed description of the processing system software is given in Section 7.

A lot of attention was placed on the quality of output from the Mk3 processing system. In particular, a seperate set of scripts and commands were used to examine the data put on the detachable hard disk used to submit the data to PCMDI. This additional process gave us confidence that we were writing the data in the appropriate format and that any run-time errors associated with the datafiles were identified so that they could be regenerated and replaced on the hard disk. This process was quite time-consuming as it regularly identified numerous problems, however, the time spent rectifying these problems before the hard disk was despatched to the U.S. more than made up for the time spent on external identification and repair of problem files. For information concerning this part of the Mk3 data submission please contact Martin Dix (email: martin.dix@csiro.au).

7.1 Monthly Atmosphere Parameters

7.1.1 2D Parameters

The 2D monthly IPCC parameters *tasmin* and *tasmax* were requested by PCMDI well after the IPCC Table A1a (Tables 5- 7) had been finalised, and were thus put into a new IPCC Table A1f (Table 8). Soil moisture quantities *morsos* and *mrso* are not available. There are various derived quantities, derived in the sense that they are not directly available from Mk3.0 raw output and an

intermediate step is required before final IPCC parameter extraction is possible.

IPCC parameters are generally processed through a one or a two step process approach. For a one step process, raw Mk3.0 parameters (see relevant ones in Tables 20- 24) are written out as IPCC parameters in one execution of the processing scripts. This is run from the main IPCC account IPCC on cherax. For the two step process, an initial processing step is required where the raw Mk3.0 parameters are processed an intermediate processed file is placed in the disk area when the experiment was executed. In other words, this step needs to be executed on the account in which the experiment was performed. A final step, akin to the one step process, is performed whether the intermediate files are converted into final IPCC parameters. As for the one step process, the second step is run from the main IPCC account *IPCC* on cherax. This distinction will be used from time to time in explaining how the parameters in the IPCC tables were satisfied.

The IPCC parameters *uas*, *vas* and *huss* have been computed from Mk3.0 histb files, and is therefore a two-step process. Parameters *uas* and *vas* available in a one-step process experiments *qv3* (Slabcntl), *qv4* (2×CO2), *n20* (20C3M 2^{nd} ensemble) and *o20* (20C3M 3^{rd} ensemble). Parameter *huss* is available in a one-step process for experiments *n20* and *o20*.

The IPCC parameters *rsntp*, *rlntp*, *rsntpcs* and *rlntpcs* in Table 7 are available in a one-step process for experiments *qv3* (Slabcntl), *qv4* (2×CO2), *n20* (20c3M 2^{nd} ensemble) and *o20* (20c3M 3^{rd} ensemble) only. For other experiments these parameters do not exist. IPCC parameter *rsdt* is the same for all experiments, based on a monthly annual cycle and interpolated in time at each model time-step. The IPCC parameters *mrros*, *snw*, *snc*, *uas*, *vas*, *huss*, *rsdt*, *rtmt*, *rsdscs*, *rsuscs*, *rldscs* in Tables 5- 7 are derived from raw Mk3.0 parameters for all experiments, and details of the scripts and the process used is described in Table 19. The IPCC parameter *ps* is derived in a two-step process for all experiments except *qv3* (Slabcntl), *qv4* (2×CO2), *n20* (20c3M 2^{nd} ensemble) and *o20* (20c3M 3^{rd} ensemble) where it is available in a one-step process. The IPCC parameters *rsntp*, *rlntp*, *rsntpcs* and *rlntpcs* are not available from a bulk of the experiments, however, for *qv3* (Slabcntl), *qv4* (2×CO2), *n20* (20c3M 2^{nd} ensemble) and *o20* (20c3M 3^{rd} ensemble) they do exist.

The following IPCC (two-step process) parameter relationships exist with Mk3.0 parameters in Tables 5-7:

$$mrros = run - per$$

$$rsus = sgd - sgn$$

$$snc = \begin{cases} 100 & \text{if snd>0} \\ 0 & otherwise \end{cases}$$

$$snw = \begin{cases} 0 & \text{if ico>0} \\ snd & otherwise \end{cases}$$

$$rsdscs = sgd \times \frac{sgc}{sgn}$$

$$rldscs = rgn + rgd - rgc$$

$$rtmt = sit - sot - rtu$$

$$rlus = rgn - rgd$$

The three character Mk3.0 parameters that appear above (and whose units and resolutions are shown in Tables 20 and 21) and are not defined by direct relationships between IPCC and Mk3.0

parameters shown in Tables 5- 18 are defined as follows

per ⇒ Soil Percolation (mm day^{-1})
rgc ⇒ Net Surface Clear-Sky Longwave (W m^{-2})
rgn ⇒ Surface Net Longwave Radiation (W m^{-2})
sgc ⇒ Net Surface Clear-Sky Longwave (W m^{-2})
sgn ⇒ Surface Net Shortwave Radiation (W m^{-2})

Together with the definitions provided in the tables of Section 7, this completes the full set of Mk3.0 parameters used in the IPCC parameter generation.

The net heat flux, hfo_{atm}[7] into the ocean/land is a combination of the radiative and turbulent heat fluxes

$$hfo_{atm} = rlds - rlus + rsds - rsus - hfls - hfss \tag{1}$$

The net freshwater flux, wfo_{atm} into the ocean/land is a combination of precipitation, evaporation and continental runoff

$$wfo_{atm} = pr + \frac{hfls}{-28.94} + mrro \tag{2}$$

which is derived from the atmospheric component in the Mk3.0 model. The quantity wfo_{atm} is closely related to the IPCC parameter wfo in Tables 5- 7 which is derived from the ocean component in the Mk3.0 model, however, differences may be detected particularly in the regions of sea ice. The evaporation is related to the IPCC parameter $hfls$ by the scaling term -28.94 which is the approximation used in the generation of $hfls$ in Tables 5- 7 and 14.

7.1.2 3D parameters

The IPCC parameter *cl* in Table 9 is available only for experiments *qv3* (Slabcntl), *qv4* (2×CO2), *n20* (20c3M 2^{nd} ensemble) and *o20* (20c3M 3^{rd} ensemble). The parameters *wap* and *zg* need to be defined and are processed through an intermediate step by the scripts described in Table 19. It is an IPCC requirement that masking out of points below the land surface be performed. This is presumably so that analysts, where these masked points are to be filled in, who wish to compare parameters across a range of AR4 models can utilise their own vertical interpolation scheme for these fictitious points. This switch to mask can be turned off in the processing script ipcc_calc_4dmon_mk3agcm.tcl found in Section 14.18 if required. The IPCC parameter *tro3* is not available for the Mk3.0 experiments.

7.2 Monthly ocean parameters

7.2.1 1D parameters

The Northward Ocean Heat Transport *hfogo* in Table 10 at latitude φ are calculated by integrating the flux of ocean surface heat, *hfo*, from the southern most point ϕ_s to ϕ:

$$hfogo(\varphi) = \int_{\phi_s}^{\phi} \int_{\lambda_w}^{\lambda_e} hfo\, r_e^2 cos(\phi) d\lambda d\phi$$

where r_e^2 is the Earth's radius. λ_w, λ_e are the eastern and western boundaries of the basin. This has been performed using Tcl-Nap scripts ipcc_hfogo_ferret.tcl[8] and ipcc_hfogo.tcl.

[7] See reference to hfo in Section 7.2.1, the subscript *atm* refers to heat leaving the atmosphere for the ocean/land
[8] The Ferret software package has been called as an external function here.

7.2.2 2D Parameters

The Meridional Overturning Streamfunction *stfmmc* in Table 11 which denotes the zonally integrated net volume transport rate is given by

$$stfmmc(\varphi, z) = -\int_{zb}^{z}\int_{\lambda_w}^{\lambda_e} \upsilon(\lambda, \varphi, z')r_e cos(\varphi)d\lambda dz'$$

where z_b is the local bottom depth where the streamfunction is defined to be zero, and υ is the Eulerian mean transport. φ, λ and z are the latitude, longitude and depth, respectively, and r_e^2 is the Earth's radius. The zonal definite integral can be conducted either along a whole zonal circle in the global ocean case or starting from the western boundary (λ_w) to the eastern boundary (λ_e) for individual basins.

This has been performed using Tcl-Nap scripts ipcc_stfmmc_ferret.tcl[2]. Although both IPCC parameters *hfogo* and *stfmmc* are both two-step processes, they are both performed on the main IPCC account IPCC on cherax. This is made possible by storing the output from the first step of the two-step process under the IPCC account (in IPCC/data/mk3.0/runname).

IPCC parameters *zos* and *rslr* are not predicted by the ocean component of Mk3.0 (MOM2.2) and therefore must be derived, taking into account pressure and steric effects. The control simulations are used in this computation and because of the method used, an elevation is not available for the control experiment. The IPCC parameter *tos* in Table 12 is equivalent to the first level (k=1) *thetao* listed in Table 13. It actually represents the average temperature in the ocean components first layer but for climate applications is an adequate approximation of the sea-surface temperature. Two further IPCC-like parameters, *sos* and *hfo*, have been generated which are salt and heat analogs of the IPCC parameters *tos* and *wfo*. As they are not IPCC requested parameters they do not appear in Table 12 but are in the local archive (not being served by PCMDI) because of their importance to past, ongoing and potential studies. The IPCC parameters *hfcorr*, *wfcorr*, *tauucorr* and *tauvcorr* are not available for the Mk3.0 model where no corrections to heat, moisture and momentum are employed.

7.2.3 3D parameters

The IPCC parameter *rhopoto* in Table 13 has been derived using script ipcc_rhopoto.tcl from parameters *so* and *thetao* using the Unesco (1981) formulation. In other words Mk3.0 IPCC salinity and temperature output is used as input to this Tcl-Nap script and therefore must be in existence before the density calculation can be performed.

Item	IPCC Parameter Name	CF Standard Name	Units
	CF Long Name	**Mk3.0 Parameter Name**	
1	psl	air_pressure_at_sea_level	Pa
	Sea Level Pressure	psl	
2	pr	precipitation_flux	kg m^{-2} s^{-1}
	Precipitation	rnd	
3	tas	air_temperature	K
	Surface Air Temperature	tsc	
4	mrsos	moisture_content_of_soil_layer	kg m^{-2}
	Moisture in Upper 0.1 m of Soil Column	not available	
5	mrso	soil_moisture_content[2]	kg m^{-2}
	Total Soil Moisture Content	not available	
6	tauu	surface_downward_eastward_stress	Pa
	Zonal Surface Wind Stress	tax	
7	tauv	surface_downward_northward_stress	Pa
	Meridional Surface Wind Stress	tay	
8	snd	surface_snow_thickness	m
	Snow Depth	snd	
9	hfls	surface_upward_latent_heat_flux	W m^{-2}
	Surface Latent Heat Flux	evp	
10	hfss	surface_upward_sensible_heat_flux	W m^{-2}
	Surface Sensible Heat Flux	hfl	
11	rlds	surface_downwelling_longwave_flux	W m^{-2}
	Surface Downwelling Longwave Radiation	rgd	
12	rlus	surface_upwelling_longwave_flux	W m^{-2}
	Surface Upwelling Longwave Radiation	to derive/rlus	
13	rsds	surface_downwelling_shortwave_flux	W m^{-2}
	Surface Downwelling Shortwave Radiation	sgd	
14	rsus	surface_upwelling_shortwave_flux	W m^{-2}
	Surface Upwelling Shortwave Radiation	to derive/rsus	
15	ts	surface_temperature	K
	Surface Skin Temperature	tsu	

Table 5: IPCC Table A1a: Monthly Mean Atmosphere + Land Surface 2-D (latitude, longitude) Data.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
16	ps	surface_air_pressure	Pa
	Surface Pressure	to derive/ps	
17	prsn	snowfall_flux	kg m^{-2} s^{-1}
	Snowfall Flux	sno	
18	prc	convective_precipitation_flux	kg m^{-2} s^{-1}
	Convective Precipitation Flux	rnc	
19	prw	atmosphere_water_vapor_content	kg m^{-2}
	Precipitable Water	pwc	
20	mrfso	soil_frozen_water_content	kg m^{-2}
	Soil Frozen Water Content	not available	
21	mrros	surface_runoff_flux	kg m^{-2} s^{-1}
	Surface Runoff	to derive/mrros	
22	mrro	runoff_flux	kg m^{-2} s^{-1}
	Surface and Subsurface Runoff	run	
23	snw	surface_snow_amount	kg m^{-2}
	Snow Amount	to derive/snw	
24	snc	surface_snow_area_fraction	percent
	Snow Area Fraction	to derive/snc	
25	snm	surface_snow_melt_flux	kg m^{-2} s^{-1}
	Snow Melt	not available	
26	uas	eastward_wind	m s^{-1}
	Zonal Surface Wind Speed	to derive/suas	
27	vas	northward_wind	m s^{-1}
	Meridional Surface Wind Speed	to derive/svas	
28	huss	specific_humidity	1
	Surface Specific Humidity	to derive/shus	
29	rsdt	toa_incoming_shortwave_flux	W m^{-2}
	TOA Incident Shortwave Radiation	to derive/rsdt	
30	rsut	toa_outgoing_shortwave_flux	W m^{-2}
	TOA Reflected Shortwave Radiation	sot	

Table 6: IPCC Table A1a (cont'd): Monthly Mean Atmosphere + Land Surface 2-D (latitude, longitude) Data.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
31	rlut	toa_outgoing_longwave_flux	W m^{-2}
	Outgoing Longwave Radiation	rtu	
32	rtmt	net_downward_radiative_flux_at_top_of_atmosphere_model	W m^{-2}
	Net Flux at Top of Model	to derive/rtmt	
33	rsntp	net_downward_shortwave_flux_in_air	W m^{-2}
	Net Downward Shortwave Flux at 200 hPa	sf2	
34	rlntp	net_upward_longwave_flux_in_air	W m^{-2}
	Net Upward Longwave Flux at 200 hPa	rf2	
35	rsntpcs	net_downward_shortwave_flux_in_air_assuming_clear_sky	W m^{-2}
	Net Downward Clear-Sky Flux at 200 hPa	sc2	
36	rlntpcs	net_upward_longwave_flux_in_air_assuming_clear_sky	W m^{-2}
	Net Upward Clear-Sky Longwave Flux	rc2	
	at 200 hPa	rc2	
37	rsdscs	surface_downwelling_shortwave_flux_assuming_clear_sky	W m^{-2}
	Incident Surface Clear-Sky Shortwave	to derive/rsdscs	
38	rsuscs	surface_upwelling_shortwave_flux_assuming_clear_sky	W m^{-2}
	Reflected Surface Clear-Sky Shortwave	to derive/rsuscs	
39	rldscs	surface_downwelling_longwave_flux_assuming_clear_sky	W m^{-2}
	Downwelling Surface Clear-Sky Longwave	to derive/rldscs	
40	rlutcs	toa_outgoing_longwave_flux_assuming_clear_sky	W m^{-2}
	Outgoing Clear-Sky Longwave	rtc	
41	rsutcs	toa_outgoing_shortwave_flux_assuming_clear_sky	W m^{-2}
	Reflected Clear-Sky Shortwave	soc	
42	clt	cloud_area_fraction	percent
	Total Cloud Fraction	cld	
43	clwvi	atmosphere_cloud_condensed_water_content	kg m^{-2}
	Column Integrated Cloud Water Content	lwp	
44	clivi	atmosphere_cloud_ice_content	kg m^{-2}
	Column Integrated Cloud Ice Content	iwp	

Table 7: IPCC Table A1a (cont'd): Monthly Mean Atmosphere + Land Surface 2-D (latitude, longitude) Data.

Item	IPCC Parameter name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
1	tasmin	air_temperature	K
	Minimum Daily Surface Air Temperature	tsl	
2	tasmax	air_temperature	K
	Maximum Daily Surface Air Temperature	tsh	

Table 8: IPCC Table A1f: Monthly-mean surface parameters, and prescribed land surface characteristics.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
1	cl	cloud_area_fraction_in_atmosphere_layer	percent
	Cloud amount	cl	
2	ta	air_temperature	K
	Air Temperature	t4d	
3	ua	eastward_wind	m s^{-1}
	Eastward Wind	u4d	
4	va	northward_wind	m s^{-1}
	Northward Wind	v4d	
5	hus	specific_humidity	1
	Specific Humidity	q4d	
6	wap	omega	Pa s^{-1}
	Vertical Motion (omega)	w4d	
7	zg	geopotential_height	m
	Geopotential Height	to derive/z4d	
8	hur	relative_humidity	percent
	Relative humidity	r4d	
9	tro3	mole_fraction_of_o3_in_air	1e-9
	Ozone Concentration	not available	

Table 9: IPCC Table A1c: Table A1c: Monthly Mean Atmosphere 3-D Data.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
1	hfogo	northward_ocean_heat_transport	W
	Northward Ocean Heat Transport	to derive/hfogo	

Table 10: Table O1a: Monthly Mean Ocean 1-D (latitude) Data.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
1	stfmmc	ocean_meridional_overturning_streamfunction	m^3 s^{-1}
	Meridional Overturning Streamfunction	to derive/stfmmc	

Table 11: Table O1b: Monthly Mean Ocean 2-D (latitude, depth) Data.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
1	zos	sea_surface_elevation	m
	Sea Surface Elevation	zos	
2	tos	sea_surface_temperature	K
	Sea Surface Temperature	tos	
3	sic	sea_ice_area_fraction	percent
	Sea Ice concentration	ico	
4	sit	sea_ice_thickness	m
	Sea Ice thickness	icd	
5	usi	eastward_sea_ice_velocity	m s^{-1}
	Eastward Sea Ice Velocity	icu	
6	vsi	northward_sea_ice_velocity	m s^{-1}
	Northward Sea Ice Velocity	icv	
7	wfo	water_flux_into_ocean	kg m^{-2} s^{-1}
	Water Flux Into Ocean	pme	
8	stfbarot	ocean_barotropic_streamfunction	$m^3 s^{-1}$
	Oceanic Barotropic Streamfunction	psi	
9	hfcorr	heat_flux_correction	W m^{-2}
	Heat Flux correction	not available	
10	wfcorr	water_flux_correction	kg $m^{-2} s^{-1}$
	Water Flux Correction	not available	
11	tauucorr	eastward_momentum_flux_correction	Pa
	Eastward Momentum Flux Correction	not available	
12	tauvcorr	northward_momentum_flux_correction	Pa
	Northward Momentum Flux Correction	not available	
13	rslr	relative_sea_level_rise	m
	Relative Sea Level Rise	rslr	

Table 12: Table O1c: Monthly Mean Ocean + Sea Ice 2-D (latitude, longitude) Data.

Item	**IPCC Parameter Name**	**CF Standard Name**	**Units**
	Long Name	**Mk3.0 Parameter Name**	
1	so	sea_water_salinity	1e-3
	Salinity	Salt	
2	thetao	sea_water_potential_temperature	K
	Potential Temperature	Temp	
3	rhopoto	sea_water_potential_density	kg m^{-3}
	Potential Density	to derive/rhopoto	
4	uo	eastward_sea_water_velocity	m s^{-1}
	Eastward Sea Water Velocity	uvel	
5	vo	northward_sea_water_velocity	m s^{-1}
	Northward Sea Water Velocity	vvel	
6	wo	upward_sea_water_velocity	m s^{-1}
	Upward Sea Water Velocity	wvel	

Table 13: Table O1e: Monthly Mean Ocean 3-D Data.

7.3 Daily parameters

7.3.1 2D Parameters

The IPCC parameters *uas* and *vas* in Table 14 have been approximated as the lowest sigma level (0.991) values from each of the respective wind components for all experiments except for the Mk3.0 *o20* (20c3M 3rd ensemble) experiment where it is available in the raw (hista) model files. Rather than establish these derived varibles in a separate file as was done in the monthly output case, they are derived with the main processing script ipcc_calc_day_mk3agcm.tcl. The IPCC parameters *hfss*, *rlus*, *rlds* and *rsus* are only available for the experiments *qv3* (Slabcntl), *qv4* (2×CO2), *n20* (20c3M 2^{nd} ensemble) and *o20* (20c3M 3^{rd} ensemble).

As for the monthly output, the follow IPCC parameter relationships exist with Mk3.0 parameters in Table 14:

$$rsus = sgd - sgn$$

$$rlus = rgn - rgd$$

7.3.2 3D Parameters

The *n20* (20c3M 2^{nd} ensemble) is the only experiment in which daily output has not been generated from daily raw model data, a simple average of the 4-times daily values has been used in that case and the relevant software is written to account for higher frequency output. Note that the IPCC parameter *zg* is not required for submission, however, the Tcl-Nap script for processing the parameters (ipcc_calc_4dday_mk3agcm.tcl) in Table 15 can generate its output. As for the 3D monthly parameters masking is required for grid points that are deemed below the surface of the land, however, the processing software can turn this feature off if required.

Item	IPCC Parameter Name	CF Standard Name	Units
	Long Name	**Mk3.0 Parameter Name**	
1	psl	air_pressure_at_sea_level	Pa
	Sea Level Pressure	psl	
2	pr	precipitation_flux	kg m^{-2} s^{-1}
	Precipitation	rnd	
3	tasmin	air_temperature	K
	Minimum Daily Surface Air Temperature	tsl	
4	tasmax	air_temperature	K
	Maximum Daily Surface Air Temperature	tsh	
5	tas	air_temperature	K
	Surface Air Temperature	tsc	
6	hfls	surface_upward_latent_heat_flux	W m^{-2}
	Surface Latent Heat Flux	evp	
7	hfss	surface_upward_sensible_heat_flux	W m^{-2}
	Surface Sensible Heat Flux	hfl	
8	rlds	surface_downwelling_longwave_flux	W m^{-2}
	Surface Downwelling Longwave Radiation	rgd	
9	rlus	surface_upwelling_longwave_flux	W m^{-2}
	Surface Upwelling Longwave Radiation	to derive/rlus	
10	rsds	surface_downwelling_shortwave_flux	W m^{-2}
	Surface Downwelling Shortwave Radiation	sgd	
11	rsus	surface_upwelling_shortwave_flux	W m^{-2}
	Surface Upwelling Shortwave Radiation	to derive/rsus	
12	uas	eastward_wind	m s^{-1}
	Zonal Surface Wind Speed	to derive/uas	
13	vas	northward_wind	m s^{-1}
	Meridional Surface Wind Speed	to derive/vas	
14	rlut	toa_outgoing_longwave_flux	W m^{-2}
	Outgoing Longwave Radiation	rtu	

Table 14: Table A2a: Daily Mean Atmosphere 2-D (latitude, longitude) Data.

Item	**IPCC Parameter Name**	**CF Standard Name**	**Units**
	Long Name	**Mk3.0 Parameter Name**	
1	ta	air_temperature	K
	Temperature	t4d	
2	ua	eastward_wind	m s^{-1}
	Zonal Wind Component	u4d	
3	va	northward_wind	m s^{-1}
	Meridional Wind Component	v4d	
4	hus	specific_humidity	1
	Specific Humidity	q4d	

Table 15: Table A2b: Daily Mean Atmosphere 3-D Data.

7.4 Time Independent Parameters

7.4.1 2D Atmosphere Parameters

The parameters in Table 16 have been generated using a suite of Tcl-Nap scripts which follow a consistent naming convention that, in particular, bears the parameter name (namely ipcc_orog.tcl, ipcc_sftlf.tcl and ipcc_sftgif.tcl). The input data for *sftlf* was taken from the histb file of the *cm3* (1%to2×) experiment and is exactly the same for the other experiments. Parameter *sftlf* uses input file Mk3_agcm.nc in each experiment. Each of the parameters listed are written out consistently for each experiment. Typically these parameters are generated swiftly and do not need to be submitted to the batch processing system.

Item	**IPCC Parameter Name**	**CF Standard Name**	**Units**
	Long Name	**Mk3.0 Parameter Name**	
1	orog	surface_altitude	m
	Surface Altitude	to derive/orog	
2	sftlf	land_area_fraction	percent
	Land Area Fraction	to derive/sftlf	
3	sftgif	land_ice_area_fraction	percent
	Fraction of Land Area Covered with Glacier	to derive/sftgif	
4	mrsofc	soil_moisture_content_at_field_capacity	kg m^{-2}
	Capacity of Soil to Store Water	not available	

Table 16: Table A1b: Time-independent Land Surface 2-D (latitude, longitude).

7.4.2 2D Ocean Parameters

The IPCC parameter *zobt* in Table 17 is computed using Tcl-Nap script ipcc_zobt.tcl. The IPCC parameter *qflux* in Table 17 is only valid for experiments *qv3* (Slabcntl), *qv4* (2×CO2). It is based on a monthly annual cycle and is interpolated in time at each model time-step and put into IPCC form using the Tcl-Nap script ipcc_qflux.tcl. Typically these parameters are generated swiftly and do not need to be submitted to the batch processing system.

Item	**IPCC Parameter Name**	**CF Standard Name**	**Units**
	Long Name	**Mk3.0 Parameter Name**	
1	zobt	sea_floor_depth	m
	Sea Floor Depth	to derive/zobt	
2	qflux	prescribed_heat_flux_into_slab_ocean	W m^{-2}
	Q-Flux	to derive/qflux	

Table 17: Table O1d: Time-independent Ocean 2-D.

7.5 Extremes Indices

The 10 IPCC parameters (indices) shown in Table 18 are computed from temperature and precipitation (IPCC parameters *tasmin*, *tasmax*, *tas* and *pr*) and require daily input (see Table 14) for their calculation with a single value for each year for each geographical location. Note that IPCC has requested only certain decades of daily data from the Mk3.0 model (see Table 4). As an index is required for each year for which daily data exists (see Table 3) it is necessary to create daily temperature and precipitation inputs for each valid year. The processing software that generates these indices has the ability to generate the indices on other time basis (e.g. seasonally and monthly) and to process daily input from other sources, including reanalysis, model and observational products. However, for the purposes of what IPCC has requested (the focus of this report), these indices are annual ones. The extreme indices Tcl-Nap software will be described in a future report, including an summary results from both Mk3.0 and Mk3.5d experiments. Typically these parameters are generated slowly, may consume large amounts of machine memory, and will therefore need to be submitted to the batch processing system.

Item	**IPCC Parameter Name**	**CF Standard Name**	**Units**
	Long Name	**Mk3.0 Parameter Name**	
1	fd	fd	days
	Total Number of Frost Days in Year	tsl	
2	etr	etr	K
	Intra-Annual Extreme Temperature Range	tsl/tsh	
3	gsl	gsl	days
	Growing Season Length	tas	
4	hwdi	hwdi	days
	Heat Wave Duration Index	tsh	
5	tn90	tn90	%
	Percent of time Tmin > the 90^{th} percentile value of	tsl	
	daily minimum temperature		
6	r10	r10	days
	Number of Days for which Precipitation Rate Exceeds	rnd	
	10 mm day^{-1}		
7	cdd	cdd	days
	Maximum Number of Consecutive Dry Days	rnd	
8	r5d	r5d	kg m-2
	Maximum 5-day Precipitation Total	rnd	
9	sdii	sdii	kg m-2 s-1
	Simple Daily Precipitation Index	rnd	
10	r95t	r95t	%
	Percent of Annual Total Precipitation due to Events	rnd	
	Exceeding the 1961-1990 95^{th} Percentile		

Table 18: Table A4: Extremes indices (latitude, longitude, time:year).

8 Illustrative Figures of Selected Parameters for all Mk3.0 IPCC Experiments

At the beginning of the following section a number of figures are presented illustrating a particular application of an IPCC parameter in the generation of an atmosphere or ocean based index often appearing in the scientific literature. This has the purpose of 1) illustrating the index derivation from basic parameters and 2) identifying realistic behaviour in these indices in the Mk3.0 model when compared with observations. Finally, a set of air-sea coupling parameters are presented to illustrate the realistic behaviour of heat, water and mass exchanges in a 1961-1990 ensemble average from the Mk3.0 model.

8.0.1 Surface Monthly Air-Temperature Anomalies

Shown in Figure 6 is the globally area averaged surface monthly air temperature anomalies, IPCC parameter *tas* listed in Table 5, smoothed by a 121-month running mean to give a more clear indication of long term trends. Results in figure 6a are relative to the 1961-1990 of the 20C3M experiment ensemble 1,2 and 3 average. Results in figure 6b are relative to the control from the full CMIP (1%to2× and PIcntrl 2^{nd} ensemble) and Slab-Ocean (Slabcntl and 2×CO2) experiments.

These figures describes many practical aspects concerning the Mk3.0 experiments and some key feature concerning the general behaviour, sensitivity and performance of the Mk3.0 model. Firstly, the calendar relationships between the control and transient experiments are apparent. Note that although a calendar has been assigned to the control experiments, this is an arbitrary choice. This is why we have placed the results from the CMIP and Slab-Ocean experiments on a separate graph, Figure 6b, where the time axis has nothing to do with the year 2000. In the case of Figure 6a we have aligned the main control experiment *uk1* (PIcntrl 1^{st} ensemble) to the *m20* (20C3M 1^{st} ensemble). In most cases the calendar years over which the model runs will not give describe the underlying model years. For example, although the three 20C3M ensembles have the calendar years 1871-2000, the underlying model years are 241, 251 and 261 for the 1^{st}, 2^{nd} and 3^{rd} ensembles, respectively. References to Table 4 will help to clarify the underlying model years of each experiment.

The *uk1* (PIcntrl 1^{st} ensemble) experiment reveals a small but steady drift. It should be noted that this cooling trend could have been taken into account in the transient experiment time series that are initiated from it.

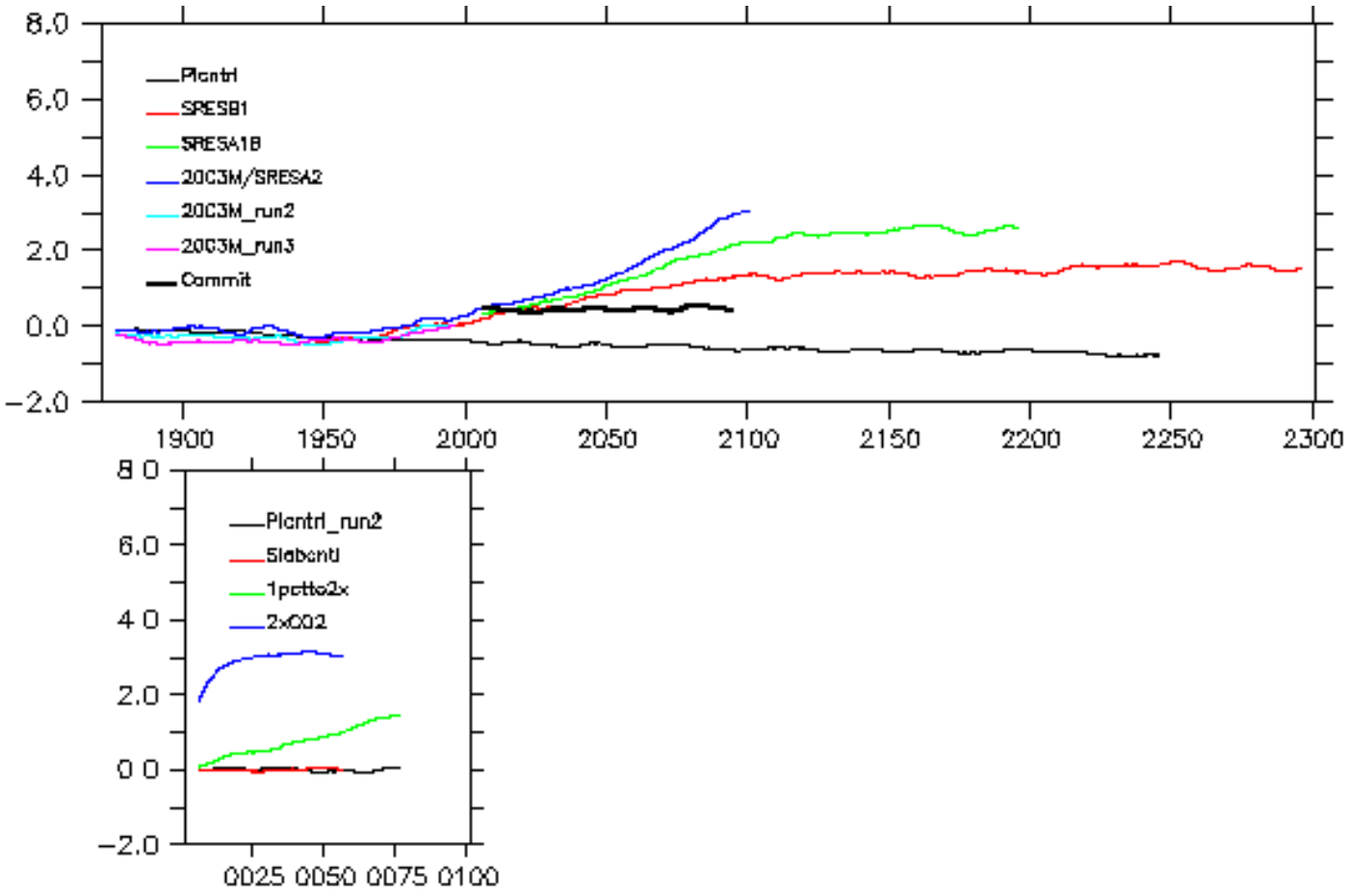


Figure 6: Globally averaged surface monthly air temperature anomalies for the a) control, 20^{th} Century and SRES experiments and b) CMIP and Slab Ocean experiments (°C). A 121-month running mean has been applied. See text for details.

8.0.2 Niño3.4 Index

The Niño3.4 Index is an area averaged value of anomalous monthly sea-surface temperatures over a region in the central to eastern Pacific defined by the latitude/longitude box 5°N-4°S and 170°W-120°W. It is most often used as an indicator of the strength and phase of the El-Niño Southern Oscillation (ENSO, in conjunction with its atmospheric counterpart, the SOI, see Section 14.31) an atmospheric/ocean phenomenon that severely impacts upon the climate and weather, predominantly through ocean temperatures and atmospheric rainfall, of the Pacific rim and further afield through atmospheric/ocean teleconnections. The Niño3.4 index shown in Figure 7 is derived from IPCC parameter *tos* listed in Table 12, smoothed by a 3 year running mean to bring out the observed 2-7 year periodicity of the ENSO behaviour. Results in figure 7a are relative to the 1961-1990 of the 20C3M experiment ensemble 1,2 and 3 average. Results in figure 7b are relative to the control from the full CMIP (1%to2× and PIcntrl 2^{nd} ensemble) experiments. Note that there are no results for the Slab-Ocean experiments, Slabcntl and 2×CO2), as they do not have an active ocean, only a prescribed sea-surface temperature. Although there is an apparent increasing trend in the Niño3.4 Index for the range of SRES and 1%to2× experiment in Figure 7, this does not necessarily mean that the ENSO behaviour in the Mark3.0 model is becoming more intense or variable under a warming earth scenario, and that the trend is more likely to be an artefact of the warming itself, and the base climatology that is used. Clarification of this point may be achieved by forming anomalies relative to a moving-window climatology, say over a 20-30 year period, for results from throughout the whole experiment. However, this additional analysis is beyond the focus of this report.

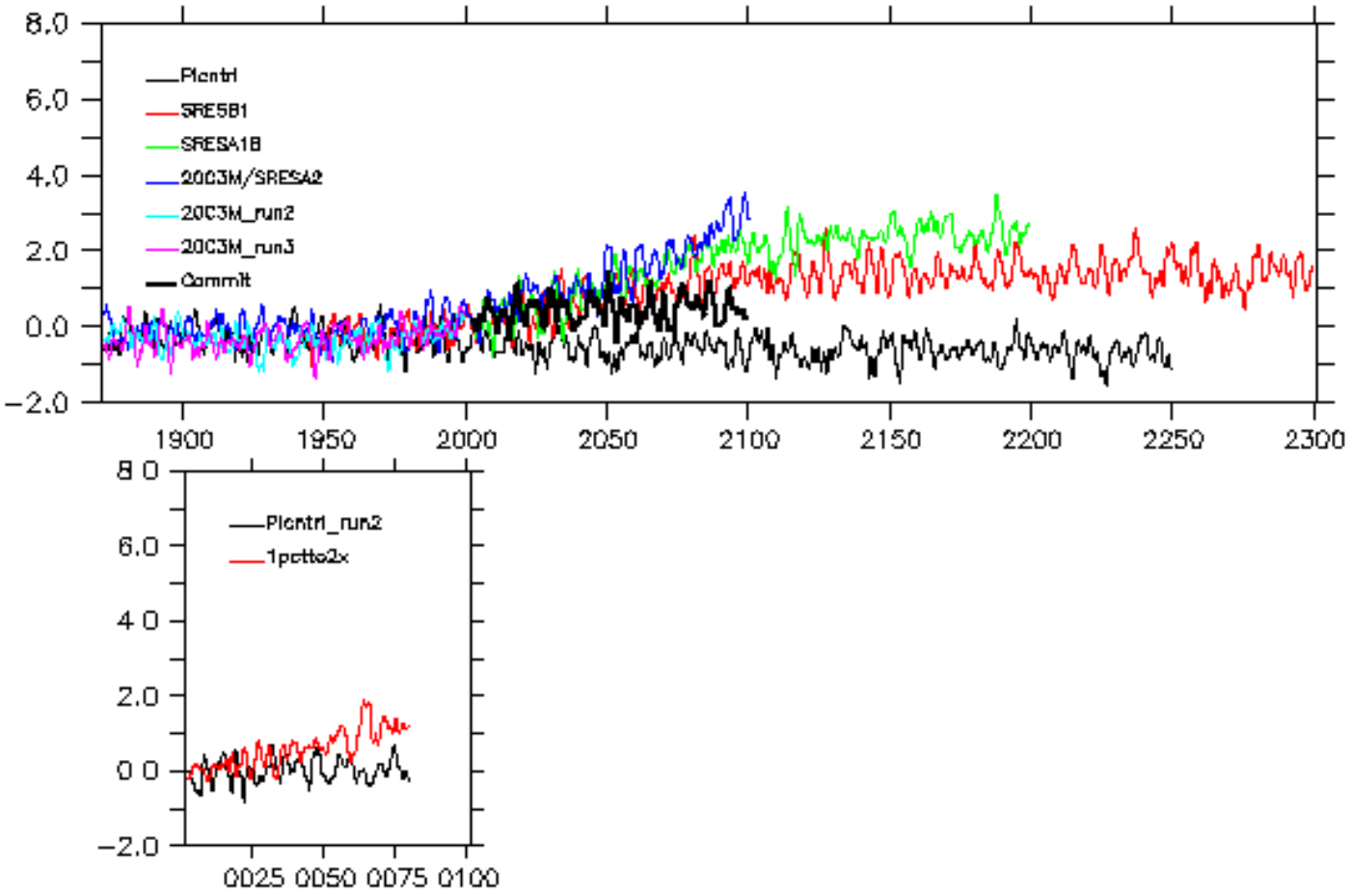


Figure 7: Niño3.4 Index, where monthly sea-surface temperature anomalies are averaged over the region 5°N-4°S and 170°W-120°W, for the a) control, 20^{th} Century and SRES experiments and b) CMIP and Slab Ocean experiments (°C). A 3-year running mean has been applied. See text for details.

8.0.3 Southern Oscillation Index

The Southern Oscillation (SO) Index is calculated from the monthly or seasonal fluctuations in the air pressure difference (in this case using IPCC parameter *psl* listed in Table 5) between Tahiti (eastern South Pacific region) and Darwin (Indian Ocean/Indonesia region) using the standard method of normalising this difference by its long-term standard deviation, and scaling by ten. The SO Index shown in Figure 8 helps to quantify the finding that when the South Pacific High is weak, equatorial sea-surface temperatures in the Pacific are relatively high along with rainfall *Rogers* (1984). Swings in the prevailing flows can last for several decades and it is this persistence that makes it particularly useful in seasonal and long-range forecasting *Chen* (1982). There is no sign of significantly different behaviour between experiments where forcings are constant (PIcntrl and Commit) or the SRES warming scenarios, although only a rigorous analysis would confirm this. To make identification of each experiment easier, and to place focus on longer period variability, a strong smoothing has been applied. A curious reponse in the SO Index is shown by the 2×CO2 experiment, with a sudden drop in value during the first 40 years or so, followed by a 50% recovery to normal values by year 60. Although this experiment represents an unrealistically (high and sudden) release of atmospheric warming CO_2, information is provided on the model's sensitivity to a sudden change and its impact on the large scale pressure differential, represented by the SO Index, across the Pacific, and the model atmosphere's ability to return to somewhat normal level in the SO index. Numerous studies have examined the characteristics of and the relationship between the SO and ENSO (and their respective indices as described, for example, in Sections 8.0.3 and 8.0.2)

because of the Pacific Ocean's large scale pressure field evolving in response to the persistent sea-surface temperature changes in the equatorial Pacific. That analysis is beyond the scope of this report, however, a correlation of 0.N exists between the Mk3.0 model control run (PIcntrl 1^{st} ensemble) Niño3.4 and SO Index shown in Figures 8 and 8.0.2.

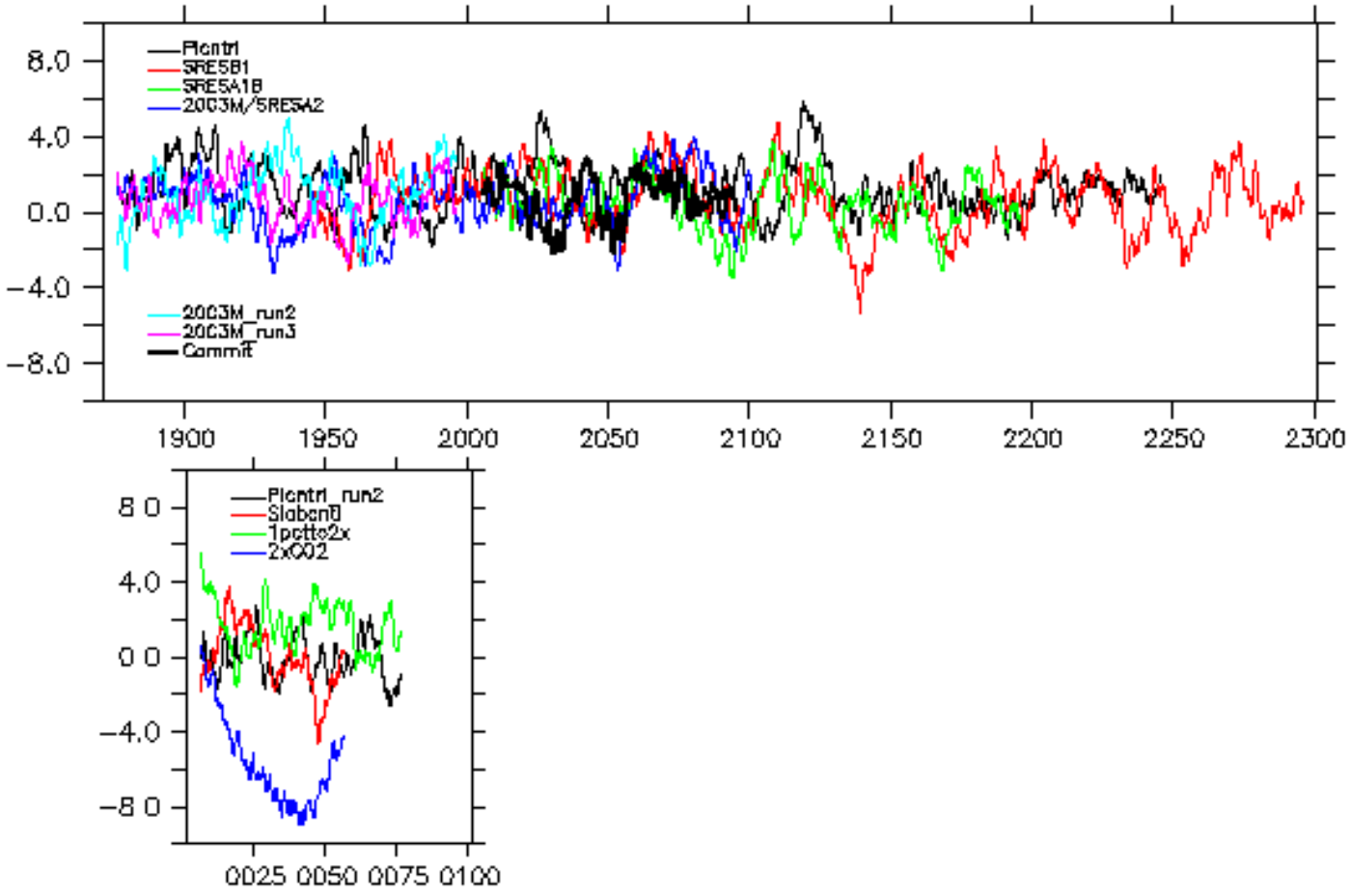


Figure 8: Southern Oscillation Index for the a) control, 20^{th} Century and SRES experiments and b) CMIP and Slab Ocean experiments. A 121-month running mean has been applied. See text for details.

8.0.4 North Atlantic Oscillation Index

The NAO Index is calculated by normalising the pressure difference (in this case using IPCC parameter *psl* listed in Table 5) between Ponta Delgadas, Azores, and Akureyri, Iceland, measuring variations in both the subtropical anticyclone belt and the subpolar low, respectively. Positive (negative) values of the index are associated with strong (weak) zonal flow between the stations (*Rogers*, 1984) leading to a strengthening Gulf Stream, higher temperatures in winter and spring in Scandinavia and the east coast of the United States, and other readily observed marked changes in the Northern Hemisphere. In the Mk3.0 model, like observations, swings in the prevailing flows often last several decades, and variations on a much longer time scale exist, as can be see in the longer model runs, for both control and warmed earth experiments. However, there is no sign of significantly different behaviour between experiments where forcings are constant (PIcntrl and Commit) or the SRES warming scenarios as shown in Figure 9, although only a rigorous analysis would confirm this. To make identification of each experiment easier, and to place focus on longer period variability, a moderate smoothing has been applied.

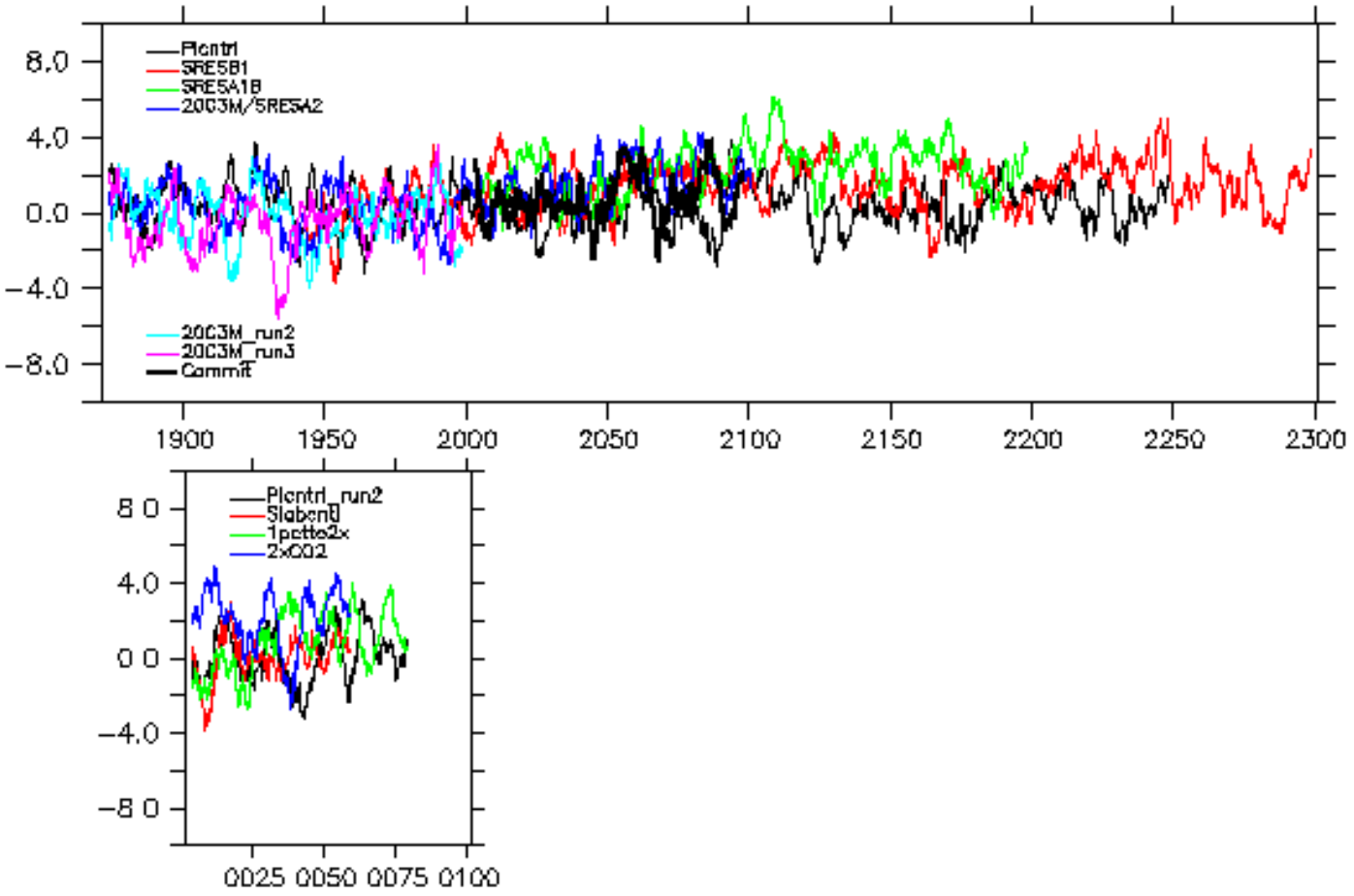


Figure 9: North Atlantic Oscillation Index for the a) control, 20^{th} Century and SRES experiments and b) CMIP and Slab Ocean experiments (°C). A 5-year running mean has been applied. See text for details.

8.0.5 Air-Sea Coupling Parameters.

Figures 10- 14 represent heat, water and momentum exchanges between the atmosphere to the ocean averaged over the 1961-1990 period, comparing the Mk3.0 climate model ensemble average[9] with the ERA40 (*Kallberg*, 2005) reanlalysis[10]. For climate modelling purposes it is important to have a realistic representation of these exchanges because any errors will be persistent and may grow unacceptably from the wide range of time and space scales that the ocean and atmosphere respond over. This is particularly true in the case of the Mk3.0 climate model where no corrections are made to keep the ocean and atmosphere on-track with the observed climate, a much more desirable air-sea boundary condition when compared to the Mk2.0 model *O'Farrell* (1998). These fluxes drive the coupling between the ocean component and the atmosphere/sea-ice/cryosphere/land-biosphere component in the Mk3.0 climate model. In general there is strong resemblance between the climate model and reanalysis results with some notable differences. The net heat flux (see Equation 1) shown in Figure 10 looks quite similar between model and reanalysis, with the major sources of positive heat into the oceans found in the Tropics and Eastern Boundary continents of the major ocean basins, and strong regions of heat loss from the oceans in the Western Boundary currents and high latitudes. The downward shortwave component, IPCC parameters $rsds$, shown in Figure 11 is often treated separately in climate models as specific parameterisations are formulated for uptake of

[9] an average of the 1^{st}, 2^{nd} and 3^{rd} ensembles from the 20C3M experiment have been used

[10] The reanalysis products are best considered as a NWP model product with observational assimilation built in. The output from the reanalysis bears strong resemblance of the observed parameters, however, they in general have a smoother and more continuous model aspect to them

heat in the upper ocean water column, however, it is included in the net heat flux shown in Figure 10 as the net heat is more familiar to climate scientists than net heat minus the downward shortwave radiation that is needed by the model. There are marked differences in the Subtropics where cloud and radiation interactions are inadequately treated in coarse resolution climate models. There is also an indication of a strong ITCZ in the climate model when compared with the reanalysis. Mid to high latitude character is quite similar between model and reanalysis. The net freshwater flux (described by Equation 2) in Figure 12 exhibit distinct differences between the model and reanalysis because of the complex and not fully understood relationships that exist between water of vastly different sources at the land-sea interface. However, the general freshwater deficit over the subtropics, large positive contributions in the Tropics, and the small positive contributions from mid to high latitudes are common between model and reanalysis. The sea-surface temperature cold-tongue bias in the tropics has a clear impact on the freshwater flux structure there, and the noisy contribution from rivers is evident where they contribute freshwater to the sea. There is close correspondence for the wind stress components between the climate model and reanalysis results as shown in Figures 13 and 14, particularly for the zonal wind stress parameter *tauu* which generally dominates the total momentum exchange between the ocean and atmosphere. Most noticeable differences for the meridional wind stress parameter *tauv* are in mid-latitudes and along the Equator where variability and magnitude are often high.

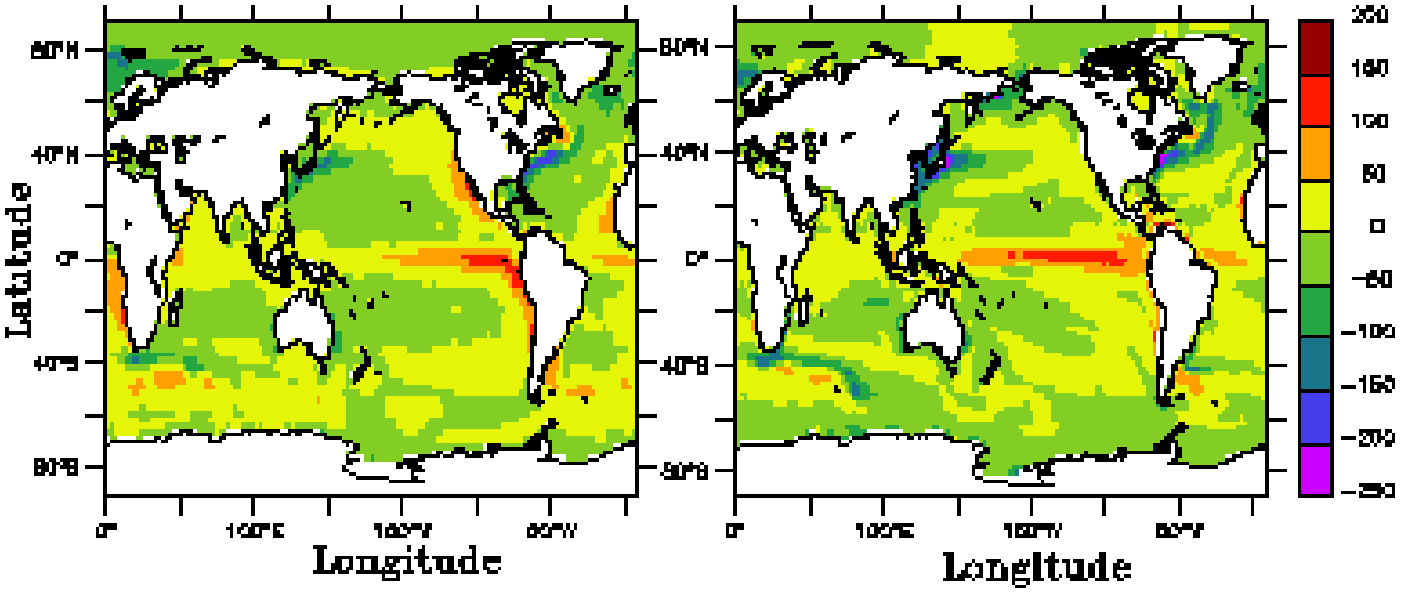


Figure 10: ERA40 reanalysis (left) and Mk3.0 climate model ensemble average (right) annually averaged net heat flux at the surface for 1961-1990 (W m^{-2}).

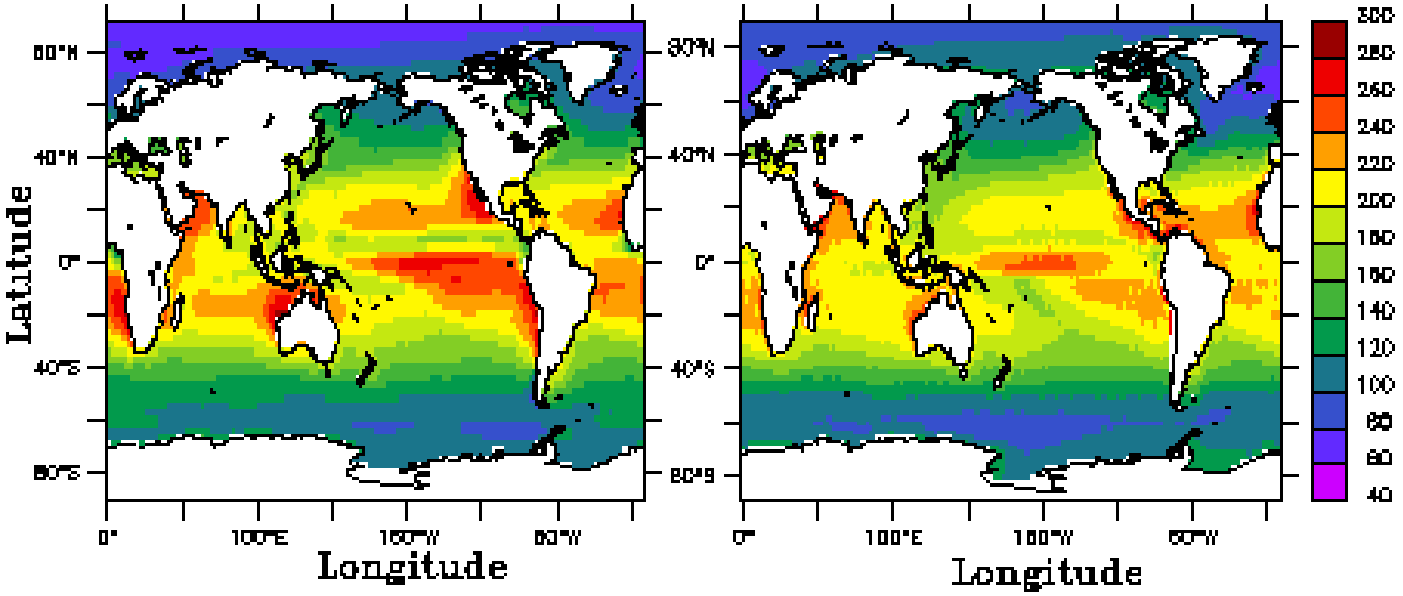


Figure 11: ERA40 reanalysis (left) and Mk3.0 climate model ensemble average (right) annually averaged ocean downward short-wave radiation at the surface for 1961-1990 (W m^{-2}).

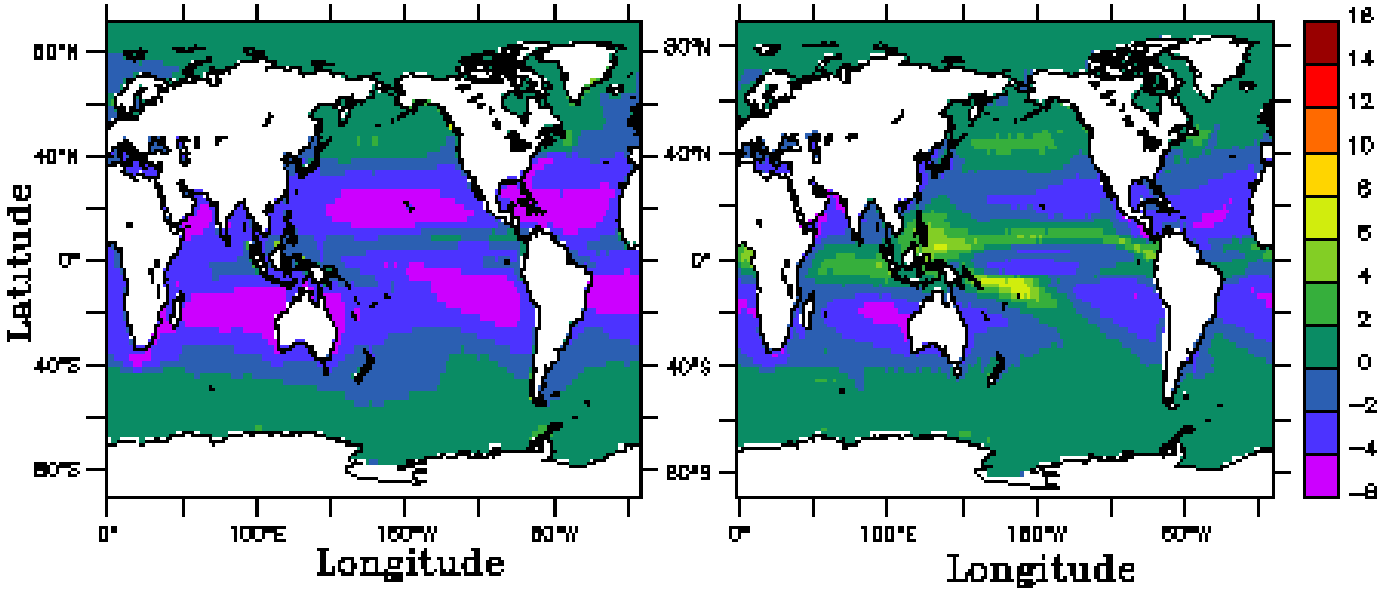


Figure 12: ERA40 reanalysis (left) and Mk3.0 climate model ensemble average (right) annually averaged ocean net freshwater flux at the surface for 1961-1990 (W m^{-2}).

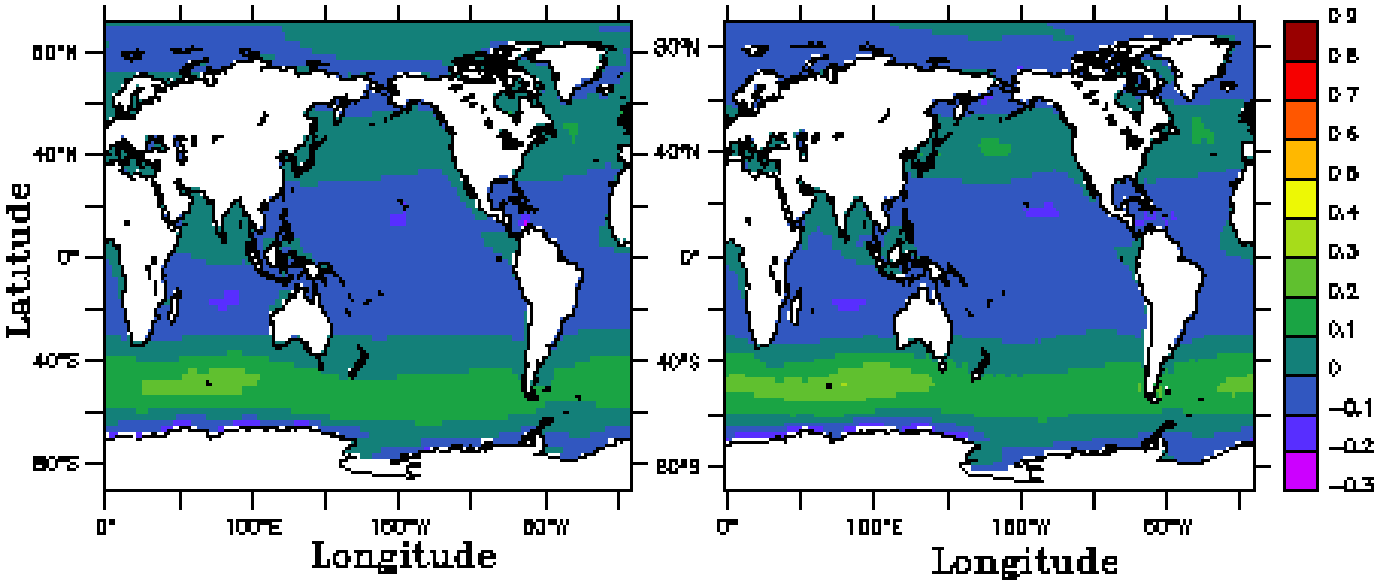


Figure 13: ERA40 reanalysis (left) and Mk3.0 climate model ensemble average (right) annually averaged ocean eastward surface wind stress for 1961-1990 (W m^{-2}).

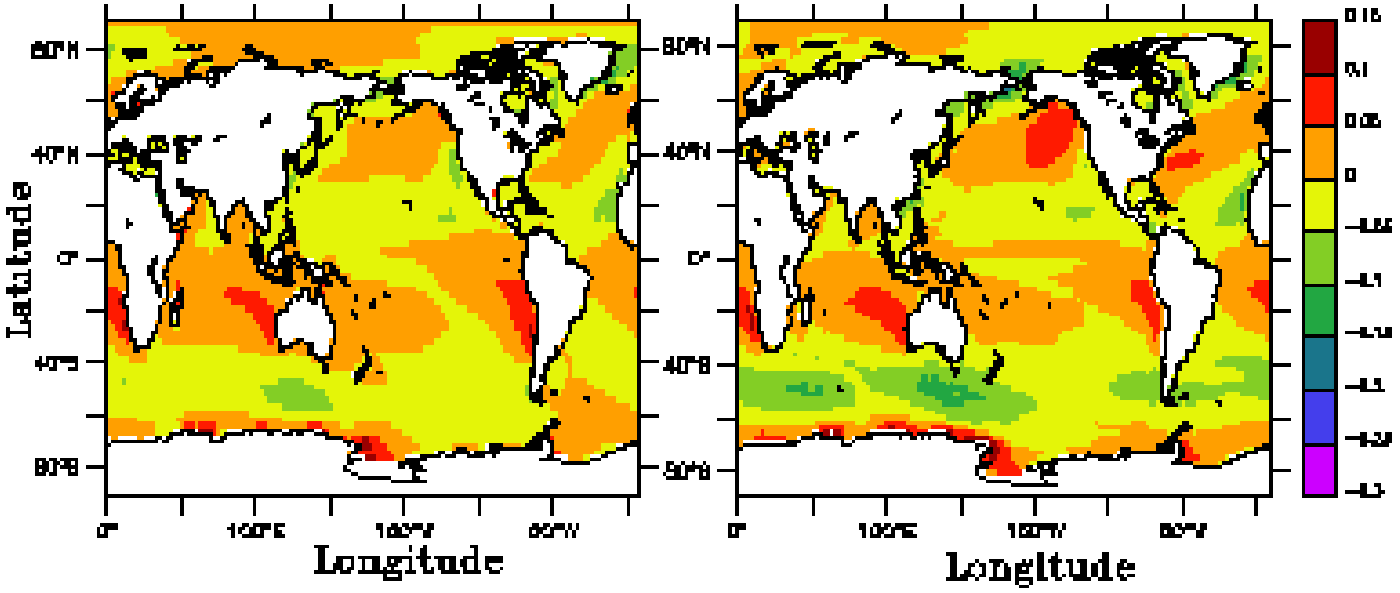


Figure 14: ERA40 reanalysis (left) and Mk3.0 climate model ensemble average (right) annually averaged ocean northward surface wind stress for 1961-1990 (W m^{-2}).

9 Derived Parameters and Processing Scripts

A number of IPCC parameters (files) have been derived and written either directly to files in the local IPCC data archive or into the model directories where the raw model outputs reside, these are all shown in Table 19. See Section 7.1.1 for a description of one and two step processing of IPCC parameters. For the two step processing of IPCC parameters it is necessary to execute the appropriate scripts under the account(s) where the original experiments were performed (see Table 2 for the location for each), as write directory permission is required. In addition, further

processing is required to put them into a form which is suitable for the local IPCC data archive, through one of the standard (monthly, daily, 2 and 3D parameter Tcl-Nap scripts and functions) processing scripts, or a component of the two-step processing scripts. From Table 19 one and two step IPCC parameters are readily identified from the column headed "Output Location". The value "IPCC" means put into the IPCC directory structure described in Section 4, and are thus final output files. The value "AGCM/netcdf.N.dir" means put the intermediate files (all monthly parameters) into the standard directory structure where the original experiments were performed (N is generally a one or two digit integer). As previously described, a further processing step will be required to convert them into the standard IPCC filename and place them into the standard directory structure. Mk3.0 parameters that have a range in their name of the structure {m-n} refer to a list with the integer part stepped in increments of one, beginning at m and finishing at n. For example, t{01-17} is the list t01, t02, t03, t04, t05, t06, t07, t08, t09, t10, t11, t12, t13, t14, t15, t16, t17 and t18, referring the air temperature at the 18 sigma levels listed in Table 11.1.4 used in defining the IPCC parameter *zg*. Another example, qf{01-12}av1, is the list qf01av1, qf02av2, qf03av3, qf04av1, qf05av1, qf06av1, qf07av1, qf08av1, qf09av1, qf10av1, qf11av1 and qf12av1, referring to the 12 monthly averages of heat fluxes[11] used in defining the IPCC parameter *qflux*.

[11] In this case the input files are "cif", referring to the "Melbourne University" Fortran binary file, with a standardised header and data structure once used frequently in CSIRO.

IPCC Parameter	Output Table	Tcl-Nap Script	Mk3.0 Parameter Dependencies	Output Location
huss	A1a	ipcc_calc_day2mon.tcl	q01	AGCM/netcdf.N.dir
mrros	A1a	ipcc_mrros.tcl	per, run	AGCM/netcdf.N.dir
ps	A1a	ipcc_calc_day2mon.tcl	psf	AGCM/netcdf.N.dir
rlus	A1a	ipcc_rlus.tcl	rgn,rgd	AGCM/netcdf.N.dir
rsus	A1a	ipcc_rsus.tcl	sgn,sgd	AGCM/netcdf.N.dir
rldscs	A1a	ipcc_rldscs.tcl	rgn,rgd,rgc	AGCM/netcdf.N.dir
rsdt	A1a	ipcc_rsdt.tcl	sint_t63.nc	IPCC
rtmt	A1a	ipcc_rtmt.tcl	rtu,sot,sit	AGCM/netcdf.N.dir
rsdscs	A1a	ipcc_rsdscs.tcl	sgn,sgc,sgd	AGCM/netcdf.N.dir
rsuscs	A1a	ipcc_rsuscs.tcl	sgn,sgc,sgd	AGCM/netcdf.N.dir
snc	A1a	ipcc_snc.tcl	snd,ico	AGCM/netcdf.N.dir
snw	A1a	ipcc_snw.tcl	snd,ico	AGCM/netcdf.N.dir
uas	A1a	ipcc_calc_day2mon.tcl	u01	AGCM/netcdf.N.dir
vas	A1a	ipcc_calc_day2mon.tcl	v01	AGCM/netcdf.N.dir
wap	A1c	ipcc_calc_day2mon_wap.tc	u{01-18},v{01-18},ps	AGCM/netcdf.N.dir
orog	A1b	ipcc_orog.tcl	histb file	IPCC
sftlf	A1b	ipcc_sftlf.tcl	Mk3_agcm.nc	IPCC
sftgif	A1b	ipcc_sftgif.tcl	Mk3_agcm.nc, glacier.txt	IPCC
zg	A1c	ipcc_zg.tcl	t{01-18},q{01-18},psl	AGCM/netcdf.N.dir
hfogo	O1a	ipcc_hfogo_ferret.tcl, ipcc_hfogo.tcl	not available	IPCC
stfmmc	O1b	ipcc_stfmmc_ferret.tcl, ipcc_stfmmc.tcl	not available	IPCC
zobt	O1d	ipcc_zobt.tcl	not available	IPCC
zos	O1c	ipcc_zos.tcl	not available	IPCC
qflux	O1d	ipcc_qflux.tcl	qf{01-12}av1.cif	IPCC
rhopoto	O1e	ipcc_rhopoto.tcl	thetao, so	IPCC
psf	not applicable	ipcc_psf.tcl	zht,psl,tsu	AGCM/netcdf.N.dir
cdd	A4	ipcc_cdd.tcl	pr	IPCC
etr	A4	ipcc_etr.tcl	tsl,tsh	IPCC
fd	A4	ipcc_fd.tcl	tsl	IPCC
gsl	A4	ipcc_gsl.tcl	tsu	IPCC
hwdi	A4	ipcc_hwdi.tcl	tsh	IPCC
r10	A4	ipcc_r10.tcl	pr	IPCC
r5d	A4	ipcc_r5d.tcl	pr	IPCC
r95t	A4	ipcc_r95t.tcl	pr	IPCC
sdii	A4	ipcc_sdii.tcl	pr	IPCC
tn90	A4	ipcc_tn90.tcl	tsl	IPCC

Table 19: Tcl-Nap scripts that are used in deriving IPCC parameters and their Mk3.0 Parameter dependencies.

10 Raw Model Parameter Resolutions by Experiment

To provide a visual indication of grid point data density, checkerboard Figures 15- 21 have been generated for Mk3.0 input[12] and output[13] atmosphere and ocean grids. These provide a handy reference for analysts who must know the grid coverage of output data and the input data that generates them. This can change from experiment to experiment, as detailed in Tables 20- 24. Figures 15, 16 and 17 display the three kinds of input horizontal atmospheric grids. Note that the grids shown in Figures 15 and 16 are not point averages of the surrounding 3×3 and 2×2 grids respectively, but rather reduced grids where a thinning out of the 9 or 4 points, respectively, are performed[14]. Figures 18 and 19 display input and output vertical atmospheric grids, respectively. Representative figures of Mk3.0 raw data and IPCC data (only parameter *cl*) on sigma-levels are not shown here. The ocean model horizontal and vertical grids are represented by Figures 20 and 21, respectively, where both input and output grids are identical.

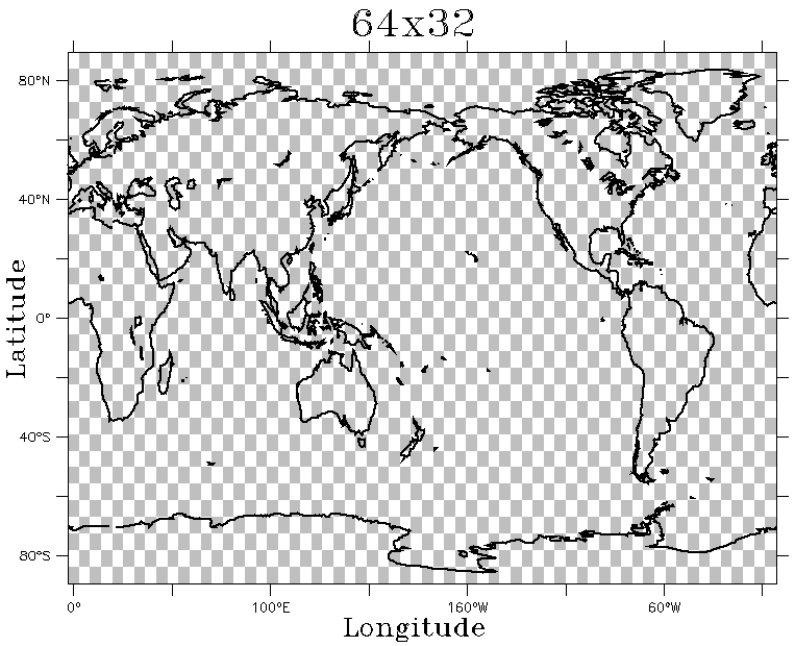


Figure 15: Mk3.0 Input (Raw) horizontal (64 longitudes by 32 latitudes) atmospheric grid.

[12]By Input it is meant the raw data that is written directly out by the Mk3.0 model before any processing is performed.

[13]By output it is meant the raw data that is processed and contributed to the IPCC activity.

[14]This was done primarily for the reason of reducing the size of data files where disk and tape space availability at the time were limited

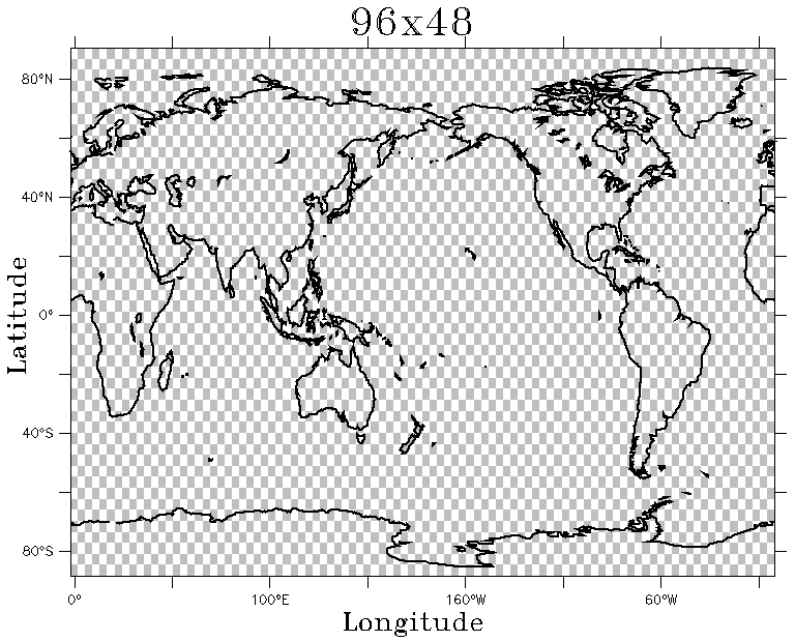


Figure 16: Mk3.0 Input (Raw) horizontal (96 longitudes by 48 latitudes) atmospheric grid.

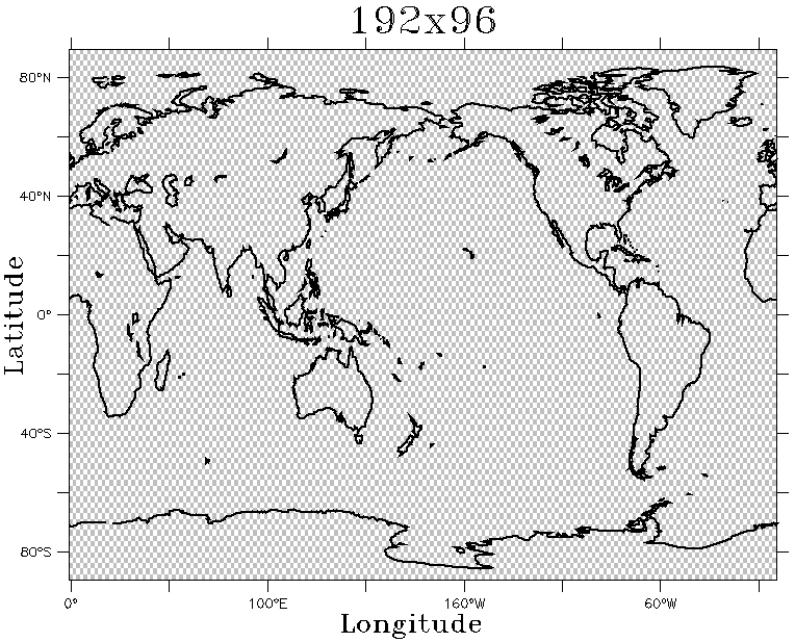


Figure 17: Mk3.0 Input (Raw) horizontal (192 longitudes by 96 latitudes) atmospheric grid.

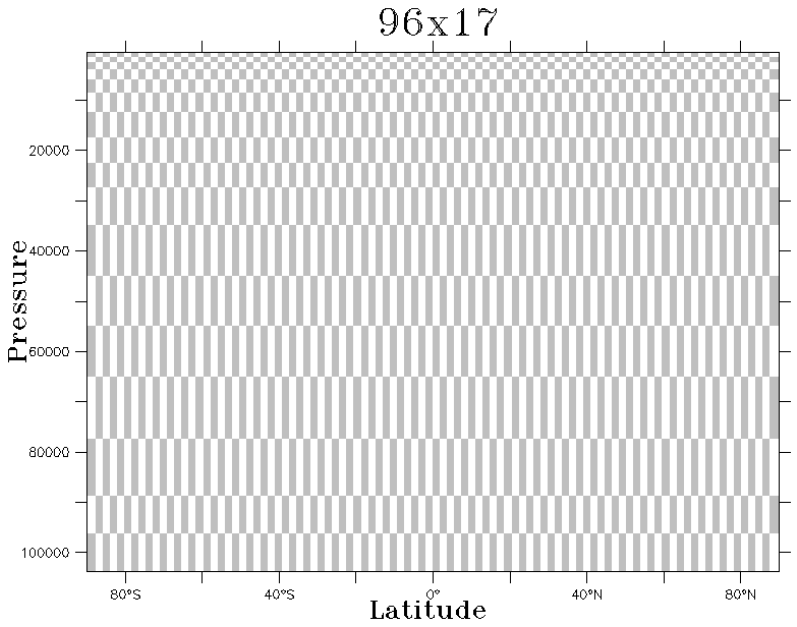


Figure 18: Mk3.0 Input (Raw) vertical (96 latitudes by 17 pressure levels) atmospheric grid.

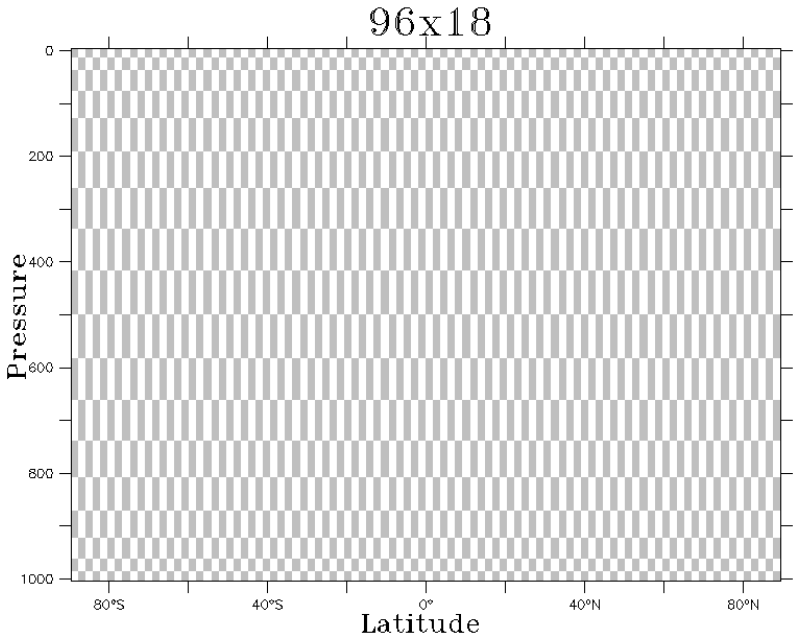


Figure 19: Mk3.0 Output (IPCC) vertical (96 latitudes by 18 pressure levels) atmospheric grid.

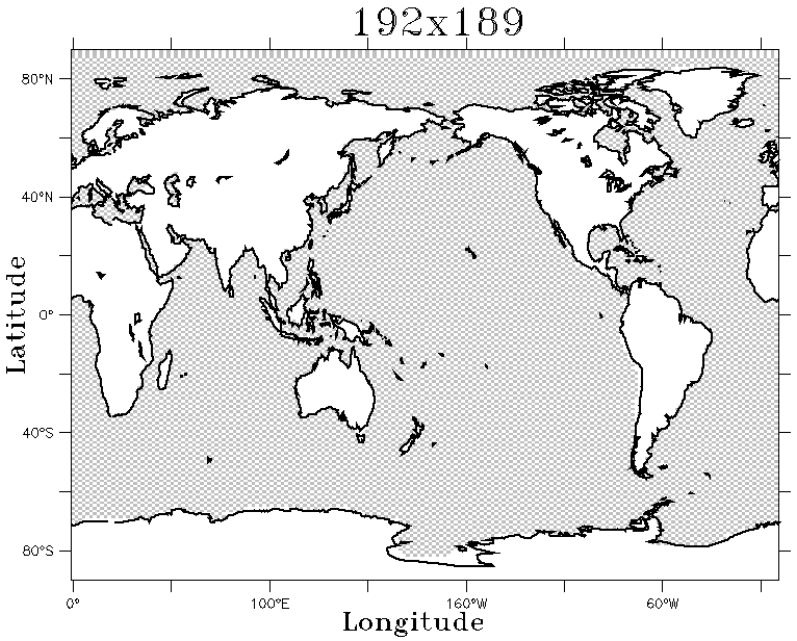


Figure 20: Mk3.0 Input (Raw)/Output (IPCC) horizontal (192 longitudes by 96 latitudes) ocean grid.

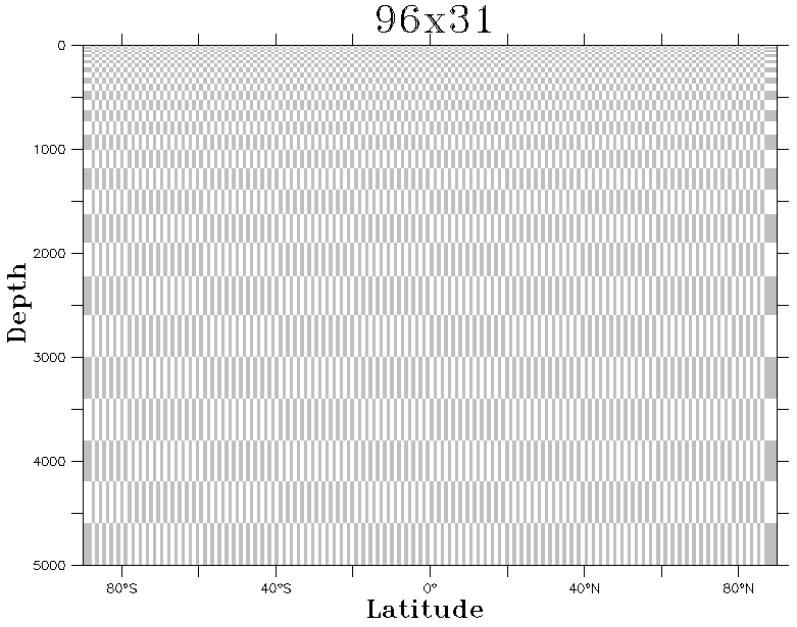


Figure 21: Mk3.0 Input (Raw)/Output (IPCC) vertical (96 latitudes by 31 depth levels) ocean grid.

In Tables 20 and 21 Mk3.0 raw parameters are shown in the left most column which have been used in defining monthly 2D and 3D (and in some cases time invariant) IPCC parameters defined in this report. In addition to the lists described in Table 19 w{0010-1000} is the list w1000, w0925, w0850, w0700, w0600, w0500, w0400, w0300, w0250, w0200, w0150, w0100, w0070, w0050, w0030, w0020 and w0010, referring the vertical velocity at the 17 pressure levels listed in Table 11.1.3 used

in defining the IPCC parameter *wap*[15]. The Mk3.0 raw parameters can be identified in Tables 20-24 can be found in Tables 5- 18 in the position under "Mk3.0 Parameter Name". These parameter names are used to select the desired IPCC output in one of the many processing scripts available (see Section 7.1.1) as a one-step process. However, if the value of "Mk3.0 Parameter Name" is an IPCC parameter name then it is the second step of a two-step processing task. In Tables 20- 24 other unambiguous values are used to select the desired IPCC output. An example of this would be *q4d* which can be used to select either monthly or daily 3D specific humidities as shown in Tables 9 and 15, respectively.

[15] And similarly for z1000...z0010 for IPCC parameter *zg*.

Mk3.0 Experiment Name							
Mk3.0 Parameter	Units	Horizontal Resolution $x \times y\{\times z\}$					
		a1b	cm3	ct3	m20	n20	o20
c{01-17}	1	192×96×17	192×96×17	192×96×17	64×32×17	192×96×17	192×96×17
cld	fraction	192×96	192×96	192×96	64×32	192×96	192×96
evp	mm day^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
hus	1	192×96	192×96	192×96	192×96	192×96	192×96
hfl	W m^{-2}	192×96	192×96	192×96	192×96	192×96	192×96
ico	fraction	192×96	192×96	192×96	192×96	192×96	192×96
icd	m	192×96	192×96	192×96	192×96	192×96	192×96
icu	m s^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
icv	m s^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
lwp	kg m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
iwp	m s^{-1}	192×96	192×96	192×96	64×32	192×96	192×96
psl	hPa	192×96	192×96	192×96	192×96	192×96	192×96
ps	hPa	192×96	192×96	192×96	192×96	192×96	192×96
pwc	mm	192×96	192×96	192×96	64×32	192×96	192×96
q{01-17}	1	192×96×17	192×96×17	192×96×17	64×32×17	192×96×17	192×96×17
rc2	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
rf2	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
rgc	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
rgd	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
rgn	W m^{-2}	192×96	192×96	192×96	192×96	192×96	192×96
rnc	mm day^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
rnd	mm day^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
run	mm day^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
rtc	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
rtu	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
sc2	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
sf2	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
sgc	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
sgd	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
sgn	W m^{-2}	192×96	192×96	192×96	192×96	192×96	192×96
sit	W m^{-2}	192×96	192×96	192×96	192×96	192×96	192×96
snd	m	192×96	192×96	192×96	192×96	192×96	192×96
sno	mm day^{-1}	192×96	192×96	192×96	64×32	192×96	192×96
sot	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
soc	W m^{-2}	192×96	192×96	192×96	64×32	192×96	192×96
per	mm day^{-1}	192×96	192×96	192×96	192×96	192×96	192×96
tsc	K	192×96	192×96	192×96	192×96	192×96	192×96
tax	N m^{-2}	192×96	192×96	192×96	192×96	192×96	192×96
tay	N m^{-2}	192×96	192×96	192×96	192×96	192×96	192×96
t{01-17}	K	192×96	192×96	192×96	64×32	192×96	192×96
tsu	K	192×96	192×96	192×96	192×96	192×96	192×96
u01	m s^{-1}	192×96	192×96	192×96	64×32	192×96	192×96
uas	m s^{-1}	192×96	192×96	192×96	64×32	192×96	192×96
vas	m s^{-1}	192×96	192×96	192×96	64×32	192×96	192×96
v{01-17}	m s^{-1}	192×96×17	192×96×17	192×96×17	64×32×17	192×96×17	192×96×17
w{1000-0010}	m s^{-1}	192×96×17	192×96×17	192×96×17	192×96×17	192×96×17	192×96×17
z{1000-0010}	m	192×96	192×96	192×96	192×96	192×96	192×96

Table 20: Unit and Resolution of raw 2/3D monthly atmosphere Mk3.0 parameters.

Mk3.0 Experiment Name						
Mk3.0 Parameter	Units	Horizontal Resolution $x \times y \{\times z\}$				
		s20	st1	uk1	qv3	qv4
c{01-17}	1	192×96×17	192×96×17	64×32×17	192×96×17	192×96×17
cld	fraction	192×96	192×96	64×32	192×96	192×96
evp	mm day^{-1}	192×96	192×96	192×96	192×96	192×96
hus	1	192×96	192×96	192×96	192×96	192×96
hfl	W m^{-2}	192×96	192×96	192×96	192×96	192×96
ico	fraction	192×96	192×96	192×96	192×96	192×96
icd	m	192×96	192×96	192×96	192×96	192×96
icu	m s^{-1}	192×96	192×96	192×96	192×96	192×96
icv	m s^{-1}	192×96	192×96	192×96	192×96	192×96
lwp	kg m^{-2}	192×96	192×96	64×32	192×96	192×96
iwp	m s^{-1}	192×96	192×96	64×32	192×96	192×96
psl	hPa	192×96	192×96	192×96	192×96	192×96
ps	hPa	192×96	192×96	192×96	192×96	192×96
pwc	mm	192×96	192×96	64×32	192×96	192×96
q{01-17}	1	192×96×17	192×96×17	64×32×17	192×96×17	192×96×17
rc2	W m^{-2}	192×96	192×96	64×32	192×96	192×96
rf2	W m^{-2}	192×96	192×96	64×32	192×96	192×96
rgc	W m^{-2}	192×96	192×96	64×32	192×96	192×96
rgd	W m^{-2}	192×96	192×96	64×32	192×96	192×96
rgn	W m^{-2}	192×96	192×96	192×96	192×96	192×96
rnc	mm day^{-1}	192×96	192×96	192×96	192×96	192×96
rnd	mm day^{-1}	192×96	192×96	192×96	192×96	192×96
run	mm day^{-1}	192×96	192×96	192×96	192×96	192×96
rtc	W m^{-2}	192×96	192×96	64×32	192×96	192×96
rtu	W m^{-2}	192×96	192×96	64×32	192×96	192×96
sc2	W m^{-2}	192×96	192×96	64×32	192×96	192×96
sf2	W m^{-2}	192×96	192×96	64×32	192×96	192×96
sgc	W m^{-2}	192×96	192×96	64×32	192×96	192×96
sgd	W m^{-2}	192×96	192×96	64×32	192×96	192×96
sgn	W m^{-2}	192×96	192×96	192×96	192×96	192×96
sit	W m^{-2}	192×96	192×96	192×96	192×96	192×96
snd	m	192×96	192×96	192×96	192×96	192×96
sno	mm day^{-1}	192×96	192×96	192×96	192×96	192×96
sot	W m^{-2}	192×96	192×96	192×96	192×96	192×96
soc	W m^{-2}	192×96	192×96	64×32	192×96	192×96
per	mm day^{-1}	192×96	192×96	192×96	192×96	192×96
tsc	K	192×96	192×96	192×96	192×96	192×96
tax	N m^{-2}	192×96	192×96	192×96	192×96	192×96
tay	N m^{-2}	192×96	192×96	192×96	192×96	192×96
t{01-17}	K	192×96	192×96×17	64×32×17	192×96×17	192×96×17
tsu	K	192×96	192×96	192×96	192×96	192×96
u{01-17}	m s^{-1}	192×96	192×96×17	64×32×17	192×96×17	192×96×17
uas	m s^{-1}	192×96	192×96	64×32	192×96	192×96
vas	m s^{-1}	192×96	192×96	64×32	192×96	192×96
v{01-17}	m s^{-1}	192×96	192×96×17	64×32×17	192×96×17	192×96×17
w{0010-1000}	m s^{-1}	192×96	192×96×17	192×96×17	192×96×17	192×96×17
z{0010-1000}	m	192×96	192×96	192×96	192×96	192×96

Table 21: Unit and Resolution of raw 2/3D monthly atmosphere Mk3.0 parameters (cont'd).

			Mk3.0 Experiment Name					
			a1b	cm3	ct3	m20	n20	o20
Mk3.0 Parameter	Hist	Units	Resolution $x \times y\{\times z\} \times t$					
evp	a	mm day^{-1}	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
hfl	a	W m^{-2}	not available	192×96×1	not available	not available	192×96×1	192×96×1
psl	a	hPa	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
q{01-17}	b	1	96×48×17×2	192×96×17×2	192×96×17×2	96×48×17×2	192×96×17x4	192×96×17x4
rnd	a	mm day^{-1}	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
rgd	a	W m^{-2}	not available	192×96×1	not available	not available	192×96×1	192×96×1
rgn	a	W m^{-2}	not available	192×96×1	not available	not available	192×96×1	192×96×1
rtu	a	W m^{-2}	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
sgd	a	W m^{-2}	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
sgn	a	W m^{-2}	not available	192×96×1	not available	not available	192×96×1	192×96×1
t{01-17}	b	K	96×48×17×2	192×96×17×2	192×96×17×2	96×48×17×2	192×96×17x4	192×96×17x4
tsl	a	K	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
tsh	a	K	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
tsc	a	K	96×48×1	192×96×1	192×96×1	192×96×1	192×96×1	192×96×1
u{01-17}	b	m s^{-1}	96×48×17×2	192×96×17×2	192×96×17×2	96×48×17×2	192×96×17x4	192×96×17x4
v{01-17}	b	m s^{-1}	96×48×17×2	192×96×17×2	192×96×17×2	96×48×17×2	192×96×17x4	192×96×17x4
uas	a	m s^{-1}	not available	not available	not available	not available	not available	192×96×1
uas	a	m s^{-1}	not available	not available	not available	not available	not available	192×96×1

Table 22: Unit and Resolution of raw 2/3D daily atmosphere Mk3.0 parameters.

			Mk3.0 Experiment Name				
			s20	st1	uk1	qv3	qv4
Mk3.0 Parameter	Hist	Units	Resolution $x \times y\{\times z\} \times t$				
evp	a	mm day^{-1}	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
hfl	a	W m^{-2}	not available	not available	not available	192×96×1	192×96×1
psl	a	hPa	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
q{01-17}	b	1	96×48×17×2	96×48×17×2	96×48×17×2	192×96×17×2	192×96×17×2
rgd	a	W m^{-2}	not available	not available	not available	192×96×1	192×96×1
rgn	a	W m^{-2}	not available	not available	not available	192×96×1	192×96×1
rnd	a	mm day^{-1}	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
rtu	a	W m^{-2}	192×96×1	96×48×1	192×96×1	192×96×1	192×96×1
sgn	a	W m^{-2}	not available	not available	not available	192×96×1	192×96×1
sgd	a	W m^{-2}	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
t{01-17}	b	K	96×48×17×2	96×48×17×2	96×48×17×2	192×96×17×2	192×96×17×2
tsc	a	K	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
tsl	a	K	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
tsh	a	K	96×48×1	96×48×1	96×48×1	192×96×1	192×96×1
u{01-17}	b	m s^{-1}	96×48×17×2	96×48×17×2	96×48×17×2	192×96×17×2	192×96×17×2
v{01-17}	b	m s^{-1}	96×48×17×2	96×48×17×2	96×48×17×2	192×96×17×2	192×96×17×2
uas	a	m s^{-1}	not available	not available	not available	not available	not available
uas	a	m s^{-1}	not available	not available	not available	not available	not available

Table 23: Unit and Resolution of raw 2/3D daily atmosphere Mk3.0 parameters (cont'd).

Mk3.0 Parameter	Units	Conversion Factor	$x \times y \times z$
kmt	m	none	192×189×1
mask	fraction	none	192×189×1
pme	mm day^{-1}	$\times \frac{1}{86400}$	192×189×1
psi	Sv	$\times 10^6$	192×189×1
Salt	10^{-3}	none	192×189×31
tos	K	-273.15	192×189×1
Temp	K	-273.15	192×189×31
uvel	m s^{-1}	$\times 10^{-2}$	192×189×31
vvel	m s^{-1}	$\times 10^{-2}$	192×189×31
wvel	m s^{-1}	$\times 10^{-2}$	192×189×31

Table 24: Unit and Resolution of 2/3D monthly ocean Mk3.0 parameters.

	0	1	2	3	4	5	6	7	8	9
0	-88.572	-86.722	-84.861	-82.998	-81.135	-79.270	-77.405	-75.541	-73.676	-71.811
1	-69.946	-68.081	-66.215	-64.350	-62.485	-60.620	-58.755	-56.889	-55.024	-53.159
2	-51.294	-49.429	-47.563	-45.698	-43.833	-41.968	-40.103	-38.237	-36.372	-34.507
3	-32.641	-30.776	-28.911	-27.046	-25.180	-23.315	-21.450	-19.585	-17.719	-15.854
4	-13.989	-12.124	-10.258	-8.393	-6.528	-4.663	-2.797	-0.932	0.932	2.797
5	4.663	6.528	8.393	10.258	12.124	13.989	15.854	17.719	19.585	21.450
6	23.315	25.180	27.046	28.911	30.776	32.641	34.507	36.372	38.237	40.103
7	41.968	43.833	45.698	47.563	49.429	51.294	53.159	55.024	56.889	58.755
8	60.620	62.485	64.350	66.215	68.081	69.946	71.811	73.676	75.541	77.405
9	79.270	81.135	82.998	84.861	86.722	88.572				

	0	1	2	3	4	5	6	7	8	9
0	0.0	1.875	3.75	5.625	7.5	9.375	11.25	13.125	15.0	16.875
1	18.75	20.625	22.5	24.375	26.25	28.125	30.0	31.875	33.75	35.625
2	37.5	39.375	41.25	43.125	45.0	46.875	48.75	50.625	52.5	54.375
3	56.25	58.125	60.0	61.875	63.75	65.625	67.5	69.375	71.25	73.125
4	75.0	76.875	78.75	80.625	82.5	84.375	86.25	88.125	90.0	91.875
5	93.75	95.625	97.5	99.375	101.25	103.125	105.0	106.875	108.75	110.625
6	112.5	114.375	116.25	118.125	120.0	121.875	123.75	125.625	127.5	129.375
7	131.25	133.125	135.0	136.875	138.75	140.625	142.5	144.375	146.25	148.125
8	150.0	151.875	153.75	155.625	157.5	159.375	161.25	163.125	165.0	166.875
9	168.75	170.625	172.5	174.375	176.25	178.125	180.0	181.875	183.75	185.625
10	187.5	189.375	191.25	193.125	195.0	196.875	198.75	200.625	202.5	204.375
11	206.25	208.125	210.0	211.875	213.75	215.625	217.5	219.375	221.25	223.125
12	225.0	226.875	228.75	230.625	232.5	234.375	236.25	238.125	240.0	241.875
13	243.75	245.625	247.5	249.375	251.25	253.125	255.0	256.875	258.75	260.625
14	262.5	264.375	266.25	268.125	270.0	271.875	273.75	275.625	277.5	279.375
15	281.25	283.125	285.0	286.875	288.75	290.625	292.5	294.375	296.25	298.125
16	300.0	301.875	303.75	305.625	307.5	309.375	311.25	313.125	315.0	316.875
17	318.75	320.625	322.5	324.375	326.25	328.125	330.0	331.875	333.75	335.625
18	337.5	339.375	341.25	343.125	345.0	346.875	348.75	350.625	352.5	354.375
19	356.25	358.125								

11 Coupled Model Horizontal and Vertical Grids

11.1 Atmosphere Grid

11.1.1 Latitudes

11.1.2 Longitudes

11.1.3 Pressures

	0	1	2	3	4	5	6	7	8	9
0	1000.0	925.0	850.0	700.0	600.0	500.0	400.0	300.0	250.0	200.0
1	150.0	100.0	70.0	50.0	30.0	20.0	10.0			

	0	1	2	3	4	5	6	7	8	9
0	0.995542	0.978395	0.945816	0.899863	0.842593	0.776063	0.702332	0.623457	0.541495	0.458505
1	0.376543	0.297668	0.223937	0.157407	0.100137	0.0541838	0.0216049	0.00445816		

11.1.4 Sigma Levels

	0	1	2	3	4	5	6	7	8	9
0	-88.195	-87.239	-86.297	-85.358	-84.422	-83.4870	-82.552	-81.618	-80.684	-79.750
1	-78.817	-77.883	-76.950	-76.017	-75.084	-74.151	-73.218	-72.285	-71.352	-70.419
2	-69.486	-68.553	-67.620	-66.688	-65.755	-64.822	-63.889	-62.956	-62.024	-61.091
3	-60.158	-59.225	-58.293	-57.360	-56.427	-55.494	-54.562	-53.629	-52.696	-51.763
4	-50.831	-49.89	-48.965	-48.033	-47.100	-46.167	-45.234	-44.302	-43.369	-42.436
5	-41.504	-40.571	-39.638	-38.706	-37.7733	-36.840	-35.908	-34.975	-34.042	-33.109
6	-32.177	-31.244	-30.311	-29.379	-28.446	-27.513	-26.581	-25.648	-24.715	-23.7831
7	-22.850	-21.917	-20.985	-20.052	-19.119	-18.187	-17.254	-16.321	-15.389	-14.456
8	-13.523	-12.591	-11.658	-10.725	-9.793	-8.860	-7.927	-6.995	-6.062	-5.129
9	-4.197	-3.264	-2.331	-1.399	-0.466	0.466	1.399	2.331	3.264	4.197
10	5.129	6.062	6.995	7.927	8.860	9.793 1	0.725	11.658	12.591	13.523
11	14.456	15.389	16.321	17.254	18.187	19.119	20.052	20.985	21.917	22.850
12	23.783	24.715	25.648	26.581	27.513	28.446	29.379	30.311	31.244	32.177
13	33.109	34.042	34.975	35.908	36.840	37.773	38.706	39.638	40.571	41.504
14	42.436	43.369	44.302	45.234	46.167	47.100	48.033	48.965	49.898	50.831
15	51.763	52.696	53.629	54.562	55.494	56.427	57.360	58.293	59.225	60.158
16	61.091	62.024	62.956	63.889	64.822	65.755	66.688	67.620 6	8.553	69.486
17	70.419	71.352	72.285	73.218	74.151	75.084	76.017	76.950	77.883	78.817
18	79.750	80.684	81.6182	82.552	83.487	84.422	85.358	86.297	87.239	

11.2 Ocean Tracer Grid

All Mk3.0 experiments were run with an ocean model having an erroneously slightly shifted latitude grid[16], these are shown in Tables 11.4.1 and 11.4.2 for reference. Various solutions to this problem were considered, however, we chose an option where all the ocean model data would remain internally consistent and needing no interpolation or subsequent loss of ocean data. This was achieved by simply shifting the ocean model latitudes so that they line up with the atmospheric land sea mask. As the ocean grid is twice that of the atmosphere in the north-south direction this error is rather small (less than one degree) however needed to be corrected in the IPCC files documented here. This means that the ocean model sea-surface temperature would be slightly out of line with the atmospheric model surface temperature. The correct values are listed in Tables 11.2.1 and 11.3.1.

11.2.1 Latitudes

11.2.2 Longitudes

11.3 Ocean Velocity Grid

11.3.1 Latitudes

11.3.2 Longitudes

11.3.3 Depths

[16] This problem was fixed in the Mk3.1 model and subsequent versions.

	0	1	2	3	4	5	6	7	8	9
0	0.0	1.875	3.75	5.625	7.5	9.375	11.25	13.125	15.0	16.875
1	18.75	20.625	22.5	24.375	26.25	28.125	30.0	31.875	33.75	35.625
2	37.5	39.375	41.25	43.125	45.0	46.875	48.75	50.625	52.5	54.375
3	56.25	58.125	60.0	61.875	63.75	65.625	67.5	69.375	71.25	73.125
4	75.0	76.875	78.75	80.625	82.5	84.375	86.25	88.125	90.0	91.875
5	93.75	95.625	97.5	99.375	101.25	103.125	105.0	106.875	108.75	110.625
6	112.5	114.375	116.25	118.125	120.0	121.875	123.75	125.625	127.5	129.375
7	131.25	133.125	135.0	136.875	138.75	140.625	142.5	144.375	146.25	148.125
8	150.0	151.875	153.75	155.625	157.5	159.375	161.25	163.125	165.0	166.875
9	168.75	170.625	172.5	174.375	176.25	178.125	180.0	181.875	183.75	185.625
10	187.5	189.375	191.25	193.125	195.0	196.875	198.75	200.625	202.5	204.375
11	206.25	208.125	210.0	211.875	213.75	215.625	217.5	219.375	221.25	223.125
12	225.0	226.875	228.75	230.625	232.5	234.375	236.25	238.125	240.0	241.875
13	243.75	245.625	247.5	249.375	251.25	253.125	255.0	256.875	258.75	260.625
14	262.5	264.375	266.25	268.125	270.0	271.875	273.75	275.625	277.5	279.375
15	281.25	283.125	285.0	286.875	288.75	290.625	292.5	294.375	296.25	298.125
16	300.0	301.875	303.75	305.625	307.5	309.375	311.25	313.125	315.0	316.875
17	318.75	320.625	322.5	324.375	326.25	328.125	330.0	331.875	333.75	335.625
18	337.5	339.375	341.25	343.125	345.0	346.875	348.75	350.625	352.5	354.375
19	356.25	358.125								

	0	1	2	3	4	5	6	7	8	9
0	-87.712	-86.766	-85.827	-84.890	-83.954	-83.019	-82.085	-81.151	-80.217	-79.283
1	-78.350	-77.417	-76.484	-75.550	-74.617	-73.684	-72.751	-71.818	-70.885	-69.953
2	-69.020	-68.087	-67.154	-66.221	-65.288	-64.356	-63.423	-62.490	-61.557	-60.6248
3	-59.692	-58.759	-57.826	-56.893	-55.961	-55.028	-54.095	-53.162	-52.230	-51.297
4	-50.364	-49.432	-48.499	-47.566	-46.633	-45.701	-44.768	-43.835	-42.9031	-41.970
5	-41.037	-40.105	-39.172	-38.239	-37.3070	-36.374	-35.441	-34.508	-33.576	-32.643
6	-31.710	-30.778	-29.845	-28.912	-27.980	-27.047	-26.114	-25.182	-24.249	-23.316
7	-22.384	-21.451	-20.518	-19.586	-18.6534	-17.720	-16.788	-15.8554	-14.922	-13.990
8	-13.057	-12.124	-11.192	-10.259	-9.326	-8.394	-7.4613	-6.528	-5.596	-4.663
9	-3.7306	-2.798	-1.865	-0.932	0.0	0.932	1.8653	2.798	3.730	4.663
10	5.596	6.528	7.461	8.394	9.326	10.2593	11.192	12.124	13.057	13.990
11	14.922	15.855	16.788	17.720	18.653	19.586	20.518	21.451	22.384	23.316
12	24.2495	25.182	26.114	27.047	27.980	28.912	29.845	30.778	31.710	32.643
13	33.576	34.508	35.441	36.37	37.307	38.239	39.172	40.105	41.037	41.970
14	42.903	43.835	44.7685	45.701	46.633	47.566	48.499	49.432	50.364	51.297
15	52.230	53.162	54.095	55.028	55.961	56.893	57.826	58.759	59.692	60.624
16	61.557	62.490	63.423	64.356	65.288	66.221	67.154	68.087	69.020	69.953
17	70.885	71.818	72.751	73.684	74.617	75.550	76.484	77.417	78.350	79.283
18	80.217	81.151	82.085	83.019	83.954	84.890	85.827	86.766	87.7126	

	0	1	2	3	4	5	6	7	8	9
0	0.937	2.812	4.687	6.562	8.437	10.312	12.187	14.062	15.937	17.812
1	19.687	21.562	23.437	25.312	27.187	29.062	30.937	32.812	34.687	36.562
2	38.437	40.312	42.187	44.062	45.937	47.812	49.687	51.562	53.437	55.312
3	57.187	59.062	60.937	62.812	64.687	66.562	68.437	70.312	72.187	74.062
4	75.937	77.812	79.687	81.562	83.437	85.312	87.187	89.062	90.937	92.812
5	94.687	96.562	98.437	100.312	102.187	104.062	105.937	107.812	109.687	111.562
6	113.437	115.312	117.187	119.062	120.937	122.812	124.687	126.562	128.437	130.312
7	132.187	134.062	135.937	137.812	139.687	141.562	143.437	145.312	147.187	149.062
8	150.937	152.812	154.687	156.562	158.437	160.312	162.187	164.062	165.937	167.812
9	169.687	171.562	173.437	175.312	177.187	179.062	180.937	182.812	184.687	186.562
10	188.437	190.312	192.187	194.062	195.937	197.812	199.687	201.562	203.437	205.312
11	207.187	209.062	210.937	212.812	214.687	216.562	218.437	220.312	222.187	224.062
12	225.937	227.812	229.687	231.562	233.437	235.312	237.187	239.062	240.937	242.812
13	244.687	246.562	248.437	250.312	252.187	254.062	255.937	257.812	259.687	261.562
14	263.437	265.312	267.187	269.062	270.937	272.812	274.687	276.562	278.437	280.312
15	282.187	284.062	285.937	287.812	289.687	291.562	293.437	295.312	297.187	299.062
16	300.937	302.812	304.687	306.562	308.437	310.312	312.187	314.062	315.937	317.812
17	319.687	321.562	323.437	325.312	327.187	329.062	330.937	332.812	334.687	336.562
18	338.437	340.312	342.187	344.062	345.937	347.812	349.687	351.562	353.437	355.312
19	357.187	359.062								

	0	1	2	3	4	5	6	7	8	9
0	5.0	15.0	28.248	42.024	59.662	78.541	102.110	127.883	159.466	194.555
1	236.968	284.645	341.690	406.377	483.194	570.865	674.860	793.761	934.072	1095.214
2	1284.648	1502.909	1758.774	2054.291	2400.0	2800.0	3200.0	3600.0	4000.0	4400.0
3	4800.0									

	0	1	2	3	4	5	6	7	8	9
0	-87.186	-86.258	-85.327	-84.396	-83.464	-82.533	-81.601	-80.668	-79.736	-78.804
1	-77.872	-76.939	-76.007	-75.075	-74.142	-73.209	-72.277	-71.345	-70.412	-69.479
2	-68.546	-67.615	-66.682	-65.749	-64.816	-63.884	-62.952	-62.019	-61.086	-60.154
3	-59.221	-58.288	-57.356	-56.423	-55.491	-54.558	-53.625	-52.693	-51.760	-50.827
4	-49.895	-48.963	-48.030	-47.097	-46.164	-45.23	-44.300	-43.366	-42.434	-41.502
5	-40.569	-39.636	-38.703	-37.771	-36.839	-35.906	-34.973	-34.041	-33.108	-32.175
6	-31.242	-30.310	-29.377	-28.445	-27.512	-26.579	-25.647	-24.714	-23.782	-22.849
7	-21.916	-20.984	-20.051	-19.118	-18.186	-17.253	-16.321	-15.388	-14.455	-13.523
8	-12.590	-11.657	-10.725	-9.792	-8.859	-7.927	-6.994	-6.062	-5.129	-4.196
9	-3.264	-2.331	-1.398	-0.466	0.466	1.398	2.331	3.264	4.196	5.129
10	6.062	6.994	7.927	8.859	9.792	10.725	11.657	12.590	13.523	14.455
11	15.388	16.321	17.253	18.186	19.118	20.051	20.984	21.916	22.849	23.782
12	24.714	25.647	26.579	27.512	28.445	29.377	30.310	31.242	32.175	33.108
13	34.041	34.973	35.906	36.839	37.771	38.703	39.636	40.569	41.502	42.434
14	43.366	44.300	45.232	46.164	47.097	48.030	48.963	49.895	50.827	51.760
15	52.693	53.625	54.558	55.491	56.423	57.356	58.288	59.221	60.154	61.086
16	62.019	62.952	63.884	64.816	65.749	66.68	67.615	68.546	69.479	70.412
17	71.3452	72.277	3.209	74.142	75.075	76.007	76.939	77.872	78.804	79.736
18	80.668 81.601	82.533	83.464	84.396	85.327	86.258	87.186	88.111		

11.4 Erroneous Latitudes

11.4.1 Ocean Tracer Grid

11.4.2 Ocean Velocity Grid

	0	1	2	3	4	5	6	7	8	9
0	-86.722	-85.793	-84.862	-83.930	-82.998	-82.067	-81.135	-80.202	-79.270	-78.338
1	-77.405	-76.472	-75.541	-74.609	-73.676	-72.742	-71.811	-70.879	-69.946	-69.012
2	-68.081	-67.149	-66.215	-65.2827	-64.350	-63.418	-62.485	-61.552	-60.620	-59.688
3	-58.7552	-57.8221	-56.89	-55.957	-55.024	-54.091	-53.159	-52.227	-51.294	-50.361
4	-49.4292	-48.497	-47.563	-46.630	-45.698	-44.766	-43.833	-42.900	-41.968	-41.036
5	-40.103	-39.1699	-38.237	-37.305	-36.372	-35.439	-34.507	-33.574	-32.642	-31.709
60	-30.776	-29.844	-28.911	-27.978	-27.046	-26.113	-25.181	-24.248	-23.315	-22.383
70	-21.450	-20.517	-19.585	-18.652	-17.72	-16.787	-15.854	-14.921	-13.989	-13.056
80	-12.124	-11.191	-10.258	-9.326	-8.393	-7.461	-6.528	-5.595	-4.663	-3.730
9	-2.797	-1.865	-0.932	0.0	0.932	1.865	2.797	3.730	4.663	5.595
10	6.528	7.461	8.393	9.326	10.258	11.191	12.124	13.056	13.989	14.921
11	15.854	16.787	17.72	18.652	19.585	20.517	21.450	22.383	23.315	24.248
12	25.181	26.113	27.046	27.978	28.911	29.844	30.776	31.709	32.642	33.574
13	34.507	35.439	36.372	37.305	38.237	39.169	40.103	41.036	41.9682	42.900
14	43.833	44.766	45.698	46.630	47.563	48.497	49.429	50.361	51.294	52.227
15	53.159	54.091	55.024	55.957	56.89	57.822	58.755	59.688	60.620	61.552
16	62.485	63.418	64.350	65.282	66.215	67.149	68.081	69.012	69.946	70.879
17	71.811	72.742	73.676	74.609	75.541	76.472	77.405	78.338	79.270	80.202
18	81.135	82.067	82.998	83.930	84.862	85.793	86.722	87.651	88.572	

12 Concluding Remarks

The successful delivery of CSIRO Mk3.0 model outupt to PCMDI marks an important milestone in numerical climate modelling in CSIRO and is recognised as one of the most important outcomes by the Division of Marine and Atmospheric Research in recent years. A discussion of the history of these efforts, including earlier versions of the CSIRO climate model, has been described by *Smith* (2007).

13 Acknowledgments

The Australian Greenhouse Office supported this work. The efforts of the Climate Modelling and Applications Team at CSIRO Atmospheric Research in developing the CSIRO Mk3.0 model are gratefully acknowledged, as is the support and facilities of the HPSC (http://www.hpsc.csiro.au). Harvey Davies Nap extension (http://sourceforge.net/projects/tcl-nap) of Tcl provided the ability to process raw model output into IPCC conforming netCDF files. FERRET was used to generate the figures in this document and is available free of charge through the Internet (http://ferret.wrc.noaa.gov/Ferret).

References

Brohan, P., Kennedy, J.J., Harris, I., Tett, S.F.B. and P.D. Jones. Uncertainty estimates in regional and global observed temperature changes: A new data set from 1850. *J. Geophys. Res.*, **111**, D12106, doi:10.1029/2005JD006548, 2006.

Cai, W.J., Collier, M.A., Gordon, H.B. and Waterman, L.J.. Strong ENSO variability and a super-ENSO pair in the CSIRO coupled climate model. *Mon. Wea. Rev.*, **131**, 1189–1210, 2002.

Cai, W. Collier, M.A., Durack, P.J., Gordon, H.B., Hirst, A.C. O'Farrell, S.P., Whetton, P.J.. The response of climate variability and mean state to climate change: preliminary results from the CSIRO Mark 3 coupled model. *CLIVAR Exchanges*, **28**, 8–11, 2003.

Cai, W.J., McPhaden, M.J. and Collier, M.A.. Multidecadal fluctuations of the relationship between equatorial Pacific heat content anomalies and ENSO amplitude. *Geophys. Res. Lett.*, **31**, L01201, doi:10.1029/2003GL018714, 2004
Cai, W., McPhaden, M. J., and Collier, M. A. (2004). Multidecadal fluctuations in the relationship between equatorial Pacific heat content anomalies and ENSO amplitude. Geophysical Research Letters , 31 (1).

Chen W.Y.. Assessment of Southern Oscillation Sea-Level Pressure Indices. *Mon. Wea. Rev.*, **110**, 800–807, 1982.

Collier, M.A.. The CSIRO NCEP reanalysis archive. CSIRO Atmospheric Research internal Paper No. 17. 193 pp., 2000.

Collier, M.A.. The CSIRO netCDF archive of the World Ocean Atlas 2001. CSIRO Atmospheric Research internal Paper No. 26. 20 pp., 2003.

Collier, M.A.. The CSIRO NCEP/NCAR/DOE *R-1/R-2* archive. CSIRO Atmospheric Research internal Paper No. 66. 76 pp., 2004.

Collier, M.A., A.C. Hirst, M.R. Dix, R.P. Cechet, H.B. Gordon, E.A. Kowalczyk, S.P. O'Farrell and L.D. Rotstayn. CMIP experimental results from the CSIRO Mk3 Climate System Model. CSIRO Atmospheric Research internal Paper No. 69. 73 pp., 2004.

Collier, M.A., M.R. Dix and A.C. Hirst. CSIRO Mk3 Climate System Model and Meeting the Strict IPCC AR4 Data Requirements. MODSIM07 International Congress on Modelling and Simulation : Land, water environmental management : integrated systems for sustainability :

Christchurch, 10-13 December [2007] : proceedings , Christchurch , N.Z., L. Oxley, D. Kulasiri, Modelling and Simulation Society of Australia and New, and Zealand. Christchurch: Modelling and Simulation Society of Australia and New Zealand. p. 582-588.

Conkright, M.E., R.A. Locarnini, H.E. Garcia, T.D. O'Brien, T.P. Boyer, C. Stephens and J.I. Antonov. World OCEAN ATLAS 2001: Objective Analyses, Data Statistics and Figures, CD-ROM Documentation. National Oceanographic Data Center, Silver Spring, MD, **17**, 17 pp., 2002.

Covey, C., AchutaRao, K.M. Cubasch, C. Ulrich, P. Jones, S.J. Lambert, M.E. Mann, T.J. Phillips and K.E. Taylor. An Overview of Results from the Coupled Model Intercomparison Project (CMIP). *Global and Planetary Change*, **37**, 103–133, 2003.

Davies, H.L.. Nap 6.3 User's Guide. Electronic edition only. CSIRO Marine and Atmospheric Research Paper No. 004., in press.

Dorman, J.L., and P.J. Sellers. A global climatology of albedo, roughness length and stomatal resistance for atmospheric general circulation models as represented by the simple biosphere model (SiB). *J. Appl. Meteor.*, **28**, 833–855, 1989.

Gordon, H.B. and S.P. OFarrell. Transient climate change in the CSIRO coupled model with dynamic sea-ice. *Mon. Wea. Rev.*, **125**, 875–907, 1997.

Gordon, H.B., L.D. Rotstayn, J.L. McGregor, M.R. Dix, E.A. Kowalczyk, S.P. O'Farrell, L.J. Waterman, A.C. Hirst, S.G. Wilson, M.A. Collier, I.G. Watterson, and T.I. Elliott. The CSIRO Mk3 Climate System Model. CSIRO Atmospheric Research technical paper No. 60. 130 pp., 2002.

Gregory, D. and P. Rowntree. A mass flux convection scheme with representation of cloud ensemble characteristics and stability dependent closure. *Mon. Wea. Rev.*, **118**, 1483–1506, 1990.

Griffies, S.M., Gnanadesikan, A., Pacanowski, R.C., Larichev, V.D, Dukowicz, J.K and R.D. Smith. Isoneutral diffusion in a z-coordinate ocean model. *J. Phys. Oceanogr.*, **28**, 805–830, 1998.

Griffies, S.M.. The Gent-McWilliams skew flux. *J. Phys. Oceanogr.*, **28**, 831–841, 1998.

Hirst, A.C., O'Farrell, S.P. and H.B. Gordon. Comparison of a coupled ocean-atmosphere model with and without oceanic eddy-induce advection. Part I: Ocean spinup and control integrations. *J. Climate*, **13**, 139–163, 2000.

Kallberg, P., P. Berrisford, B. Hoskins, A. Simmons, S. Uppala, S. Lamy-Thépaut and R. Hine. ERA-40 Atlas ERA-40 Project Report Series No. 19, 2005.

Kalnay, E., M. Kanamitsu, R. Kistler, W. Collins, D. Deaven, L. Gandin, M. Iredell, S. Saha, G. White, J. Woollen, Y. Zhu, M. Chelliah, W. Ebisuzaki, W. Higgins, J. Janowiak, K.C. Mo, C. Ropelewski, J. Wang, A. Leetmaa, R. Reynolds, R. Jenne and **D. Jospher** The NCEP/NCAR 40-Year Reanalysis Project. *Bul. Amer. Meteor. Soc.*, **3**, 437–471, 1996.

Kanamitsu, M., W. Ebisuzaki, J. Woollen, S-K. Yang, J.J. Hnilo, M. Fiorino and **G.L. Potter**. NCEP-DEO AMIP-II Reanalysis ($R-2$). *Bul. Amer. Meteor. Soc.*, **3**, 437–471, 2002.

Kistler, R., E. Kalnay, W. Collins, S. Saha, G. White, J. Woollen, M. Chelliah, W. Ebisuzaki, M. Kanamitsu, V. Kousky, H. van den Dool, R. Jenne and M. Fiorino. The NCEP-NCAR 50-year Reanalysis: Monthly Means CD-ROM and Documentation. *Bul. Amer. Meteor. Soc.*, **82**, 247–267, 2001.

Kowalczyk, E.A., Garratt, J.R. and P.B. Krummel. A soil-canopy scheme for use in a numerical model of the atmosphere – 1D stand alone model. CSIRO Division of Atmospheric Research Tech. Paper, **23**, 56 pp, 1991.

Kowalczyk, E.A., Garratt, J.R. and P.B. Krummel. Implementation of a soil-canopy scheme into the CSIRO GCM – Regional aspects of the model response. CSIRO Division of Atmospheric Research Tech. Paper, **32**, 59 pp, 1994.

McGregor, J.L., Gordon, H.B., Watterson, I.G., Dix, M.R., and L.D. Rotstayn The CSIRO 9-level atmospheric general circulation model. CSIRO Atmospheric Research technical paper No. 26. 89 pp., 1993.

McGregor J.L.. Economical determination of departure points for semi-Lagrangian models. *Mon. Wea. Rev.*, **121**, 221–230, 1993.

Meehl, G.A., C. Covey, M. Latif, B. McAvaney and R.J. Stouffer. CMIP: The Coupled Model Intercomparison Project. *CLIVAR Exchanges*, **26**, 1–4, 2003.

Mitchell, J.F.B., Davis, M.R., Ingram, W.J. and C.A. Senior. On surface temperature, greenhouse gases, and aerosols: Models and observations. *J. Climate*, **8**, 2364–2386, 1995.

O'Farrell, S.P.. Investigation of the dynamic sea ice component of a coupled atmosphere sea ice general circulation model. *Geophys. Res.*, **103**(C8), 15751–15782, 1998.

Rogers J.C.. The Association between the North Atlantic Oscillation and the Southern Oscillation in the Northern Hemisphere. *Mon. Wea. Rev.*, **112**, 1999–2015, 1984.

Rotstayn, L.D., Ryan, B.F. and J.J. Katzfey. A scheme for calculation of the liquid fraction in mixed-phase stratiform clouds in large-scale models. *Mon. Wea. Rev.*, **128**, 1070–1088, 2000.

Smith I.N.. Global climate modelling within CSIRO: 1981 to 2006. *Aus. Met. Mag*, 56 (3): 153-166.

Taylor K.E.. Summarizing multiple aspects of model performance in single diagram. *J. Geophys. Res.*, **106**, D7, 7183–7192, 2001.

Watterson, I.G. and Dix, M.R.. Effective sensitivity and heat capacity in the response of climate models to greenhouse gas and aerosol forcings. *Q. J. Roy. Met. Soc.*, **131**, 259–280, 2005.

Wilson, S.G.. How ocean vertical mixing and accumulation of warm surface water influence the "sharpness" of the equatorial thermocline. *J. Climate*, **13**, 3638–3656, 2000.

14 Appendix: Tcl-Nap Scripts and Procedures

In the following section the Tcl-Nap scripts are listed and briefly described. The following sections have titles coloured red, and refer to the name of a Tcl-Nap script. The ".tcl" suffix has been left of for tidiness of the section headings. The location and relationship between their directory location are shown by Figure 4. Tcl-Nap scripts which are not a top-level scripts (i.e. those that are invoked by the Tcl "source" command) have an alphebetically listed set of scripts that call them them listed below the section title in blue. Below this list and before the script verbatim listing a small description of the top-level scripts is provided in green text.
Most of the daily and monthly Mk3.0 IPCC parameters were generated by the top-level Tcl-Nap processing script ipcc_data_final.tcl found in Section 14.23. In essence, parameters are specified to be extracted from specific Mk3.0 experiments over specific year ranges. This is achieved by calling on one of the next level Tcl-Nap procedures defined by the Tcl-Nap processing scripts provided in Table 25.

Tcl-Nap Script	Parameters	Section
ipcc_calc_mon_mk3ogcm.tcl	2/3D Monthly Ocean	14.21
ipcc_calc_mon_mk3agcm.tcl	2D Monthly Atmosphere	14.20
ipcc_calc_4dmon_mk3agcm.tcl	3D Monthly Atmosphere	14.18
ipcc_calc_4dday_mk3agcm.tcl	3D Daily Atmosphere	14.24
ipcc_calc_day_mk3agcm.tcl	2D Daily Atmosphere	14.19

Table 25: Tcl-Nap scripts defining procedures for computing Mk3.0 2/3D daily and monthly ocean and atmosphere IPCC parameters.

14.1 Preprocessing Scripts

14.1.1 Miscellaneous

14.2 ipcc_data_pcmdi

ipcc_data_pcmdi.tcl is a Tcl-Nap script for copying Mk3.0 IPCC data from the main data store on cherax to another destination disk, and providing reports of missing, problem or transferred data. This has been used to transfer data to disks external to cherax for transferring the data to PCMDI in the U.S.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier October 2004

source ~/.tclshrc

source ~col414/tcl_fun/pad10000.tcl
source ~IPCC/processing/miscellaneous/ipcc_fil3_list1.tcl
source ~IPCC/processing/miscellaneous/ipcc_fil3_list2.tcl

global model_version
set model_version mk3.0

########################################################################
# final files...

source exps1.tcl
##########

source A1a.tcl
ipcc_fil3_list1 $exps $vars

set vars {{rsdt A1a}}
ipcc_fil3_list2 $exps $vars

source A1b.tcl
ipcc_fil3_list2 $exps $vars

source A1c.tcl
ipcc_fil3_list1 $exps $vars

source O1a.tcl
ipcc_fil3_list1 $exps $vars

source O1b.tcl
ipcc_fil3_list1 $exps $vars

source O1c.tcl
ipcc_fil3_list1 $exps $vars

source O1d.tcl
ipcc_fil3_list2 $exps $vars

source O1e.tcl
ipcc_fil3_list1 $exps $vars

source A1f.tcl
ipcc_fil3_list1 $exps $vars

source exps2.tcl
```

```
##########

source A2a.tcl
ipcc_fil3_list1 $exps $vars

source exps3.tcl
##########

source A2b.tcl
ipcc_fil3_list1 $exps $vars

source exps4.tcl
##########

source A3.tcl
ipcc_fil3_list1 $exps $vars

source exps1d.tcl
##########

source O1db.tcl
ipcc_fil3_list2 $exps $vars

source A1ab.tcl
ipcc_fil3_list1 $exps $vars

source A1cb.tcl
ipcc_fil3_list1 $exps $vars

source A2ab.tcl
ipcc_fil3_list1 $exps $vars

#########################################################################

return
exit
```

14.3 ipcc_copy_core

ipcc_data_pcmdi

```
if {[file exists $odir/$ofil] == 1 && [file size $odir/$ofil] == 0} {
file delete -force $idir/$ifil
puts "Deleting null output file."
puts "$odir/$ofil"
}

if {[file exists $idir/$ifil] == 1 && [file size $idir/$ifil] == 0} {
file delete -force $idir/$ifil
puts "Deleting null input file and continuing."
puts "$idir/$ifil"
continue
}

if {[file exists $odir/${ofil}.tmp] == 1} {
puts "Output temporary file exists now, therefore deleting it."
puts "$odir/$ofil"
file delete -force $odir/${ofil}.tmp
}

if {[file exists $idir/$ifil] == 1 && [file exists $odir/$ofil] != 1} {
puts "Copying"

source [glob ~IPCC/processing/miscellaneous/ipcc_copy_basic.tcl]

} elseif {[file exists $idir/$ifil] == 1 && [file exists $odir/$ofil] == 1} {

  if {[file size $idir/$ifil] != [file size $odir/$ofil]} {
  puts "Files have different sizes therefore updating."

source [glob ~IPCC/processing/miscellaneous/ipcc_copy_basic.tcl]

  } elseif {[file mtime $idir/$ifil] == [file mtime $odir/$ofil]} {
  puts "Doing nothing as files have same modification time."
  puts "$idir/$ifil and $odir/$ofil"
  } elseif {[file mtime $idir/$ifil] < [file mtime $odir/$ofil]} {
  puts "Something strange as image archive file newer than local."
  puts "input: $idir/$ifil output:$odir/$ofil"
  } elseif {[file mtime $idir/$ifil] > [file mtime $odir/$ofil]} {
  #if file exists in output directory but its timestamp is older than input directory.
  puts "Overwriting"

source [glob ~IPCC/processing/miscellaneous/ipcc_copy_basic.tcl]

  };#mtime

};#exists
```

14.4 ipcc_copy_basic

ipcc_copy_core.tcl

```
puts "$idir/$ifil to $odir/$ofil"

file copy -force $idir/$ifil /mnt/laciedisk/ipcc/${ofil}.tmp

if {[file exists /mnt/laciedisk/ipcc/${ofil}.tmp] == 1} {

file rename -force /mnt/laciedisk/ipcc/${ofil}.tmp $odir/${ofil}

} else {
puts "Problem, temporary file should exist on remote disk but doesn't."
exit
}
```

14.5 ipcc_fil1_list1

ipcc_data_pcmdi

```
proc ipcc_fil1_list1 {exps vars del du_incoming du_outgoing} {
upvar $du_outgoing dusage
global model_version

set dusage $du_incoming

#these are a function of time

set laciedisk 1;#switch on
set laciedisk 0;#switch off

set pdir [glob ~IPCC/data/${model_version}]
set rdir /mnt/laciedisk/ipcc/data/${model_version}
set rdir /work/IPCC/data/${model_version}

set odir [glob ~IPCC/data/${model_version}]
set ofil [lindex [lindex $vars 0] 1].txt
if {$del == 0} {
file delete $odir/$ofil
set ofhd [open $odir/$ofil w]
} else {
set ofhd [open $odir/$ofil a]
}
for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

catch {unset ifils};set pcnt 0
catch {unset rfils};set rcnt 0

for {set ynow [lindex [lindex $exps $en] 3]} {$ynow <= [lindex [lindex $exps $en] 4]}
{set ynow [expr $ynow+[lindex [lindex $exps $en] 5]]} {

set yfst [pad10000 $ynow];set ylst [pad10000 [expr $ynow+[lindex [lindex $exps $en] 5]-1]]

set idir "[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]"

if {$yfst == $ylst} {
set ifil "[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_${ylst}_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2].nc"
} else {
set ifil "[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_${yfst}-${ylst}_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2].nc"
};#yfst

################################################################################

lappend ifils $ifil

################################################################################
if {[file exists $pdir/$idir/$ifil] == 1} {
puts $ofhd "[clock format [file mtime $pdir/$idir/$ifil] -format "%Y %m %d %H %M %S"] $ifil"
```

```
set dusage [expr $dusage+[file size $pdir/$idir/$ifil]]
} else {
incr pcnt
};#exists

if {$laciedisk == 1} {
if {[file exists $rdir/$idir/$ifil] == 1} {
} else {
incr rcnt
};#exists
}
##############################################################################

};#ynow

set dfil "[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2]"

##############################################################################
if {$pcnt == 0} {
set psuff "des"
} else {
set psuff "txt"
};#pcnt

if {$rcnt == 0} {
set rsuff "des"
} else {
set rsuff "txt"
};#rcnt
##############################################################################

if {[file exists /cs/datastore/IPCC/IPCC/des] == 1} {
ipcc_make_des $ifils $pdir/$idir /cs/datastore/IPCC/IPCC/des $dfil $psuff
[lindex [lindex $exps $en] 3] [lindex [lindex $vars $vn] 1]
}

if {$laciedisk == 1 && [file exists /mnt/laciedisk/ipcc/des] == 1} {
ipcc_make_des $ifils $rdir/$idir /mnt/laciedisk/ipcc/des $dfil $rsuff
[lindex [lindex $exps $en] 3] [lindex [lindex $vars $vn] 1]
}

##############################################################################

};#en
};#vn
close $ofhd
};#ipcc_fil1_list1
```

14.6 ipcc_fil1_list2

ipcc_data_pcmdi

```
proc ipcc_fil1_list2 {exps vars} {
#these are not a function of time
global model_version
set pdir [glob ~IPCC/data/${model_version}]
set odir [glob ~IPCC/data/${model_version}]
set ofil [lindex [lindex $vars 0] 1].txt
file delete $odir/$ofil
set ofhd [open $odir/$ofil w]
for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

set ifil "[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]/
[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2].nc"

if {[file exists $pdir/$ifil] == 1} {
puts $ofhd "[clock format [file mtime $pdir/$ifil] -format "%Y %m %d %H %M %S"] $ifil"
} else {
puts $ofhd "
}

};#en
};#vn
close $ofhd
};#ipcc_fil1_list2
```

14.7 ipcc_fil1_list3

ipcc_data_pcmdi

```
proc ipcc_fil1_list3 {exps vars} {
#this is to list files used in preparation of final ipcc files hfogo, stfmmc etc.
global model_version
set pdir [glob ~IPCC/data/${model_version}]
set odir [glob ~IPCC/data/${model_version}]
set ofil [lindex [lindex $vars 0] 1]_prep.txt
file delete $odir/$ofil
set ofhd [open $odir/$ofil w]
for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

for {set ynow [lindex [lindex $exps $en] 3]} {$ynow <= [lindex [lindex $exps $en] 4]}
{set ynow [expr $ynow+[lindex [lindex $exps $en] 5]]} {

set yfst [pad10000 $ynow];set ylst [pad10000 [expr $ynow+[lindex [lindex $exps $en] 5]-1]]

if {$yfst == $ylst} {
set ifil "[lindex [lindex $vars $vn] 0]/[lindex [lindex $vars $vn] 0]_
[lindex [lindex $vars $vn] 1]_${ylst}_[lindex [lindex $exps $en] 0]_${model_version}_
[lindex [lindex $exps $en] 2].nc"
} else {
set ifil "[lindex [lindex $vars $vn] 0]/[lindex [lindex $vars $vn] 0]_
[lindex [lindex $vars $vn] 1]_${yfst}-${ylst}_[lindex [lindex $exps $en] 0]_
${model_version}_[lindex [lindex $exps $en] 2].nc"
}

if {[file exists $pdir/$ifil] == 1} {
puts $ofhd "[clock format [file mtime $pdir/$ifil] -format "%Y %m %d %H %M %S"] $ifil"
} else {
puts $ofhd "
}

};#ynow

};#en
};#vn
close $ofhd
};#ipcc_fil1_list3
```

14.8 ipcc_fil2_list1

ipcc_data_pcmdi

```
proc ipcc_fil2_list1 {exps vars} {
set stype "tstamp"
set stype "modified"
#these are a function of time
set pdir [glob ~IPCC/data/mk3.0]
for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

catch {unset ifils};set cnt 0
for {set ynow [lindex [lindex $exps $en] 3]} {$ynow <= [lindex [lindex $exps $en] 4]}
{set ynow [expr $ynow+[lindex [lindex $exps $en] 5]]} {

set yfst [pad10000 $ynow];set ylst [pad10000 [expr $ynow+[lindex [lindex $exps $en] 5]-1]]

if {$yfst == $ylst} {
set ifil "[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]/[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_
${ylst}_[lindex [lindex $exps $en] 0]_[lindex [lindex $exps $en] 2].nc"
} else {
set ifil "[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]/[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_
${yfst}-${ylst}_[lindex [lindex $exps $en] 0]_[lindex [lindex $exps $en] 2].nc"
}

lappend ifils $ifil

if {[file exists $pdir/$ifil] == 1} {

if {$stype == "modified"} {
set tokens [clock format [file mtime $pdir/$ifil] -format "%Y %m %d %H %M %S"]

set year [lindex $tokens 0]
set month [lindex $tokens 1]
set day [lindex $tokens 2]
set hour [lindex $tokens 3]
set minute [lindex $tokens 4]
set second [lindex $tokens 5]

if {$year == 2004 && $month <= 9 && $day <= 14} {
puts "delting $pdir/$ifil year=$year month=$month day=$day"
file delete -force $pdir/$ifil
};#delete

} elseif {$stype == "tstamp"} {
set history [[nap_get netcdf $pdir/$ifil :history]]
set stamp [string range $history [string first year: $history]
[expr [string first :UTC $history]+3]]
set tokens [split $stamp :]
puts "$tokens"
};#stype

} else {
```

```
incr cnt
};#exists

};#ynow

};#en
};#vn
};#ipcc_fil2_list1
```

14.9 ipcc_fil2_list2

ipcc_data_pcmdi

```
proc ipcc_fil2_list2 {exps vars} {
set stype "tstamp"
set stype "modified"
#these are not a function of time
set pdir [glob ~IPCC/data/mk3.0]
for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

set ifil "[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]/[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_
[lindex [lindex $exps $en] 0]_[lindex [lindex $exps $en] 2].nc"

if {[file exists $pdir/$ifil] == 1} {

if {$stype == "modified"} {
set tokens [clock format [file mtime $pdir/$ifil] -format "%Y %m %d %H %M %S"]

set year [lindex $tokens 0]
set month [lindex $tokens 1]
set day [lindex $tokens 2]
set hour [lindex $tokens 3]
set minute [lindex $tokens 4]
set second [lindex $tokens 5]

if {$year == 2004 && $month <= 9 && $day <= 14} {
puts "delting $pdir/$ifil year=$year month=$month day=$day"
file delete -force $pdir/$ifil
};#delete

} elseif {$stype == "tstamp"} {
set history [[nap_get netcdf $pdir/$ifil :history]]
set stamp [string range $history [string first year: $history]
[expr [string first :UTC $history]+3]]
set tokens [split $stamp :]
puts "$tokens"
};#stype
};#exists

};#en
};#vn
};#ipcc_fil2_list2
```

14.10 ipcc_fil3_list1

ipcc_data_pcmdi

```
proc ipcc_fil3_list1 {exps vars} {
#these are a function of time
global model_version
set pdir [glob ~IPCC/data/$model_version]
set rdir /work/cec006/ipcc/data
set rdir /work/IPCC/ipcc/data
set rdir /mnt/laciedisk/ipcc/data

for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

catch {unset ifils};set cnt 0
for {set ynow [lindex [lindex $exps $en] 3]} {$ynow <= [lindex [lindex $exps $en] 4]}
{set ynow [expr $ynow+[lindex [lindex $exps $en] 5]]} {

set yfst [pad10000 $ynow];set ylst [pad10000 [expr $ynow+[lindex [lindex $exps $en] 5]-1]]

if {$yfst == $ylst} {
set ifil "[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_${ylst}_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2].nc"
} else {
set ifil "[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_${yfst}-${ylst}_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2].nc"
}

set ofil $ifil

set idir "$pdir/[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]"
set odir "$rdir/[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]"

puts $idir
puts $odir

if {[file isdirectory $odir/$ifil] != 1} {
file mkdir $odir
}

lappend ifils $ifil

source [glob ~IPCC/processing/miscellaneous/ipcc_copy_core.tcl]

};#ynow

};#en
};#vn
};#ipcc_fil3_list1
```

14.11 ipcc_fil3_list2

ipcc_data_pcmdi

```
proc ipcc_fil3_list2 {exps vars} {
#these are not a function of time
global model_version
set pdir [glob ~IPCC/data/$model_version]
set rdir /work/IPCC/data/mk3.0
set rdir /mnt/laciedisk/ipcc/data/mk3.0
set rdir /mnt/laciedisk/ipcc/data
for {set vn 0} {$vn < [llength $vars]} {incr vn} {
for {set en 0} {$en < [llength $exps]} {incr en} {

set ifil "[lindex [lindex $vars $vn] 0]_[lindex [lindex $vars $vn] 1]_
[lindex [lindex $exps $en] 0]_${model_version}_[lindex [lindex $exps $en] 2].nc"

set ofil $ifil

set idir "$pdir/[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]"
set odir "$rdir/[lindex [lindex $exps $en] 0]/[lindex [lindex $vars $vn] 1]/
[lindex [lindex $exps $en] 2]"

if {[file isdirectory $odir/$ifil] != 1} {
file mkdir $odir
}

source [glob ~IPCC/processing/miscellaneous/ipcc_copy_core.tcl]

};#en
};#vn
};#ipcc_fil3_list2
```

14.12 unesco

ipcc_rhopoto.tcl
unesco.tcl is a Tcl-Nap script (procedure) used to help define parameter *rhopoto* in Table 13.

```
proc unesco {t s pin} {
#     this subroutine calculates the density of seawater using the
#     standard equation of state recommended by unesco(1981).
#
#     input [units]:
#       in-situ temperature (t): [degrees centigrade]
#       salinity (s): [practical salinity units]
#       pressure (pin): [decibars, approx. as meters of depth]
#     output [units]:
#       density(rho): kilograms per cubic meter
#
#     references:
#        Gill, A., Atmosphere-Ocean Dynamics: International Geophysical
#         Series No. 30. Academic Press, London, 1982, pp 599-600.
#        UNESCO, 10th report of the joint panel on oceanographic tables
#          and standards. UNESCO Tech. Papers in Marine Sci. No. 36,
#          Paris, 1981.
#
# taken from subroutine unesco mom2.2 routine denscoef.F
      nap "c1p5 = 1.5e0"
#
#  convert from depth [m] (decibars) to bars
      nap "p = pin * 1.0e-1"
#
      nap "rw =     9.99842594e2 + 6.793952e-2*t - 9.095290e-3*t**2
              + 1.001685e-4*t**3 - 1.120083e-6*t**4 + 6.536332e-9*t**5"
#
      nap "rsto =   rw + (8.24493e-1 - 4.0899e-3*t + 7.6438e-5*t**2
              - 8.2467e-7*t**3 + 5.3875e-9*t**4) * s
             + (-5.72466e-3 + 1.0227e-4*t - 1.6546e-6*t**2) * s**c1p5
             + 4.8314e-4 * s**2"
#
      nap "xkw =     1.965221e4 + 1.484206e2*t - 2.327105e0*t**2 +
               1.360477e-2*t**3 - 5.155288e-5*t**4"
#
      nap "xksto =   xkw + (5.46746e1 - 6.03459e-1*t + 1.09987e-2*t**2
              - 6.1670e-5*t**3) * s
             + (7.944e-2 + 1.6483e-2*t - 5.3009e-4*t**2) * s**c1p5"
#
      nap "xkstp =   xksto + (3.239908e0 + 1.43713e-3*t + 1.16092e-4*t**2
              - 5.77905e-7*t**3) * p
             + (2.2838e-3 - 1.0981e-5*t - 1.6078e-6*t**2) * p * s
             + 1.91075e-4 * p * s**c1p5
             + (8.50935e-5 - 6.12293e-6*t + 5.2787e-8*t**2) * p**2
             + (-9.9348e-7 + 2.0816e-8*t + 9.1697e-10*t**2) * p**2 * s"
#
      nap "result =    rsto / (1.0e0 - p/xkstp)"
}
#
#return
#exit
```

14.12.1 Tools

14.13 hostname

ipcc_calc_4dday_mk3agcm.tcl, ipcc_calc_4dmon_mk3agcm.tcl, ipcc_calc_day2mon.tcl, ipcc_calc_day2mon_wap.tcl, ipcc_calc_day_mk3agcm.tcl, ipcc_calc_mon_mk3ogcm.tcl, ipcc_ei_cdd.tcl, ipcc_ei_etr.tcl, ipcc_ei_fd.tcl, ipcc_ei_gsl.tcl, ipcc_ei_hwdi.tcl, ipcc_ei_pq95.tcl, ipcc_ei_r10.tcl, ipcc_ei_r5d.tcl, ipcc_ei_r95t.tcl, ipcc_ei_sdii.tcl, ipcc_ei_tmaxnorm.tcl, ipcc_ei_tn90.tcl, ipcc_ei_tnq90.tcl, ipcc_hfogo_ferret.tcl, ipcc_hfogo.tcl, ipcc_intro.tcl, ipcc_mrros.tcl, ipcc_orog.tcl, ipcc_psf.tcl, ipcc_qflux.tcl, ipcc_rhopoto.tcl, ipcc_rldscs.tcl, ipcc_rlus.tcl, ipcc_rsdscs.tcl, ipcc_rsdt.tcl, ipcc_rsuscs, pcc_rsus.tcl, ipcc_rtmt.tcl, ipcc_scripts.tcl, ipcc_sftgif.tcl, ipcc_sftlf.tcl, ipcc_snc.tcl, ipcc_snw.tcl, ipcc_stfmmc_ferret.tcl, ipcc_stfmmc.tcl, ipcc_zg.tcl, ipcc_zobt.tcl

```
proc hostname {} {
set hostname [exec hostname]
set HN [string range $hostname 0 [expr [string first "." $hostname]-1]];
if {$HN == ""} {set HN $hostname}
return $HN
}
```

14.14 ipcc_att

ipcc_calc_4dday_mk3agcm.tcl, ipcc_calc_4dmon_mk3agcm.tcl, ipcc_calc_day_mk3agcm.tcl , ipcc_calc_mon_mk3agcm.tcl, ipcc_calc_mon_mk3ogcm.tcl, ipcc_ei_cdd.tcl, ipcc_ei_etr.tcl, ipcc_ei_fd.tcl, ipcc_ei_gsl.tcl, ipcc_ei_hwdi.tcl, ipcc_ei_pq95.tcl, ipcc_ei_r10.tcl, ipcc_ei_r5d.tcl, ipcc_ei_r95t.tcl, ipcc_ei_sdii.tcl, ipcc_ei_tmaxnorm.tcl, ipcc_ei_tn90.tcl, ipcc_ei_tnq90.tcl, ipcc_hfogo.tcl, ipcc_orog.tcl, ipcc_qflux.tcl, ipcc_rep.tcl, ipcc_rhopoto.tcl, ipcc_rsdt.tcl, ipcc_sftgif.tcl, ipcc_sftlf.tcl, ipcc_stfmmc.tcl, ipcc_zobt.tcl

```
nap "lgn = '$olgn'"
nap "stn = '$ostn'"
nap "uni = '$ouni'"
nap "orn = '$ivar'"

if {$time_freq == "yearly"} {
  nap "cem = 'time: mean (interval: 365 days)'"
} elseif {$time_freq == "monthly"} {
  if {$ovar == "tasmin"} {
  nap "cem = 'time: minimum within days time: mean over days'"
  } elseif {$ovar == "tasmax"} {
  nap "cem = 'time: maximum within days time: mean over days'"
  } else {
  nap "cem = 'time: mean (interval: 15 minutes)'"
  }
} elseif {$time_freq == "daily"} {
  if {$ovar == "tasmin"} {
  nap "cem = 'time: minimum'"
  } elseif {$ovar == "tasmax"} {
  nap "cem = 'time: maximum'"
  } else {
  nap "cem = 'time: mean (interval: 15 minutes)'"
  }

} elseif {$time_freq == "12hourly"} {

  nap "cem = 'time: mean (interval: 15 minutes)'"

} elseif {$time_freq == "6hourly"} {

  nap "cem = 'time: mean (interval: 15 minutes)'"

} elseif {$time_freq == "3hourly"} {
puts "No valid at moment."
return
}

if {$ovar == "hfogo"} {
nap "cem = 'time: mean (interval: 15 minutes) longitude: sum'"
}

nap "history = 'Processed from model output using tcl-nap procedure $proc_name.
Input variable $ivar either raw model file or IPCC derived file.'"

if {$nbeg == 0 && $nend == 0} {
} elseif {$time_freq != "invariant"} {
 if {$nbeg == $nend} {
 nap "history = history // ' Model year=$nbeg.'"
```

```
 } else {
 nap "history = history // ' Model years=$nbeg-$nend.'"
 }
}

if {$var_derivation != ""} {
nap "history = history // ' $var_derivation.'"
}

if {$exter_scale != ""} {
nap "history = history // ' Scale factor=$exter_scale.'"
}

if {$exter_offset != ""} {
nap "history = history // ' Offset=$exter_offset.'"
}

#probably can get rid of this line as land_sea_interp set for all variables.
if {$land_sea_interp != ""} {
if {$land_sea_interp == "yes"} {
nap "history = history // ' Land/sea mask used for horizontal
regridding (if applicable) to full grid.'"
} else {
nap "history = history // ' Land/sea mask not used for horizontal regridding
(if applicable) to full grid.'"
}
}

##############################################################################
if {[info vars land_sea_horiz_interp] != ""} {

set tags [split $land_sea_horiz_interp ,]

if {$land_sea_interp == "yes" && [lindex $tags 0] == "no"} {
puts "Conflict with variable settings."
return
}

if {$land_sea_interp == "no" && [lindex $tags 0] == "yes"} {
puts "Conflict with variable settings."
return
}

if {[expr fmod([llength $tags]-1,3)] == 0.0} {
set cnt [expr ([llength $tags]-1)/3.0]
for {set ccc 0} {$ccc < $cnt} {incr ccc} {
set ia [expr $ccc*3+1]
set ib [expr $ccc*3+2]
set ic [expr $ccc*3+3]
nap "history = history // ' [lindex $tags $ia]=[lindex $tags $ib]x[lindex $tags $ic].'"
}
} else {
puts "Problem with number of elements in list."
return
}
};#land_sea_horiz_interp
```

```
################################################################################

$lgn netcdf $tdir/$ofil $ovar:long_name
$stn netcdf $tdir/$ofil $ovar:standard_name
$orn netcdf $tdir/$ofil $ovar:original_name
$uni netcdf $tdir/$ofil $ovar:units
$history netcdf $tdir/$ofil $ovar:history

if {$time_freq != "invariant"} {
$cem netcdf $tdir/$ofil $ovar:cell_methods
}

if {$ovar == "pr"} {
nap "com = 'includes all types (rain, snow, large-scale, convective, etc.)'"
$com netcdf $tdir/$ofil $ovar:comment
}

if {$zdim == "dummy" || $ovar == "tasmin" || $ovar == "tasmax" || $ovar == "uas"
|| $ovar == "vas"} {
nap "coo = 'height'"
$coo netcdf $tdir/$ofil $ovar:coordinates

if {$ovar == "tas" || $ovar == "tasmin" || $ovar == "tasmax" || $ovar == "huss"} {
nap "height = f64(2.0)"
} elseif {$ovar == "uas" || $ovar == "vas"} {
nap "height = f64(10.0)"
} elseif {$ovar == "mrsos" || $ovar == "vas"} {
nap "height = f64(0.1)"
} else {
puts "Don't know that one."
return
}

$height set missing
$height netcdf $tdir/$ofil height
nap "long_name = `height`"
nap "standard_name = `height`"
nap "axis = `Z`"
nap "units = `m`"
nap "positive = `up`"
$long_name netcdf $tdir/$ofil height:long_name
$standard_name netcdf $tdir/$ofil height:standard_name
$axis netcdf $tdir/$ofil height:axis
$units netcdf $tdir/$ofil height:units
$positive netcdf $tdir/$ofil height:positive
};#zdim

if {$zdim == "dummy2"} {
nap "coo = 'plev'"
$coo netcdf $tdir/$ofil $ovar:coordinates

if {$ovar == "rsntp" || $ovar == "rlntp" || $ovar == "rsntpcs" || $ovar == "rlntpcs"} {
nap "plev = f64(20000.0)"
} else {
puts "Don't know that one."
return
```

```
}

$plev set missing
$plev netcdf $tdir/$ofil plev
nap "long_name = ‘plev‘"
nap "standard_name = ‘plev‘"
nap "axis = ‘Z‘"
nap "units = ‘hPa‘"
nap "positive = ‘up‘"
$long_name netcdf $tdir/$ofil plev:long_name
$standard_name netcdf $tdir/$ofil plev:standard_name
$axis netcdf $tdir/$ofil plev:axis
$units netcdf $tdir/$ofil plev:units
$positive netcdf $tdir/$ofil plev:positive
};#zdim

if {$ovar == "hfss" || $ovar == "hfls" || $ovar == "rlus" || $ovar == "rsus" || $ovar == "rsu
nap "positive = ‘up‘"
$positive netcdf $tdir/$ofil $ovar:positive
} elseif {$ovar == "rlds" || $ovar == "rsds" || $ovar == "rsdt" || $ovar == "rtmt"
|| $ovar == "rsntp" || $ovar == "rsntpcs" || $ovar == "rsdscs" || $ovar == "rldscs"
|| $ovar == "rgn" || $ovar == "sgn" || $ovar == "tauu" || $ovar == "tauv"} {
nap "positive = ‘down‘"
$positive netcdf $tdir/$ofil $ovar:positive
};#ovar

if {$xdim == "region"} {
nap "coo = ’geo_region’"
$coo netcdf $tdir/$ofil $ovar:coordinates
};#xdim

nap "title = ’CSIRO model output prepared for IPCC Fourth Assessment’"
nap "institution = ’CSIRO (CSIRO Atmospheric Research, Melbourne, Australia)’"
if {$model_version == "mk3.0"} {
nap "source = ’CSIRO Mk3.0 (2000): atmosphere: spectral (T63L18);
ocean: MOM2.2 (1.875x0.925L31)’"
} elseif {$model_version == "mk3.5"} {
nap "source = ’CSIRO Mk3.5d (2005): atmosphere: spectral (T63L18);
ocean: MOM2.2 (1.875x0.925L31)’"
} elseif {$model_version == "ncep" || $model_version == "era40" || $model_version == "silo"
|| $model_version == "era40hr"} {
} elseif {$model_version == "mk3.5A"} {
nap "source = ’CSIRO Mk3.5A (2005): atmosphere: spectral (T63L18);
ocean: MOM2.2 (1.875x0.925L31)’"
nap "source = ’’"
} else {
puts "Don’t know that version."
return
}
nap "contact = ’Mark Collier (Mark.Collier@csiro.au), Martin Dix (Martin.Dix@csiro.au),
Tony Hirst (Tony.Hirst@csiro.au)’"
nap "project_id = ’IPCC Fourth Assessment’"
nap "Conventions = ’CF-1.0’"
nap "references = ’Model described by Gordon et al. The CSIRO Mk3 Climate System Model,
2002, www.dar.csiro.au/publications/gordon_2002a.pdf’"
```

```
nap "history = 'Date/Time stamp=[exec date
-u "+year:%Y:month:%m:day:%d:hour:%H:minute:%m:second:%S:UTC"].
Processed from model output using tcl version [info tclversion].'"

if {$time_freq == "monthly"} {
nap "table_id = 'Table [lindex $table 0]'"
} elseif {$time_freq == "daily"} {
nap "table_id = 'Table [lindex $table 1]'"
} elseif {$time_freq == "3hourly"} {
nap "table_id = 'Table [lindex $table 2]'"
} elseif {$time_freq == "yearly"} {
nap "table_id = 'Table [lindex $table 3]'"
} elseif {$time_freq == "6hourly"} {
nap "table_id = 'Table [lindex $table 4]'"
} else {
#eg invariant
nap "table_id = 'Table [lindex $table 0]'"
}

if {$iflg == 1} {
nap "comment = 'SRES B1 experiment with CSIRO Mk 3.0 model, starting from year 1939
(model year 189) of 20C3M expierment. Radiative forcings held constant from year 2100.'"
nap "experiment_id = '550 ppm stabilization experiment (SRES B1)'"

} elseif {$iflg == 2} {
nap "comment = 'Pre-industrial control experiment with CSIRO Mk 3.0 model, for model
years 121 to 500.'"
nap "experiment_id = 'pre-industrial control experiment'"

} elseif {$iflg == 3} {
nap "comment = '20th Century Experiment (years 1871-2000) with CSIRO Mk 3.0 model,
starting from model year 120 of PIcntrl experiment.'"
nap "experiment_id = 'climate of the 20th Century experiment (20C3M)'"

} elseif {$iflg == 4} {
nap "comment = 'Not part of the IPCC set.'"
nap "experiment_id = 'not part of IPCC'"

} elseif {$iflg == 5} {
nap "comment = 'Commitment experiment with CSIRO Mk 3.0 model, starting from year 2000
(model year 250) of 20C3M experiment. Radiative forcings held constant at year 2000 values
for entire experiment.'"
nap "experiment_id = 'committed climate change experiment'"

} elseif {$iflg == 6} {
nap "comment = 'CMIP transient experiment with CSIRO Mk 3.0 model, starting from year 120
of PIcntrl and continuing for 80 years. CO2 increased at 1 percent per annum.'"
nap "experiment_id = '1%/year CO2 increase experiment (to doubling)'"

} elseif {$iflg == 7} {
nap "comment = 'Separate realisation of control experiment with CSIRO Mk 3.0 model, starting
from model year 120 of PIcntrl and continuing over same interval as 1pctto2x experiment, and
with radiative forcings held same as PIcntrl. Atmospheric fields retained on the full
model grid.'"
nap "experiment_id = 'pre-industrial control experiment'"
```

```
} elseif {$iflg == 8} {
nap "comment = 'SRES A1B experiment with CSIRO Mk 3.0 model, starting from year 2000
(model year 250) of 20C3M experiment. Radiative forcings held constant from year 2100.'"
nap "experiment_id = '720 ppm stabilization experiment (SRES A1B)'"

} elseif {$iflg == 9} {
nap "comment = 'Slab ocean control experiment, years 18-37.'"
nap "experiment_id = 'slab ocean control experiment'"

} elseif {$iflg == 10} {
nap "comment = 'Slab ocean 2xCO2 equilibrium experiment, years 18-37.'"
nap "experiment_id = '2xCO2 equilibrium experiment'"

} elseif {$iflg == 11} {
nap "comment = 'Not part of the IPCC set.'"
nap "experiment_id = 'not part of IPCC'"

} elseif {$iflg == 12} {
nap "comment = 'SRES A2 experiment (years 2001-2250) with CSIRO Mk 3.0 model, starting
from model year 251 of 20C3M experiment.'"
nap "experiment_id = 'SRES A2 experiment'"

} elseif {$iflg == 13} {
nap "comment = 'Slab ocean control experiment, years 1-50.'"
nap "experiment_id = 'slab ocean control experiment'"

} elseif {$iflg == 14} {
nap "comment = 'Slab ocean 2xCO2 equilibrium experiment, years 1-50.'"
nap "experiment_id = '2xCO2 equilibrium experiment'"

} elseif {$iflg == 15} {
nap "comment = '20th Century Experiment (years 1871-2000) with CSIRO Mk 3.0 model,
starting from model year 130 of PIcntrl experiment.'"
nap "experiment_id = 'climate of the 20th Century experiment (20C3M)'"

} elseif {$iflg == 16} {
nap "comment = 'Pre-industrial control experiment with CSIRO Mk 3.5d model, for
model years 1 to 261.'"
nap "experiment_id = 'pre-industrial control experiment'"

} elseif {$iflg == 17} {
nap "comment = 'Spinup control experiment with CSIRO Mk 3.0 model, for model years
1701 to 1710. Synchronous time-stepping.'"
nap "experiment_id = 'spinup control experiment'"

} elseif {$iflg == 19} {
nap "comment = '20th Century Experiment (years 1871-2000) with CSIRO Mk 3.0 model,
starting from model year 140 of PIcntrl experiment.'"
nap "experiment_id = 'climate of the 20th Century experiment (20C3M)'"

} elseif {$iflg == 20} {
nap "comment = '20th Century Experiment (years 1871-2000) with CSIRO Mk 3.5d model,
starting from model year 171 of PIcntrl experiment.'"
nap "experiment_id = 'climate of the 20th Century experiment (20C3M)'"

} elseif {$iflg == 21} {
```

```
nap "comment = 'SRES A2 Experiment (years 2001-2100) with CSIRO Mk 3.5d model, starting
from model year 171 of PIcntrl experiment.'"
nap "experiment_id = 'SRES A2 experiment'"

} elseif {$iflg == 22} {
nap "comment = 'CMIP transient experiment with CSIRO Mk 3.5d model, starting from year
171 of PIcntrl and continuing for 80 years. CO2 increased at 1 percent per annum.'"
nap "experiment_id = '1%/year CO2 increase experiment (to doubling)'"

} elseif {$iflg == 23} {

nap "comment = 'SRES A1B experiment with CSIRO Mk 3.5d model, starting from year
2000 (model year 300) of 20C3M experiment. Radiative forcings held constant from year 2100.'"
nap "experiment_id = '720 ppm stabilization experiment (SRES A1B)'"

} elseif {$iflg == 24} {

nap "comment = 'SRES B1 experiment with CSIRO Mk 3.5d model, starting from year 2000
(model year 300) of 20C3M expierment. Radiative forcings held constant from year 2100.'"
nap "experiment_id = '550 ppm stabilization experiment (SRES B1)'"

} elseif {$iflg == 26} {
nap "comment = 'Commitment experiment with CSIRO Mk 3.5d model, starting from year 2000
(model year 300) of 20C3M experiment. Radiative forcings held constant at year 2000 values
for entire experiment.'"
nap "experiment_id = 'committed climate change experiment'"

} elseif {$iflg == 27} {
nap "comment = '20th Century Experiment (years 1871-2000) with CSIRO Mk 3.5d model,
starting from model year 191 of PIcntrl experiment.'"
nap "experiment_id = 'climate of the 20th Century experiment (20C3M)'"

} elseif {$iflg == 28} {
nap "comment = '20th Century Experiment (years 1871-2000) with CSIRO Mk 3.5d model,
starting from model year 201 of PIcntrl experiment.'"
nap "experiment_id = 'climate of the 20th Century experiment (20C3M)'"

} elseif {$iflg == 29} {
nap "comment = 'Slab ocean control experiment, years 21-40.'"
nap "experiment_id = 'slab ocean control experiment'"

} elseif {$iflg == 30} {
nap "comment = 'Slab ocean 2xCO2 equilibrium experiment, years 21-40.'"
nap "experiment_id = '2xCO2 equilibrium experiment'"

} elseif {$iflg == 31} {
nap "comment = 'Pre-industrial control experiment with CSIRO Mk 3.5A model, for model
years 1 to 261.'"
nap "experiment_id = 'pre-industrial control experiment'"

} elseif {$iflg == 36} {
nap "comment = 'Present day control experiment with CSIRO Mk 3.5A model, for model
years 1 to 70.'"
nap "experiment_id = 'present day control experiment'"

} elseif {$iflg == 100} {
```

```
#ncep
return

} elseif {$iflg == 101} {

#era40
return

} elseif {$iflg == 102} {

#silo
return

} elseif {$iflg == 103} {

#era40hr
return

} else {
puts "Not prepared yet."
return
}

$title netcdf ${tdir}/${ofil} :title
$institution netcdf ${tdir}/${ofil} :institution
$source netcdf ${tdir}/${ofil} :source
$contact netcdf ${tdir}/${ofil} :contact
$project_id netcdf ${tdir}/${ofil} :project_id
$experiment_id netcdf ${tdir}/${ofil} :experiment_id
$realization netcdf ${tdir}/${ofil} :realization
$Conventions netcdf ${tdir}/${ofil} :Conventions
$references netcdf ${tdir}/${ofil} :references
$comment netcdf ${tdir}/${ofil} :comment
$history netcdf ${tdir}/${ofil} :history
$table_id netcdf ${tdir}/${ofil} :table_id
```

14.15 ipcc_bnds_agcm

ipcc_calc_4dday_mk3agcm.tcl, ipcc_calc_4dmon_mk3agcm, ipcc_calc_day_mk3agcm.tcl, ipcc_calc_mon_mk3agcm.tcl, ipcc_ei_cdd.tcl, ipcc_ei_etr.tcl, ipcc_ei_fd.tcl, ipcc_ei_gsl.tcl, ipcc_ei_hwdi.tcl, ipcc_ei_pq95.tcl, ipcc_ei_r10.tcl, ipcc_ei_r5d.tcl, ipcc_ei_r95t.tcl, ipcc_ei_sdii.tcl, ipcc_ei_tmaxnorm.tcl, ipcc_ei_tn90.tcl, ipcc_ei_tnq90.tcl, ipcc_orog.tcl, ipcc_qflux.tcl, ipcc_rsdt.tcl, ipcc_sftgif.tcl, ipcc_sftlf.tcl, ipcc_zobt

```
if {[info vars surface_pressure] != ""} {
nap "lc = [nap_get netcdf $tdir/$ofil time]"
xyzt_edges $lc "" lbnd llo lhi lb
nap "yc=cv(va,2)"
nap "xc=cv(va,3)"
} elseif {$time_freq != "invariant"} {
nap "lc = [nap_get netcdf $tdir/$ofil time]"
xyzt_edges $lc "" lbnd llo lhi lb
nap "yc=cv(va,1)"
nap "xc=cv(va,2)"
} else {
nap "yc=cv(va,0)"
nap "xc=cv(va,1)"
}

if {$time_freq == "monthly"} {
nap "num_times = nels(lc)"
nap "days_in_month = {31 28 31 30 31 30 31 31 30 31 30 31}"
nap "day_list_half = reshape(days_in_month,num_times)/2.0"

nap "tlo = lc - day_list_half"
nap "thi = lc + day_list_half"

nap "lbnd = transpose(tlo /// thi)"
};#time_freq=monthly

if {$iflg == 102} {
xyzt_edges $yc "" ybnd ylo yhi yb
} else {
xyzt_edges $yc {-90 90} ybnd ylo yhi yb
}

xyzt_edges $xc "" xbnd xlo xhi xb

if {$time_freq == "3hourly"} {
if {$ovar == "pr"} {
$lbnd set dim $ldim bnds
$lbnd set missing
$lbnd netcdf $tdir/$ofil time_bnds
}
} elseif {$time_freq != "invariant"} {
$lbnd set dim $ldim bnds
$lbnd set missing
$lbnd netcdf $tdir/$ofil time_bnds
}

$xbnd set dim $xdim bnds
$xbnd set missing
```

```
$xbnd netcdf $tdir/$ofil lon_bnds

$ybnd set dim $ydim bnds
$ybnd set missing
$ybnd netcdf $tdir/$ofil lat_bnds

if {[info vars surface_pressure] == "" && $zdim != " " && $zdim != "dummy" &&
$zdim != "dummy2"} {
$zbnd set dim $zdim bnds
$zbnd set missing
$zbnd netcdf $tdir/$ofil plev_bnds
}

if {$ovar == "cl"} {
source [glob ~IPCC/processing/tools/ipcc_sigma.tcl]
};#ovar=cl
```

14.16 ipcc_bnds_ogcm

ipcc_calc_mon_mk3ogcm.tcl ipcc_hfogo.tcl, ipcc_rhopoto.tcl, ipcc_stfmmc.tcl

```
nap "lc = [nap_get netcdf $tdir/$ofil time]"
nap "yc = [nap_get netcdf $tdir/$ofil lat]"
if {$xdim != "region"} {
nap "xc = [nap_get netcdf $tdir/$ofil lon]"
xyzt_edges $xc "" xbnd xlo xhi xb
}

xyzt_edges $lc "" lbnd llo lhi lb

if {$time_freq == "monthly"} {
nap "num_times = nels(lc)"
nap "days_in_month = {31 28 31 30 31 30 31 31 30 31 30 31}"
nap "day_list_half = reshape(days_in_month,num_times)/2.0"

nap "tlo = lc - day_list_half"
nap "thi = lc + day_list_half"

nap "lbnd = transpose(tlo /// thi)"
};#time_freq=monthly

xyzt_edges $yc {-90 90} ybnd ylo yhi yb

$lbnd set dim $ldim bnds
$lbnd set missing
$lbnd netcdf $tdir/$ofil time_bnds

if {$xdim != "region"} {
$xbnd set dim $xdim bnds
$xbnd set missing
$xbnd netcdf $tdir/$ofil lon_bnds
}

$ybnd set dim $ydim bnds
$ybnd set missing
$ybnd netcdf $tdir/$ofil lat_bnds

if {$rank == 4 && [lindex $shape 1] > 1 && $zdim != " "} {
nap "zc = [nap_get netcdf $tdir/$ofil depth]"
xyzt_edges $zc "" zbnd zlo zhi zb

$zbnd set dim $zdim bnds
$zbnd set missing
$zbnd netcdf $tdir/$ofil depth_bnds
}
```

14.17 ipcc_calc_4dday_mk3agcm

ipcc_data_final.tcl
ipcc_calc_4dday_mk3agcm.tcl is a Tcl-Nap script used to generate 3-dimensional atmospheric fields as a function of time defined in Table 9.

```
proc ipcc_calc_4dday_mk3agcm {rnam ivar predir tdir ybas yoff ygap y1st ynth pdir iflg
pack surface_pressure hfine ipccout mask clobber} {

set proc_name "[info level 1]"
puts "Entering procedure $proc_name"
set time_freq "daily"

set hostname [hostname]

source ~IPCC/processing/tools/ipcc_tspace.tcl

#CSIRO Australia
#Mark Collier November 2003
#last modified: April 2004

if {$iflg == "15"} {
#4 times daily for this experiment.
set ntpd 4
} else {
#s times daily for all others.
set ntpd 2
}

if {$ivar == "t4d"} {
set varp "t"
} elseif {$ivar == "u4d"} {
set varp "u"
} elseif {$ivar == "v4d"} {
set varp "v"
} elseif {$ivar == "w4d"} {
set varp "omega"
} elseif {$ivar == "z4d"} {
set varp "z"
} elseif {$ivar == "s4d"} {
set varp "s"
} elseif {$ivar == "q4d"} {
set varp "q"
} elseif {$ivar == "r4d"} {
set varp "r"
} else {
puts "Don't know that variable."
return
}

source ~IPCC/processing/tools/ipcc_intro.tcl

if {$surface_pressure != ""} {

nap "plev=f64{100000 92500 85000 70000 60000 50000 40000 30000 20000}"

$plev set unit Pa
```

```
}

if {$hfine == "yes"} {
source ~IPCC/processing/tools/ipcc_mask.tcl
}

if {$hfine == "no" && $mask == "yes"} {
puts "As masking variable (psf) is on fine grid, then must"
puts "request output data on fine grid."
return
}

if {$pack == "yes"} {
set rmis -32768
nap "mv = i16($rmis)"
} else {
set rmis -7777777
set rmis -1e34
set rmis 1e20
nap "mv = f32($rmis)"
}
set dim {31 28 31 30 31 30 31 31 30 31 30 31}

set nmy 12;set nmym1 [expr $nmy-1]

set ybeg "";unset ybeg
for {set y $y1st} {$y <= $ynth} {set y [expr $y+$ygap]} {
lappend ybeg $y
}

set yend "";unset yend
foreach y $ybeg {
lappend yend [expr $y+$ygap-1]
}

set tot_tnt 1
for {set dec 1} {$dec <= [llength $ybeg]} {incr dec} {
set decm1 [expr $dec-1]

set nbeg [lindex $ybeg $decm1]
set nend [lindex $yend $decm1]

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

if {$nbeg > 250 && $rnam == "m20"} {
set expname "SRESA2"
} elseif {$nbeg > 260 && $rnam == "n20"} {
set expname "SRESA2"
} elseif {$nbeg > 300 && $rnam == "B35"} {
set expname "SRESA2"
}

catch {unset clevs}
for {set ccc 0} {$ccc < [llength $levs]} {incr ccc} {
if {$ccc < [expr [llength $levs]-1]} {
```

```
append clevs "[lindex $levs $ccc],"
} else {
append clevs "[lindex $levs $ccc]"
}
}

if {$surface_pressure == ""} {
nap "plev=(f64{$levs})*100.0"
$plev set unit Pa
} else {
nap "level=f64{$levs}"
$level set unit 1
}

set odir $predir/$expname/[lindex $table 1]/$ensnum;file mkdir $odir

nap "vert = f32(0..[expr [llength $lvars]-1])"

if {$ipccout == "yes" } {
set cbeg [expr $ybas-$yoff+$nbeg]
set cend [expr $cbeg+$ygap-1]
set YBEG [pad10000 ${cbeg}]
set YEND [pad10000 ${cend}]
} else {
set YBEG [pad10000 ${nbeg}]
set YEND [pad10000 ${nend}]
}

if {$YBEG == $YEND} {
set YSTR "$YBEG"
} else {
set YSTR "$YBEG-$YEND"
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 1]_${YSTR}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 1]_${YSTR}_${rnam}_${model_version}_${ensnum}.nc
}

puts "Output file=$odir/$ofil"
puts "Temporary file=$tdir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
} else {
file delete -force ${tdir}/${ofil}
file delete -force ${odir}/${ofil}
}

set day_beg 1
set time_beg 1

set tnt 1
for {set ynow $nbeg} {$ynow <= $nend} {incr ynow} {
```

```
source ~IPCC/processing/tools/ipcc_dinfo_histb_mk3agcm.tcl

set mfils ""
for {set mnow 1} {$mnow <= $nmy} {incr mnow} {

if {$runname == "st1"} {
  if {$ynow <= 385} {
  lappend mfils $mdir/histb.m[pad10 $mnow].y$ynow.nc
  } else {
  lappend mfils $mdir/histb.m[pad10 $mnow].y$ynow.$runname.nc
  }
} elseif {$runname == "B35" || $runname == "D35" || $runname == "E35" || $runname == "M35"
|| $runname == "a1b" || $runname == "s20" || $rnam == "qv3" || $rnam == "qv4"
|| $rnam == "n20"} {

lappend mfils $mdir/histb.m[pad10 $mnow].y$ynow.$runname.nc

} else {
lappend mfils $mdir/histb.m[pad10 $mnow].y$ynow.nc
}
};#mnow

if {$mask == "yes"} {
lappend mfils [glob ~IPCC/data/mk3.0/20C3M/A1b/run1/orog_A1b_20C3M_mk3.0_run1.nc]
};#mask

if {$surface_pressure == "" && $ovar == "zg"} {
lappend mfils [glob ~dix043/bin/histheights]
}

puts $mfils
dmget_tcl $mdir/$mfils

for {set mnow 1} {$mnow <= $nmy} {incr mnow} {
set mnowm1 [expr $mnow-1]

puts "nbeg=$nbeg ynow=$ynow mnow=$mnow tnt=$tnt"

if {$runname == "st1"} {
  if {$ynow <= 385} {
  set mfil histb.m[pad10 $mnow].y$ynow.nc
  } else {
  set mfil histb.m[pad10 $mnow].y$ynow.$runname.nc
  }
} elseif {$runname == "B35" || $runname == "D35" || $runname == "E35" || $runname == "M35"
|| $runname == "a1b" || $runname == "s20" || $rnam == "qv3" || $rnam == "qv4"
|| $rnam == "n20"} {
set mfil histb.m[pad10 $mnow].y$ynow.$runname.nc
} else {
set mfil histb.m[pad10 $mnow].y$ynow.nc
}
puts "Input file=$mfil"

if {$mnow == 1 && $ynow == $nbeg} {
catch {set land_sea_horiz_interp [[nap_get netcdf $mdir/$mfil $ivar:land_sea_horiz_interp]]}
}
```

```
if {$surface_pressure == "" && $ovar == "zg"} {
file delete -force $tdir/$mfil
exec [glob ~dix043/bin/histheights] --single -z $clevs -i $mdir/$mfil -o $tdir/$mfil
}

if {$mask == "yes"} {
if {$iflg != 6 && $iflg != 7 && $iflg != 13 && $iflg != 14 && $iflg != 15 && $iflg != 16
&& $iflg != 20 && $iflg != 21 && $iflg != 22 && $iflg != 23 && $iflg != 24} {
file delete -force $tdir/psf.m[pad10 $mnow].y$ynow.nc
puts "[glob ~dix043/bin/calc_psf_daily] -i $mdir/$mfil -z [glob
~IPCC/data/20C3M/A1b/run1/orog_A1b_20C3M_run1.nc] -o $tdir/psf.m[pad10 $mnow].y$ynow.nc"
exec [glob ~dix043/bin/calc_psf_daily] -i $mdir/$mfil -z [glob
~IPCC/data/20C3M/A1b/run1/orog_A1b_20C3M_run1.nc] -o $tdir/psf.m[pad10 $mnow].y$ynow.nc
};#iflg
};#mask

set tntm1 [expr $tnt-1]
set YNOW [expr $ybas-$yoff+$ynow]
set NBEG [expr $ybas-$yoff+$nbeg]

nap "time0 = f64([nap_get netcdf $mdir/$mfil time])"
nap "time1 = transpose(reshape(time0,{[expr [$time0 nels]/$ntpd] $ntpd}))"
nap "time = am(time1)"

set day_end [expr $day_beg+[$time nels]-1]
set day_begm1 [expr $day_beg-1]
set day_endm1 [expr $day_end-1]

set time_end [expr $time_beg+[$time nels]-1]
set time_begm1 [expr $time_beg-1]
set time_endm1 [expr $time_end-1]

nap "time_ind = $time_begm1 .. $time_endm1"
set time_beg [expr $time_end+1]

nap "day_ind = $day_begm1..[expr $day_end-$tincre]...$tincre"
nap "t0 = (time*0+${NBEG}*365*1)+day_ind+0.5"

set day_beg [expr $day_end+1]

set lnt 0
if {$surface_pressure == ""} {
set dimen [nap_get netcdf -dimension $tdir/$mfil [lindex $lvars 0]]
set shape [nap_get netcdf -shape $tdir/$mfil [lindex $lvars 0]]
set rank [nap_get netcdf -rank $tdir/$mfil [lindex $lvars 0]]
} else {
set dimen [nap_get netcdf -dimension $mdir/$mfil [lindex $lvars 0]]
set shape [nap_get netcdf -shape $mdir/$mfil [lindex $lvars 0]]
set rank [nap_get netcdf -rank $mdir/$mfil [lindex $lvars 0]]
}

if {$surface_pressure != ""} {
set newshapeA "[expr [lindex $shape 0]/$ntpd] [$plev nels] [lrange $shape 1 2]"
set newshapeC "[expr [lindex $shape 0]/$ntpd] [llength $lvars] [lrange $shape 1 2]"
} else {
```

```
set newshapeA "[expr [lindex $shape 0]/$ntpd] [llength $lvars] [lrange $shape 1 2]"
set newshapeC "[expr [lindex $shape 0]/$ntpd] [llength $lvars] [lrange $shape 1 2]"
}

if {$hfine == "yes"} {
set newshapeA [lreplace $newshapeA 2 3 [lindex $l1s 0] [lindex $l1s 1]]
set newshapeC [lreplace $newshapeC 2 3 [lindex $l1s 0] [lindex $l1s 1]]
}

if {$tnt == 1} {

if {$hfine == "yes"} {
nap "$xdim = f64(cv(l1,1))"
nap "$ydim = f64(cv(l1,0))"
nap "xcoarse = f64([nap_get netcdf $mdir/$mfil lon])"
nap "ycoarse = f64([nap_get netcdf $mdir/$mfil lat])"
} else {
nap "$xdim = f64([nap_get netcdf $mdir/$mfil lon])"
nap "$ydim = f64([nap_get netcdf $mdir/$mfil lat])"
}

if {$pack == "yes"} {
nap "add_offset = [nap_get netcdf $mdir/$mfil ${varp}[pad10 [expr $lnt+1]]:add_offset]"
nap "scale_factor = [nap_get netcdf $mdir/$mfil ${varp}[pad10 [expr $lnt+1]]:scale_factor]"
}

if {$pack == "yes"} {
nap "va = i16(reshape(0,{$newshapeA}))"
$va set missing $mv
nap "vax = reshape(i16(0),{$newshapeC})"
$vax set missing $mv
} else {
nap "va = f32(reshape(0,{$newshapeA}))"
$va set missing $mv
nap "vax = reshape(f32(0),{$newshapeC})"
$vax set missing $mv
}

if {$surface_pressure != ""} {
foreach lv $lvars {
$vert set value $lnt "$lnt"
incr lnt
};#lv
nap "$zdim = f64(vert)"
}

if {$surface_pressure != ""} {
$va set dim $ldim plev $ydim $xdim
$va set coo $ldim plev $ydim $xdim
$vax set dim $ldim level $ydim $xdim
$vax set coo $ldim level $ydim $xdim
} else {
$va set dim $ldim $zdim $ydim $xdim
$va set coo $ldim $zdim $ydim $xdim
$vax set dim $ldim $zdim $ydim $xdim
$vax set coo $ldim $zdim $ydim $xdim
```

```
}

$va netcdf -unlimited -index "time_ind,," ${tdir}/${ofil} $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

if {$pack == "yes"} {
[nap_get netcdf $mdir/$mfil $ovar:scale_factor] netcdf \
$tdir/$ofil $ovar:scale_factor
[nap_get netcdf $mdir/$mfil $ovar:add_offset] netcdf \
$tdir/$ofil $ovar:add_offset

nap "valid_min = i16(-32767)"
nap "valid_max = i16(32767)"
$valid_min netcdf $tdir/$ofil $ovar:valid_min
$valid_max netcdf $tdir/$ofil $ovar:valid_max
}

source ~IPCC/processing/tools/ipcc_att.tcl

};#tnt==1

if {$surface_pressure != ""} {
$vax set dim $ldim level $ydim $xdim
$vax set coo $ldim level $ydim $xdim
} else {
$vax set dim $ldim plev $ydim $xdim
$vax set coo $ldim plev $ydim $xdim
}

if {$mask == "yes"} {
if {$iflg != 6 && $iflg != 7 && $iflg != 13 && $iflg != 14 && $iflg != 15} {
nap "psfa = ([nap_get netcdf $tdir/psf.m[pad10 $mnow].y$ynow.nc psf ",,"])*100.0"
nap "s1 = {[expr [lindex [$psfa shape] 0]/$ntpd] $ntpd [lrange [$psfa shape] 1 2]}"
nap "psfb = am(transpose(reshape(psfa,s1),{0 2 3 1}),1)"
} else {
nap "psfa = ([nap_get netcdf $mdir/$mfil psf ",,"])*100.0"
}
nap "s1 = {[expr [lindex [$psfa shape] 0]/$ntpd] $ntpd [lrange [$psfa shape] 1 2]}"
nap "psfb = am(transpose(reshape(psfa,s1),{0 2 3 1}),1)"
};#imask

if {$surface_pressure != ""} {
nap "psa = ([nap_get netcdf $mdir/$mfil psf ",,"])*100.0"
nap "s2 = {[expr [lindex [$psfa shape] 0]/$ntpd] $ntpd [lrange [$psfa shape] 1 2]}"
nap "ps = am(transpose(reshape(psa,s2),{0 2 3 1}),1)"
$ps set dim $ldim $ydim $xdim
$ps set coo $ldim $ydim $xdim
};#surface_pressure

set lnt 0
foreach lv $lvars {
```

```
set lntp1 [expr $lnt+1]

if {$surface_pressure == ""} {
nap "v1 = fill_holesA([nap_get netcdf $tdir/$mfil $lv ",,"])"
} else {
nap "v1 = [nap_get netcdf $mdir/$mfil $lv ",,"]"
}

if {$lnt == 0} {
nap "s1 = {[expr [lindex [$v1 shape] 0]/$ntpd] $ntpd [lrange [$v1 shape] 1 2]}"
}

if {$hfine == "yes"} {
nap "v2a = am(transpose(reshape(v1,s1),{0 2 3 1}),1)"

$v2a set dim time ycoarse xcoarse
$v2a set coo time ycoarse xcoarse

nap "v2 = v2a(,@yfine,@xfine)"

} else {
nap "v2 = am(transpose(reshape(v1,s1),{0 2 3 1}),1)"
}

$vax set value $v2 ",$lnt,,"

incr lnt
};#lv

if {$surface_pressure != ""} {

set tmax [lindex [$vax shape] 0]

for {set ttt 0} {$ttt < $tmax} {incr ttt} {
nap "vb = vax($ttt,,,)"
$vb set unit $units
nap "psb = ps($ttt,,)"
nap "vc = mk3atPressure(plev/100.0,psb/100.0,vb)"
$va set value vc "$ttt,,,"
};#ttt

} else {

#there should be no interpolation as required input
#levels should match available levels.
nap "va = vax(,@plev,,)"
}

$va set missing $rmis
$va set dim $ldim plev $ydim $xdim
$va set coo $ldim plev $ydim $xdim

if {$mask == "yes"} {

nap "vd = va"
for {set ppp 0} {$ppp < [$plev nels]} {incr ppp} {
```

```
nap "vsing=vd(,$ppp,,)"
nap "psing=(psfb*0.0)+plev($ppp)"
nap "vmask = (psing>psfb) ? _ : vsing"
$va set value vmask ",$ppp,,"
};#ppp

$va set dim $ldim plev $ydim $xdim
$va set coo $ldim plev $ydim $xdim

};#mask

$va netcdf -unlimited -index "time_ind,," ${tdir}/${ofil} $ovar
$t0 netcdf -unlimited -index "time_ind" ${tdir}/${ofil} $ldim

if {$surface_pressure == ""} {
file delete -force $tdir/$mfil
};#surface_pressure

if {$mask == "yes"} {
if {$iflg != 6 && $iflg != 7 && $iflg != 13 && $iflg != 14 && $iflg != 15} {
file delete -force $tdir/psf.m[pad10 $mnow].y$ynow.nc
};#iflg
};#mask

};#mnow

incr tnt
};#ynow

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

puts "Copying $tdir/$ofil to $odir/$ofil..."
file copy -force $tdir/$ofil $odir/$ofil
file delete -force $tdir/$ofil

incr tot_tnt

};#dec

return
exit
}
```

14.18 ipcc_calc_4dmon_mk3agcm

ipcc_data_final.tcl
ipcc_calc_4dmon_mk3agcm.tcl is a Tcl-Nap script used to generate 3 dimensional atmospheric fields as a function of time defined in Table 9.

```
proc ipcc_calc_4dmon_mk3agcm {rnam ivar predir tdir ybas yoff ygap y1st ynth pdir iflg
pack surface_pressure hfine ipccout mask clobber} {
set proc_name "[info level 1]"
puts "Entering procedure $proc_name"
set time_freq "monthly"

set hostname [hostname]

#CSIRO Australia
#Mark Collier November 2003
#generate monthly time-series from monthly atmosphere files...

source ~IPCC/processing/tools/ipcc_intro.tcl

if {$ivar == "z4d" || $ivar == "w4d" } {
set fpref ""
} else {
set fpref "s"
}

if {$surface_pressure != ""} {
nap "plev=f64{100000 92500 85000 70000 60000 50000 40000 30000 25000 20000 15000 10000 7000
5000 3000 2000 1000}"
$plev set unit Pa
nap "oldp=f64{99550.0 97840.0 94580.0 89990.0 84260.0 77610.0 70230.0 62350.0 54150.0
45850.0 37650.0 29770.0 22390.0 15740.0 10010.0 5420.0 2160.0 450.0}"
nap "oldp = (2 * oldp(0) - oldp(1)) // oldp"
$oldp set unit Pa
};#surface_pressure

if {$hfine == "yes"} {
source ~IPCC/processing/tools/ipcc_mask.tcl
}

if {$hfine == "no" && $mask == "yes"} {
puts "As masking variable (psf) is on fine grid, then must"
puts "request output data on fine grid."
return
}

if {$pack == "yes"} {
set rmis -7777777
set rmis -1e34
set rmis -32768
nap "mv = i16($rmis)"
} else {
set rmis 1e20
nap "mv = f32($rmis)"
}

set tv {348 1056 1764 2496 3228 3960 4692 5436 6168 6900 7632 8364}
```

```
nap "TVA = f64{$tv}"
nap "TV = (TVA+24.0)/24.0"

set nmy 12;set nmym1 [expr $nmy-1]

set ybeg "";unset ybeg
for {set y $y1st} {$y <= $ynth} {set y [expr $y+$ygap]} {
lappend ybeg $y
}

set yend "";unset yend
foreach y $ybeg {
lappend yend [expr $y+$ygap-1]
}

puts "ybeg=$ybeg"
puts "yend=$yend"

set tot_tnt 1
for {set dec 1} {$dec <= [llength $ybeg]} {incr dec} {
set decm1 [expr $dec-1]

set nbeg [lindex $ybeg $decm1]
set nend [lindex $yend $decm1]

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

if {$nbeg > 250 && $rnam == "m20"} {
set expname "SRESA2"
} elseif {$nbeg > 260 && $rnam == "n20"} {
set expname "SRESA2"
} elseif {$nbeg > 300 && $rnam == "B35"} {
set expname "SRESA2"
}

if {$ivar == "c4d"} {
nap "lev=f64{$levs}"
$lev set unit 1
} elseif {$surface_pressure == ""} {
nap "plev=(f64{$levs})*100.0"
$plev set unit Pa
}

set odir $predir/$model_version/$expname/[lindex $table 0]/$ensnum;file mkdir $odir

if {$ipccout == "yes" } {
set cbeg [expr $ybas-$yoff+$nbeg]
set cend [expr $cbeg+$ygap-1]
set YBEG [pad10000 ${cbeg}]
set YEND [pad10000 ${cend}]
} else {
set YBEG [pad10000 ${nbeg}]
set YEND [pad10000 ${nend}]
}
```

```
if {$YBEG == $YEND} {
set YSTR "$YBEG"
} else {
set YSTR "$YBEG-$YEND"
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR}_${rnam}_${model_version}_${ensnum}.nc
}

puts "Output file=$odir/$ofil"
puts "Temporaray file=$tdir/$ofil"
if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
} else {
file delete -force ${tdir}/${ofil}
file delete -force ${odir}/${ofil}
}

set tnt 1
for {set ynow $nbeg} {$ynow <= $nend} {incr ynow} {

set tntm1 [expr $tnt-1]
set YNOW [expr $ybas-$yoff+$ynow]

source ~IPCC/processing/tools/ipcc_dinfo_mon_mk3agcm.tcl

nap "t0 = ${YNOW}*365*1 + $TV"

if { $ynow == $nbeg } {
nap "c1=f64($nbeg-1)/20"
nap "c2=f64(c1%1)"
nap "c3=i8(c2>=0.5)"
if { [$c3] == 1 } {
set ynum 11
} else {
set ynum 1
}
}

set ynumm1 [expr $ynum-1]

set month_beg [expr ($ynow-$nbeg)*12+1]
set month_end [expr $month_beg+$nmym1]
set month_begm1 [expr $month_beg-1]
set month_endm1 [expr $month_end-1]
nap "mon_ind = $month_begm1..$month_endm1"

puts "ynow=$ynow YNOW=$YNOW $month_beg $month_end $ynumm1"

if {$iflg == 1 && $ynow <= 200} {
set yfudge [expr ${ynumm1}-9]

} elseif {$iflg == 1 && $ynow <= 385} {
```

```
set yfudge $ynumm1

} elseif {$iflg == 1 && $ynow >= 386} {

if {$ynow == 386} {
set ynum 1
set ynumm1 0
} elseif {$ynow == 391} {
set ynum 6
set ynumm1 5
}

set yfudge $ynumm1

} elseif {$iflg == 8 && $ynow <= 260} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 5 && $ynow <= 260} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 15 && $ynow <= 140} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 20 && $ynow <= 180} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 22 && $ynow <= 180} {

set yfudge [expr ${ynumm1}-10]

} else {

set yfudge $ynumm1

}

if {$yfudge < 0 || $yfudge > 19} {
puts "Error with yfudge=$yfudge."
return
}

if {$tnt == 1 } {

if {$iflg == 160 || $iflg == 200 || $iflg == 210 || $iflg == 220 || $iflg == 230
|| $iflg == 240 || $iflg == 26} {
if {$ivar == "t4d"} {puts "aaa";set mfils [glob $mdir/${fpref}tnl*]}
if {$ivar == "u4d"} {set mfils [glob $mdir/${fpref}unl*]}
if {$ivar == "v4d"} {set mfils [glob $mdir/${fpref}vnl*]}
if {$ivar == "c4d"} {set mfils [glob $mdir/${fpref}Cnl*]}
} else {
set mfils ""
foreach lv $lvars {
```

```
lappend mfils [glob $mdir/${fpref}${lv}_${rnam}*]
};#lv
};#new form of pressure level data for specific runs

if {$mask == "yes"} {
if {$iflg == 16 || $iflg == 20 || $iflg == 21 || $iflg == 22 || $iflg == 23 || $iflg == 24
|| $iflg == 26} {
lappend mfils [glob $mdir/spsf_${rnam}*]
} else {
lappend mfils [glob $mdir/psf_${rnam}*]
}
}

puts "dmgetting...$mfils"
dmget_tcl $mfils

if {$mask == "yes"} {
if {$iflg == 16 || $iflg == 20 || $iflg == 21 || $iflg == 22 || $iflg == 23 || $iflg == 24
|| $iflg == 26} {
set pfil spsf_${rnam}
} else {
set pfil psf_${rnam}
}
set pfil1 [glob $mdir/${pfil}*]
set pfil2 [string range $pfil1 [expr [string last "/" $pfil1]+1] [string length $pfil1]]
puts "Input mask=$mdir/$pfil2"
}

nap "vert = f32(0..[expr [llength $lvars]-1])"

set lnt 0
foreach lv $lvars {
set mfil ${fpref}${lv}_${rnam}

set mfil1 [glob $mdir/${mfil}*]
set mfil2 [string range $mfil1 [expr [string last "/" $mfil1]+1] [string length $mfil1]]

puts "Input file=$mdir/$mfil2"

if {$tot_tnt == 1} {
catch {set land_sea_horiz_interp [[nap_get netcdf $mdir/$mfil2 $ivar:
land_sea_horiz_interp]]}
}

if {$surface_pressure != ""} {
set string [[nap "[nap_get netcdf $mdir/$mfil2 $lv:long_name]"]]

$vert set value [lindex $string $pnum] "$lnt"
nap "$zdim = f64(vert)"
};#surface_pressure

if { $lnt == 0 } {

if {$pack == "yes"} {
nap "add_offset = [nap_get netcdf $mdir/$mfil2 $lv:add_offset]"
nap "scale_factor = [nap_get netcdf $mdir/$mfil2 $lv:scale_factor]"
```

```
}

set dimen [nap_get netcdf -dimension $mdir/$mfil2 $lv]
set shape [nap_get netcdf -shape $mdir/$mfil2 $lv]
set rank [nap_get netcdf -rank $mdir/$mfil2 $lv]

if {$surface_pressure != ""} {
set newshapeA [lreplace $shape 0 1 $nmy [$plev nels]]
set newshapeC [lreplace $shape 0 1 $nmy [$oldp nels]]
} else {
set newshapeA [lreplace $shape 0 1 $nmy [llength $lvars]]
set newshapeC [lreplace $shape 0 1 $nmy [llength $lvars]]
}

if {$hfine == "yes"} {
set newshapeA [lreplace $newshapeA 2 3 [lindex $l1s 0] [lindex $l1s 1]]
set newshapeC [lreplace $newshapeC 2 3 [lindex $l1s 0] [lindex $l1s 1]]
}

if {$hfine == "yes"} {
set newshape "[lindex $shape 1] $l1s"
nap "$xdim = f64(cv(l1,1))"
nap "$ydim = f64(cv(l1,0))"
} else {
nap "$xdim = f64([nap_get netcdf $mdir/$mfil2 longitude])"
nap "$ydim = f64([nap_get netcdf $mdir/$mfil2 latitude])"
}

nap "$ldim = TV"

if {$pack == "yes"} {
nap "va = reshape(i16(0),{$newshapeA})"
$va set missing $mv
nap "vax = reshape(i16(0),{$newshapeC})"
$vax set missing $mv
} else {
nap "va = reshape(f32(0),{$newshapeA})"
$va set missing $mv
nap "vax = reshape(f32(0),{$newshapeC})"
$vax set missing $mv
}

};#lnt==0

set lnt [expr $lnt+1]
};#lv

if {$surface_pressure != ""} {
$va set dim $ldim plev $ydim $xdim
$va set coo $ldim plev $ydim $xdim
$vax set dim $ldim oldp $ydim $xdim
$vax set coo $ldim oldp $ydim $xdim
} else {
$va set dim $ldim $zdim $ydim $xdim
$va set coo $ldim $zdim $ydim $xdim
$vax set dim $ldim $zdim $ydim $xdim
```

```
$vax set coo $ldim $zdim $ydim $xdim
};#surface_pressure

$va netcdf -unlimited -index "$mon_ind,,," ${tdir}/${ofil} $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

$mv netcdf ${tdir}/$ofil $ovar:missing_value

source ~IPCC/processing/tools/ipcc_att.tcl

if {$pack == "yes"} {
[nap_get netcdf $mdir/$mfil2 $lv:scale_factor] netcdf \
$tdir/$ofil $ovar:scale_factor
[nap_get netcdf $mdir/$mfil2 $lv:add_offset] netcdf \
$tdir/$ofil $ovar:add_offset
}
};#tnt==1

if {$mask == "yes"} {
if {$iflg == 16 || $iflg == 20 || $iflg == 21 || $iflg == 22 || $iflg == 23 || $iflg == 24
|| $iflg == 26} {
set pfil spsf_${rnam}
} else {
set pfil psf_${rnam}
}
set pfil1 [glob $mdir/${pfil}*]
set pfil2 [string range $pfil1 [expr [string last "/" $pfil1]+1] [string length $pfil1]]
nap "psf = ([nap_get netcdf $mdir/$pfil2 psf "$yfudge,,,"])*100.0"
$psf set coo $ldim $ydim $xdim
$psf set dim $ldim $ydim $xdim
};#mask

set lnt 0
foreach lv $lvars {
set lntp1 [expr $lnt+1]
set mfil ${fpref}${lv}_${rnam}

set mfil1 [glob $mdir/${mfil}*]
set mfil2 [string range $mfil1 [expr [string last "/" $mfil1]+1] [string length $mfil1]]

puts "Input file=$mdir/$mfil2"

set newshapeB [lreplace $shape 0 1 $nmy 1]

if {$hfine == "yes"} {
set newshapeB [lreplace $newshapeB 2 3 [lindex $l1s 0] [lindex $l1s 1]]
}

nap "v1 = fill_holesA([nap_get netcdf $mdir/$mfil2 $lv "{$yfudge},,,"])"

$v1 set missing $rmis

nap "v2 = transpose(v1,{1 0 2 3})"
nap "v3 = v2(,,@yfine,@xfine)"
```

```
$v3 set missing $rmis
if {$surface_pressure != ""} {
$vax set value v3 ",$lntp1,,"
} else {
$vax set value v3 ",$lnt,,"
}

set lnt [expr $lnt+1]
};#lv

if {$surface_pressure != ""} {
$vax set value "2 * vax(,1,,) - vax(,2,,)" ",0,,"
}

if {$ovar == "cl"} {
nap "vc = vax"

nap "zeros = f32(vc(,17,,)*0.0)"
$vc set value $zeros ",17,,"

$vc set missing $rmis
$vc set dim $ldim $zdim $ydim $xdim
$vc set coo $ldim $zdim $ydim $xdim
} else {
nap "vc = vax(,@plev,,)"
$vc set missing $rmis
$vc set dim $ldim plev $ydim $xdim
$vc set coo $ldim plev $ydim $xdim
};#ovar

if {$mask == "yes"} {

nap "vd = vc"

for {set ppp 0} {$ppp < [$plev nels]} {incr ppp} {

nap "vsing=vd(,$ppp,,)"
nap "psing=(psf*0.0)+plev($ppp)"
nap "vmask = (psing>psf) ? _ : vsing"

$vc set value vmask ",$ppp,,"

};#ppp

$vc set dim $ldim plev $ydim $xdim
$vc set coo $ldim plev $ydim $xdim

};#mask

$vc netcdf -unlimited -index "$mon_ind,,," ${tdir}/${ofil} $ovar

$t0 netcdf -unlimited -index "$mon_ind" ${tdir}/${ofil} time

incr tnt
incr ynum
};#ynow
```

```
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

if {$ovar == "cl"} {
set ipcc_ps [regsub A1c $predir/$model_version/$expname/A1a/$ensnum/ps[string range $ofil
[string first _ $ofil] end] A1a]
nap "time = [nap_get netcdf $ipcc_ps time]"
nap "ps = [nap_get netcdf $ipcc_ps ps]"
$ps set dim time $ydim $xdim
$ps set coo time $ydim $xdim
$ps set missing
$ps set label "Surface Pressure"
$ps set unit Pa
$ps netcdf $tdir/$ofil ps
nap "standard_name = ‘surface_air_pressure‘"
$standard_name netcdf $tdir/$ofil ps:standard_name
};#ovar

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dec

return
exit
}
```

14.19 ipcc_calc_day_mk3agcm

ipcc_data_final.tcl
ipcc_calc_day_mk3agcm.tcl is a Tcl-Nap script used to generate 2 dimensional atmospheric fields as a function of time defined in Table 14.

```
proc ipcc_calc_day_mk3agcm {rnam ivar predir tdir ybas yoff ygap y1st ynth pdir iflg pack
hfine ipccout clobber time_freq} {
set proc_name "[info level 1]"
puts "Entering procedure $proc_name"

if {$time_freq == "6hourly"} {
set table_pos 4
set ntpd_out 4
} elseif {$time_freq == "12hourly"} {
set table_pos 5
set ntpd_out 2
} elseif {$time_freq == "daily"} {
set table_pos 1
set ntpd_out 1
} else {
puts "Haven't examined other possibilities."
return
}

set hostname [hostname]

source ~IPCC/processing/tools/ipcc_tspace.tcl

#note that rsus/rlus will only ever be computed from
#components that are on full grid.

#CSIRO Australia
#Mark Collier November 2003
#last modified: April 2004

source ~IPCC/processing/tools/ipcc_intro.tcl

if {$hfine == "yes"} {
source ~IPCC/processing/tools/ipcc_mask.tcl
}

if {$pack == "yes"} {
set rmis -32768
nap "mv = i16($rmis)"
} else {
set rmis -7777777
set rmis -1e34
set rmis 1e20
nap "mv = f32($rmis)"
}
set dim {31 28 31 30 31 30 31 31 30 31 30 31}

set nmy 12;set nmym1 [expr $nmy-1]

set ybeg "";unset ybeg
for {set y $y1st} {$y <= $ynth} {set y [expr $y+$ygap]} {
```

```
lappend ybeg $y
}

set yend "";unset yend
foreach y $ybeg {
lappend yend [expr $y+$ygap-1]
}

set tot_tnt 1
for {set dec 1} {$dec <= [llength $ybeg]} {incr dec} {
set decm1 [expr $dec-1]

set nbeg [lindex $ybeg $decm1]
set nend [lindex $yend $decm1]

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

#set ivar to something sensible, reset later.
if {$ivar == "rlus"} {
set ivar "rgd"
set tvar "rlus"
} elseif {$ivar == "rsus"} {
set ivar "sgd"
set tvar "rsus"
} else {
set tvar $ivar
}

if {$nbeg > 250 && $rnam == "m20"} {
set expname "SRESA2"
} elseif {$nbeg > 260 && $rnam == "n20"} {
set expname "SRESA2"
} elseif {$nbeg > 300 && $rnam == "B35"} {
set expname "SRESA2"
}

#note that tsu (histb) and tsua (hista) are available,
#possibly best to use histb where often there are
#more times per day.

if {$ivar == "psf" || $ivar == "tsu" || $ivar == "u01" || $ivar == "v01"} {
set hist "b"
} else {
set hist "a"
}

set odir $predir/$model_version/$expname/[lindex $table 1]/$ensnum;file mkdir $odir

if {$ipccout == "yes" } {
set cbeg [expr $ybas-$yoff+$nbeg]
set cend [expr $cbeg+$ygap-1]
set YBEG [pad10000 ${cbeg}]
set YEND [pad10000 ${cend}]
} else {
set YBEG [pad10000 ${nbeg}]
```

```
set YEND [pad10000 ${nend}]
}

if {$YBEG == $YEND} {
set YSTR "$YBEG"
} else {
set YSTR "$YBEG-$YEND"
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 1]_${YSTR}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 1]_${YSTR}_${rnam}_${model_version}_${ensnum}.nc
}

puts "Output file=$odir/$ofil"
puts "Temporary file=$tdir/$ofil"
if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
  if {$tvar == "rsus" && $ivar == "sgd"} {
  set ivar "rsus"
  } elseif {$tvar == "rlus" && $ivar == "rgd"} {
  set ivar "rlus"
  }
continue
} else {
file delete -force ${tdir}/${ofil}
file delete -force ${odir}/${ofil}
}

set day_beg 1
set time_beg 1

set tnt 1
for {set ynow $nbeg} {$ynow <= $nend} {incr ynow} {

source ~IPCC/processing/tools/ipcc_dinfo_hist_mk3agcm.tcl

catch {unset hfils}
for {set mnow 1} {$mnow <= $nmy} {incr mnow} {

if {$rnam == "st1"} {
  if {$ynow <= 385} {
  lappend hfils $hdir/hist$hist.m[pad10 $mnow].y$ynow.nc
  } else {
  lappend hfils $hdir/hist$hist.m[pad10 $mnow].y$ynow.$rnam.nc
  }
} elseif {$rnam == "a1b" || $rnam == "s20" || $rnam == "qv3" || $rnam == "qv4"
|| $rnam == "n20" || $rnam == "o20" || $rnam == "B35" || $rnam == "D35" || $rnam == "E35"
|| $rnam == "M35" || $rnam == "N35"} {
lappend hfils $hdir/hist$hist.m[pad10 $mnow].y$ynow.$rnam.nc
} else {
lappend hfils $hdir/hist$hist.m[pad10 $mnow].y$ynow.nc
}

};#mnow
```

```
puts $hfils
dmget_tcl $hdir/$hfils

for {set mnow 1} {$mnow <= $nmy} {incr mnow} {
set mnowm1 [expr $mnow-1]

puts "nbeg=$nbeg ynow=$ynow mnow=$mnow tnt=$tnt"

if {$rnam == "st1"} {
  if {$ynow <= 385} {
  set hfil hist$hist.m[pad10 $mnow].y$ynow.nc
  } else {
  set hfil hist$hist.m[pad10 $mnow].y$ynow.$rnam.nc
  }
} elseif {$rnam == "a1b" || $rnam == "s20" || $rnam == "qv3" || $rnam == "qv4"
|| $rnam == "n20" || $rnam == "o20" || $rnam == "B35" || $rnam == "D35" || $rnam == "E35"
|| $rnam == "M35" || $rnam == "N35"} {
set hfil hist$hist.m[pad10 $mnow].y$ynow.$rnam.nc
} elseif {$rnam == "qv1" || $rnam == "qv2"} {

set hfil hist$hist[pad10000 $ynow][pad10 $mnow].$rnam.nc

} else {
set hfil hist$hist.m[pad10 $mnow].y$ynow.nc
}

puts "Input file=$hdir/$hfil"

if {$mnow == 1 && $ynow == $nbeg} {
catch {set land_sea_horiz_interp [[nap_get netcdf $hdir/$hfil $ivar:land_sea_horiz_interp]]}
}

set tntm1 [expr $tnt-1]
set YNOW [expr $ybas-$yoff+$ynow]
set NBEG [expr $ybas-$yoff+$nbeg]

if {$hist == "b"} {
nap "$ldim = f64([nap_get netcdf $hdir/$hfil time])"
nap "time = am(transpose(reshape(time,{[expr [$time nels]/2] 2}),{1 0}))"
} else {
nap "$ldim = f64([nap_get netcdf $hdir/$hfil time])"
}

set day_end [expr $day_beg+[$time nels]-1]
set day_begm1 [expr $day_beg-1]
set day_endm1 [expr $day_end-1]

set time_end [expr $time_beg+[$time nels]-1]
set time_begm1 [expr $time_beg-1]
set time_endm1 [expr $time_end-1]

nap "time_ind = $time_begm1 .. $time_endm1"
set time_beg [expr $time_end+1]

nap "day_ind = $day_begm1..[expr $day_end-$tincre]...$tincre"
nap "t0 = (time*0+${NBEG}*365*1)+day_ind+0.5"
```

```
set day_beg [expr $day_end+1]

set dimen [nap_get netcdf -dimension $hdir/$hfil $ivar]
set shape [nap_get netcdf -shape $hdir/$hfil $ivar]
set rank [nap_get netcdf -rank $hdir/$hfil $ivar]
set newshape [lreplace $shape 0 0 [expr [lindex $shape 0]/2] 2]

if {$tnt == 1} {

if {$hist == "b"} {
  if {$ovar == "rlus"} {
  nap "v1a = am(transpose(reshape([nap_get netcdf $hdir/$hfil rgd ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1b = am(transpose(reshape([nap_get netcdf $hdir/$hfil rgn ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1 = v1a+v1b"
  } elseif {$ovar == "rsus"} {
  nap "v1a = am(transpose(reshape([nap_get netcdf $hdir/$hfil sgd ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1b = am(transpose(reshape([nap_get netcdf $hdir/$hfil sgn ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1 = v1a-v1b"
  } else {
  nap "v1 = am(transpose(reshape([nap_get netcdf $hdir/$hfil $ivar ",,"],{$newshape}),
{0 2 3 1}),1)"
  }
} else {
  if {$ovar == "rlus"} {
  nap "v1a = [nap_get netcdf $hdir/$hfil rgd ",,"]"
  nap "v1b = [nap_get netcdf $hdir/$hfil rgn ",,"]"
  nap "v1 = v1a+v1b"

nap "r1 = range(v1)"

  } elseif {$ovar == "rsus"} {
  nap "v1a = [nap_get netcdf $hdir/$hfil sgd ",,"]"
  nap "v1b = [nap_get netcdf $hdir/$hfil sgn ",,"]"
  nap "v1 = v1a-v1b"
  } else {
  nap "v1 = [nap_get netcdf $hdir/$hfil $ivar ",,"]"
  }
}

if {$hfine == "yes"} {
nap "$xdim = f64(xfine)"
nap "$ydim = f64(yfine)"
nap "xcoarse = f64([nap_get netcdf $hdir/$hfil lon])"
nap "ycoarse = f64([nap_get netcdf $hdir/$hfil lat])"
} else {
nap "$xdim = f64([nap_get netcdf $hdir/$hfil lon])"
nap "$ydim = f64([nap_get netcdf $hdir/$hfil lat])"
}

set v1s [$v1 shape]
set v1x [lindex $v1s 2]
```

```
set v1y [lindex $v1s 1]

$v1 set dim time ycoarse xcoarse
$v1 set coo time ycoarse xcoarse

if {$v1x == 96 && $v1y == 48} {
  nap "v1_full = v1(,@yfine,@xfine)"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full = v1"
} else {
puts "Don't know that resolution."
return
}

if {$pack == "yes"} {
nap "add_offset = [nap_get netcdf $hdir/$hfil $ivar:add_offset]"
nap "scale_factor = [nap_get netcdf $hdir/$hfil $ivar:scale_factor]"
}

nap "$ldim = f64([nap_get netcdf $hdir/$hfil time])"

if {$pack == "yes"} {
nap "va = i16(v1_full)"
$va set missing $mv
} else {
nap "va = f32(v1_full)"
$va set missing $mv
}

$va set dim $ldim $ydim $xdim
$va set coo $ldim $ydim $xdim

$va netcdf -unlimited -index "time_ind,," ${tdir}/${ofil} $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

if {$pack == "yes"} {
[nap_get netcdf $hdir/$hfil $ovar:scale_factor] netcdf \
$tdir/$ofil $ovar:scale_factor
[nap_get netcdf $hdir/$hfil $ovar:add_offset] netcdf \
$tdir/$ofil $ovar:add_offset

nap "valid_min = i16(-32767)"
nap "valid_max = i16(32767)"
$valid_min netcdf $tdir/$ofil $ovar:valid_min
$valid_max netcdf $tdir/$ofil $ovar:valid_max
}

source ~IPCC/processing/tools/ipcc_att.tcl

##tnt==1
}
```

```
if {$hist == "b"} {
  if {$ovar == "rlus"} {
  nap "v1a = am(transpose(reshape([nap_get netcdf $hdir/$hfil rgd ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1b = am(transpose(reshape([nap_get netcdf $hdir/$hfil rgn ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1 = v1a+v1b"
  } elseif {$ovar == "rsus"} {
  nap "v1a = am(transpose(reshape([nap_get netcdf $hdir/$hfil sgd ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1b = am(transpose(reshape([nap_get netcdf $hdir/$hfil sgn ",,"],{$newshape}),
{0 2 3 1}),1)"
  nap "v1 = v1a-v1b"
  } else {
  nap "v1 = am(transpose(reshape([nap_get netcdf $hdir/$hfil $ivar ",,"],{$newshape}),
{0 2 3 1}),1)"
  }
} else {
  if {$ovar == "rlus"} {
  nap "v1a = [nap_get netcdf $hdir/$hfil rgd ",,"]"
  nap "v1b = [nap_get netcdf $hdir/$hfil rgn ",,"]"
  nap "v1 = v1a+v1b"
  } elseif {$ovar == "rsus"} {
  nap "v1a = [nap_get netcdf $hdir/$hfil sgd ",,"]"
  nap "v1b = [nap_get netcdf $hdir/$hfil sgn ",,"]"
  nap "v1 = v1a-v1b"
  } else {
  nap "v1 = [nap_get netcdf $hdir/$hfil $ivar ",,"]"
  }
}

$v1 set dim time ycoarse xcoarse
$v1 set coo time ycoarse xcoarse

if {$v1x == 96 && $v1y == 48} {
  if {$land_sea_interp == "yes"} {
  nap "v1l = fill_holesA(v1*l2,12)"
  nap "v1o = fill_holesA(v1*o2,12)"
  nap "v1_full = (v1l(,@yfine,@xfine)*l1p)+(v1o(,@yfine,@xfine)*o1p)"
  } else {
  nap "v1_full = v1(,@yfine,@xfine)"
  }

} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full = v1"
} else {
puts "Don't know that resolution."
return
}

nap "va = v1_full*$exter_scale-$exter_offset"

$va set dim $ldim $ydim $xdim
$va set coo $ldim $ydim $xdim

$va netcdf -unlimited -index "time_ind,," ${tdir}/${ofil} $ovar
```

```
$t0 netcdf -unlimited -index "time_ind" ${tdir}/${ofil} $ldim

}

set tnt [expr $tnt+1]
}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

puts "Copying $tdir/$ofil to $odir/$ofil..."
file copy -force $tdir/$ofil $odir/$ofil
file delete -force $tdir/$ofil

set tot_tnt [expr $tot_tnt+1]

if {$tvar == "rsus" && $ivar == "sgd"} {
set ivar "rsus"
} elseif {$tvar == "rlus" && $ivar == "rgd"} {
set ivar "rlus"
}

}

return
exit
}
```

14.20 ipcc_calc_mon_mk3agcm

ipcc_data_final.tcl
ipcc_calc_mon_mk3agcm.tcl is a Tcl-Nap script used to generate 2 dimensional atmospheric fields as a function of time defined in Tables 5- 7.

```
proc ipcc_calc_mon_mk3agcm {rnam ivar predir tdir ybas yoff ygap y1st ynth pdir iflg pack
hfine ipccout clobber} {
set proc_name "[info level 1]"
puts "Entering procedure $proc_name"
set time_freq "monthly"

catch {unset land_sea_horiz_interp}

set hostname [hostname]

#CSIRO Australia
#Mark Collier November 2003
#last modified: April 2004
source ~IPCC/processing/tools/ipcc_intro.tcl

if {$hfine == "yes"} {
source ~IPCC/processing/tools/ipcc_mask.tcl
}

if {$pack == "yes"} {
set rmis -32768
nap "mv = i16($rmis)"
} else {
set rmis -7777777
set rmis -1e34
set rmis 1e20
nap "mv = f32($rmis)"
}
set tv {348 1056 1764 2496 3228 3960 4692 5436 6168 6900 7632 8364}
nap "TVA = f64{$tv}"
nap "TV = (TVA+24.0)/24.0"

set nmy 12;set nmym1 [expr $nmy-1]

catch {unset ybeg}
for {set y $y1st} {$y <= $ynth} {set y [expr $y+$ygap]} {
lappend ybeg $y
}

catch {unset yend}
foreach y $ybeg {
lappend yend [expr $y+$ygap-1]
}

puts "ybeg=$ybeg"
puts "yend=$yend"

set tot_tnt 1
for {set dec 1} {$dec <= [llength $ybeg]} {incr dec} {
set decm1 [expr $dec-1]
```

```
set nbeg [lindex $ybeg $decm1]
set nend [lindex $yend $decm1]

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

if {$nbeg > 250 && $rnam == "m20"} {
set expname "SRESA2"
} elseif {$nbeg > 260 && $rnam == "n20"} {
set expname "SRESA2"
} elseif {$nbeg > 300 && $rnam == "B35"} {
set expname "SRESA2"
}

set odir $predir/$model_version/$expname/[lindex $table 0]/$ensnum;file mkdir $odir

if {$ivar == "xxx"} {
set tvar snd
set mfil s${tvar}_${rnam}
} elseif {$ivar == "uas" || $ivar == "vas" || $ivar == "huss" || $ivar == "mrros"
|| $ivar == "ps" || $ivar == "rtmt" || $ivar == "rlus" || $ivar == "rsus"
|| $ivar == "pev" || $ivar == "V10" || $ivar == "rsdt" || $ivar == "rsdscs"
|| $ivar == "rsuscs" || $ivar == "rldscs" || $ivar == "snc" || $ivar == "snw"} {
set mfil ${ivar}_${rnam}
set tvar $ivar
} elseif {$ivar == "suas" } {
#only relevant for qv3/qv4/n20 currently
#for cases where these are calculated directly by model
set mfil ${ivar}_${rnam}
set tvar "uas"
  if {$time_freq == "monthly"} {
  set var_derivation "Computed from monthly values. Note that for this experiment monthly
uas is computed DIRECTLY and EXACTLY by the model, unlike the earlier ones"
     }
} elseif {$ivar == "svas" } {
#only relevant for qv3/qv4/n20 currently
set mfil ${ivar}_${rnam}
set tvar "vas"
  if {$time_freq == "monthly"} {
  set var_derivation "Computed from monthly values. Note that for this experiment monthly
vas is computed DIRECTLY and EXACTLY by the model, unlike the earlier ones"
     }
} elseif {$ivar == "shus" } {
#only relevant for qv3/qv4/n20 currently
set mfil ${ivar}_${rnam}
set tvar "hus"
  if {$time_freq == "monthly"} {
  set var_derivation "Computed from monthly values. Note that for this experiment monthly
hus is computed DIRECTLY and EXACTLY by the model, unlike the earlier ones"
     }
} else {
set mfil s${ivar}_${rnam}
set tvar $ivar
}

if {$ivar == "psf"} {
```

```
#only relevant for qv3/qv4 currently
  if {$time_freq == "monthly"} {
  set var_derivation "Computed from monthly values. Note that for this experiment monthly
ps is computed DIRECTLY and EXACTLY by the model, unlike the earlier ones"
}
}

puts "Input file=${mfil}.nc"

if {$ipccout == "yes" } {
set cbeg [expr $ybas-$yoff+$nbeg]
set cend [expr $cbeg+$ygap-1]
set YBEG [pad10000 ${cbeg}]
set YEND [pad10000 ${cend}]
} else {
set YBEG [pad10000 ${nbeg}]
set YEND [pad10000 ${nend}]
}

if {$YBEG == $YEND} {
set YSTR "$YBEG"
} else {
set YSTR "$YBEG-$YEND"
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR}_${rnam}_${model_version}_${ensnum}.nc
}

puts "Output file=$odir/$ofil"
puts "Temporary file=$tdir/$ofil"
if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
} else {
file delete -force ${tdir}/${ofil}
file delete -force ${odir}/${ofil}
}

set tnt 1
for {set ynow $nbeg} {$ynow <= $nend} {incr ynow} {

set tntm1 [expr $tnt-1]
set YNOW [expr $ybas-$yoff+$ynow]

source ~IPCC/processing/tools/ipcc_dinfo_mon_mk3agcm.tcl

puts $mdir

nap "t0 = ${YNOW}*365*1 + $TV"

if { $ynow == $nbeg } {
nap "c1=f64($nbeg-1)/20"
nap "c2=f64(c1%1)"
nap "c3=i8(c2>=0.5)"
```

```
if { [$c3] == 1 } {
set ynum 11
} else {
set ynum 1
}
}

set ynumm1 [expr $ynum-1]

set month_beg [expr ($ynow-$nbeg)*12+1]
set month_end [expr $month_beg+$nmym1]
set month_begm1 [expr $month_beg-1]
set month_endm1 [expr $month_end-1]
nap "mon_ind = $month_begm1..$month_endm1"

if {$iflg == 1 && $ynow <= 200} {
set yfudge [expr ${ynumm1}-9]

} elseif {$iflg == 1 && $ynow <= 385} {
set yfudge $ynumm1

} elseif {$iflg == 1 && $ynow >= 386} {

  if {$ynow == 386} {
  set ynum 1
  set ynumm1 0
  } elseif {$ynow == 391} {
  set ynum 6
  set ynumm1 5
  }

set yfudge $ynumm1

} elseif {$iflg == 8 && $ynow <= 260} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 5 && $ynow <= 260} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 15 && $ynow <= 140} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 20 && $ynow <= 180} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 22 && $ynow <= 180} {

set yfudge [expr ${ynumm1}-10]

} elseif {$iflg == 27 && $ynow <= 200} {

set yfudge [expr ${ynumm1}-10]
```

```
} elseif {$iflg == 28 && $ynow <= 220} {

set yfudge [expr ${ynumm1}-10]

} else {

set yfudge $ynumm1

}

if {$yfudge < 0 || $yfudge > 19} {
puts "Error with yfudge=$yfudge."
return
}

puts "yfudge=$yfudge"
puts ""

set mfil1 [glob $mdir/${mfil}*]
set mfil2 [string range $mfil1 [expr [string last "/" $mfil1]+1] [string length $mfil1]]

if {$tot_tnt == 1} {
catch {set land_sea_horiz_interp [[nap_get netcdf $mdir/$mfil2 $tvar:land_sea_horiz_interp]]}
}

if {$tnt == 1} {
nap "v1 = [nap_get netcdf $mdir/$mfil2 $tvar "{$yfudge}[commas 3]"]"

catch {nap "add_offset = [nap_get netcdf $mdir/$mfil2 $tvar:add_offset]"}
catch {nap "scale_factor = [nap_get netcdf $mdir/$mfil2 $tvar:scale_factor]"}

set dimen [nap_get netcdf -dimension $mdir/$mfil2 $tvar]
set shape [nap_get netcdf -shape $mdir/$mfil2 $tvar]
set rank [nap_get netcdf -rank $mdir/$mfil2 $tvar]

if {$hfine == "yes"} {
set newshape "[lindex $shape 1] $l1s"
nap "$xdim = f64(cv(l1,1))"
nap "$ydim = f64(cv(l1,0))"

if {[lindex $shape 2] != [$yfine shape] || [lindex $shape 3] != [$xfine shape]} {
set land_sea_horiz_interp "$land_sea_interp,$tvar,[lindex $shape 3],[lindex $shape 2]"
}

} else {
set newshape [lrange $shape 1 3]
nap "$xdim = f64([nap_get netcdf $mdir/$mfil2 longitude])"
nap "$ydim = f64([nap_get netcdf $mdir/$mfil2 latitude])"
}

nap "$ldim = TV"

if {$pack == "yes"} {
nap "va = i16(reshape(0,{$newshape}))"
$va set missing $mv
```

```
} else {
nap "va = f32(reshape(0,{$newshape}))"
$va set missing $mv
}

$va set dim $ldim $ydim $xdim
$va set coo $ldim $ydim $xdim

$va netcdf -unlimited -index "$mon_ind,," ${tdir}/${ofil} $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

if {$pack == "yes"} {
[nap_get netcdf $mdir/$mfil2 $ovar:scale_factor] netcdf \
$tdir/$ofil $ovar:scale_factor
[nap_get netcdf $mdir/$mfil2 $ovar:add_offset] netcdf \
$tdir/$ofil $ovar:add_offset

nap "valid_min = i16(-32767)"
nap "valid_max = i16(32767)"
$valid_min netcdf $tdir/$ofil $ovar:valid_min
$valid_max netcdf $tdir/$ofil $ovar:valid_max
}

source ~IPCC/processing/tools/ipcc_att.tcl

};#tnt==1

nap "v1 = [nap_get netcdf $mdir/$mfil2 $tvar "${yfudge}[commas 3]"]"

if {$hfine == "yes"} {

if {$land_sea_interp == "yes"} {
set v1s [$v1 shape]
set v1x [lindex $v1s 2]
set v1y [lindex $v1s 1]

if {$v1x == 64 && $v1y == 32} {
  nap "v1l = fill_holesA(v1*l3,8)"
  nap "v1o = fill_holesA(v1*o3,8)"
  nap "va = f32( ((v1l(,@yfine,@xfine)*l1p)+(v1o(,@yfine,@xfine)*o1p))*
$exter_scale-$exter_offset)"
} elseif {$v1x == 96 && $v1y == 48} {
  nap "v1l = fill_holesA(v1*l2,12)"
  nap "v1o = fill_holesA(v1*o2,12)"
  nap "va = f32( ((v1l(,@yfine,@xfine)*l1p)+(v1o(,@yfine,@xfine)*o1p))*
$exter_scale-$exter_offset)"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "va = f32(v1*$exter_scale-$exter_offset)"
} else {
puts "Don't know that resolution."
return
}
```

```
} else {

nap "va = f32(v1(,@yfine,@xfine))*$exter_scale-$exter_offset"
};#land_sea_interp

} else {
nap "va = reshape(v1,{$newshape})*$exter_scale-$exter_offset"
};#hfine

if {$ovar == "huss" } {
#b/c of packing precision problem, shus output file sometimes
#goes slightly negative. Here we replace them with 0. This
#may not affect all cases, including where monthly huss is
#computed from daily files, however, it can't hurt it.
nap "va = va >>> 0"
}

$va set dim $ldim $ydim $xdim
$va set coo $ldim $ydim $xdim
$va netcdf -unlimited -index "$mon_ind,," ${tdir}/${ofil} $ovar
$t0 netcdf -unlimited -index "$mon_ind" ${tdir}/${ofil} $ldim

incr tnt
incr ynum
};#ynow

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

puts "Copying $tdir/$ofil to $odir/$ofil..."
file copy -force $tdir/$ofil $odir/$ofil;file delete -force $tdir/$ofil

incr tot_tnt

};#dec

return
exit
}
```

14.21 ipcc_calc_mon_mk3ogcm

ipcc_data_final.tcl
ipcc_calc_mon_mk3ogcm.tcl is a Tcl-Nap script used to generate 2 and 3-dimensional oceanographic fields as a function of time defined in Table 12.

```
proc ipcc_calc_mon_mk3ogcm {rnam ivar predir tdir ybas yoff ygap y1st ynth pdir pack subset
offset scale iflg llev ipccout clobber} {

set proc_name "ipcc_calc_mon_mk3ogcm"
puts "Entering procedure $proc_name"
set time_freq "monthly"

set hostname [hostname]

set fix "yes"

if {$fix == "yes"} {
set ffix [glob ~gor079/cshxr/NEWoc.3.dir/MOM2/avg.dir/avg.m12.y5610.nc]
}

if {$ivar == "uedd" || $ivar == "vedd" || $ivar == "wedd"} {
set eddy "_eddy"
} else {
set eddy ""
}

#CSIRO Australia
#Mark Collier November 2003

nap "full_lat_ind = 0 .. 188";nap "full_lon_ind = 1 .. 192"
set del_items {_FillValue valid_range}

source ~IPCC/processing/tools/ipcc_intro.tcl

if {$pack != "yes" && $pack != "no" } {
puts "pack must be yes or no"
return
}

nap "mi = 1e99";nap "ma = -1e99"

if {$ivar == "sst" || $ivar == "sss"} {
set llev {1 1}
};#tos=sst sos=sss

set lbeg [lindex $llev 0];set lend [lindex $llev 1];set lbegm1 [expr $lbeg-1];
set lendm1 [expr $lend-1]

nap "lev_ind=$lbegm1..$lendm1"

set rmis 1e20
nap "mv = f32($rmis)"

set tv {348 1056 1764 2496 3228 3960 4692 5436 6168 6900 7632 8364}

set nmy 12;set nmym1 [expr $nmy-1]
```

```
set ybeg "";unset ybeg
for {set y $y1st} {$y <= $ynth} {set y [expr $y+$ygap]} {
lappend ybeg $y
}

set yend "";unset yend
foreach y $ybeg {
lappend yend [expr $y+$ygap-1]
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

if {$ovar == "tos"} {
set ivar Temp
} elseif {$ovar == "sos"} {
set ivar Salt
}

set tot_tnt 1
for {set dec 1} {$dec <= [llength $ybeg]} {incr dec} {
set decm1 [expr $dec-1]

set nbeg [lindex $ybeg $decm1]
set nend [lindex $yend $decm1]

if {$nbeg > 250 && $rnam == "m20"} {
set expname "SRESA2"
} elseif {$nbeg > 260 && $rnam == "n20"} {
set expname "SRESA2"
} elseif {$nbeg > 300 && $rnam == "B35"} {
set expname "SRESA2"
}

set odir $predir/$model_version/$expname/[lindex $table 0]/$ensnum;file mkdir $odir

set ynow $nbeg
source ~IPCC/processing/tools/ipcc_dinfo_mon_mk3ogcm.tcl

set shape [nap_get netcdf -shape $mdir/avg${eddy}.m01.y$ynow.nc $ivar]

if {[llength $shape] == 3 || [lindex $shape 0] == [lindex $shape 1]
|| ($lbeg == 1 && $lend == 31) || $ovar == "tos" || $ovar == "sos"} {
set lstr ""
} elseif { $lbeg == $lend } {
set lstr "l$lbeg"
} else {
set lstr "l$lbeg-$lend"
}

if {$ipccout == "yes" } {
set cbeg [expr $ybas-$yoff+$ynow]
set cend [expr $cbeg+$ygap-1]
set YBEG [pad10000 ${cbeg}]
set YEND [pad10000 ${cend}]
```

```
} else {
set YBEG [pad10000 ${nbeg}]
set YEND [pad10000 ${nend}]
}

if {$YBEG == $YEND} {
set YSTR "$YBEG"
} else {
set YSTR "$YBEG-$YEND"
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]${lstr}_${YSTR}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]${lstr}_${YSTR}_${rnam}_${model_version}_${ensnum}.nc
}

puts "Output file=$odir/$ofil"
puts "Temporary file=$tdir/$ofil"
if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
} else {
file delete -force ${tdir}/${ofil}
file delete -force ${odir}/${ofil}
}

nap "t0 = f64(0..[expr (($nend-$nbeg+1)*$nmy)-1])"

set tnt 1
for {set ynow $nbeg} {$ynow <= $nend} {incr ynow} {

set YNOW [expr $ybas-$yoff+$ynow]

source ~IPCC/processing/tools/ipcc_dinfo_mon_mk3ogcm.tcl

set files "";
for {set m 1} {$m <= $nmy} {incr m} {
lappend files $mdir/avg${eddy}.m[pad10 $m].y${ynow}.nc
}

dmget_tcl $files

for {set mnow 1} {$mnow <= $nmy} {incr mnow} {
set mnowm1 [expr $mnow-1]
set tntm1 [expr $tnt-1]
set tot_tntm1 [expr $tot_tnt-1]

if { $mnow < 10 } {
set pmnow "0$mnow"
} else {
set pmnow "$mnow"
}

set mfil avg${eddy}.m${pmnow}.y${ynow}.nc

puts stdout "Input file=$mdir/$mfil"
```

```
if {$tot_tnt == 1} {
catch {set land_sea_horiz_interp [[nap_get netcdf $mdir/$mfil $ivar:land_sea_horiz_interp]]}
}

nap "Time_tmp = ${YNOW}*365 + [expr [expr [lindex $tv $mnowm1]+24.0]/24.0]"

$t0 set value "Time_tmp" "$tntm1"

if { $tnt == 1 } {

nap "FV = [nap_get netcdf $mdir/$mfil $ivar:_FillValue]"

set tfull_lat_shape "[$full_lat_ind shape]"
set tfull_lon_shape "[$full_lon_ind shape]"

set dimen [nap_get netcdf -dimension $mdir/$mfil $ivar]
set shape [nap_get netcdf -shape $mdir/$mfil $ivar]
set rank [nap_get netcdf -rank $mdir/$mfil $ivar]

if {[lindex $shape end] != $tfull_lon_shape} {
set shape [lreplace $shape end end $tfull_lon_shape]
}

if {[lindex $shape end-1] != $tfull_lat_shape} {
set shape [lreplace $shape end-1 end-1 $tfull_lat_shape]
}

#now ovar called tos and not Temp
if {$ovar == "tos" || $ovar == "sos"} {
set shape [lreplace $shape 1 1 1]
}

if { $subset == "yes" } {

nap "v1 = [nap_get netcdf $mdir/$mfil $ivar ",,$lat_ind,$lon_ind"]"
} elseif { $rank == 4 && [lindex $shape 1] > 1 } {

set newshape [lreplace $shape 1 1 $lend]
set d1 [lindex $dimen 0]
set d2 [lindex $dimen 1]
set d3 [lindex $dimen 2]
set d4 [lindex $dimen 3]
nap "va = [nap_get netcdf $mdir/$mfil $ivar ",lev_ind,full_lat_ind,full_lon_ind"]"

nap "time = f64([nap_get netcdf $mdir/$mfil $d1])"
nap "depth = f64([nap_get netcdf $mdir/$mfil $d2])"
if {$fix == "yes"} {
nap "lat = f64([nap_get netcdf $ffix $d3 "full_lat_ind"])"
} else {
nap "lat = f64([nap_get netcdf $mdir/$mfil $d3 "full_lat_ind"])"
}
nap "lon = f64([nap_get netcdf $mdir/$mfil $d4 "full_lon_ind"])"

$va set dim $ldim $zdim $ydim $xdim
$va set coo $ldim $zdim $ydim $xdim
```

```
nap "mask1 = f32(va*0)+reshape(FV,{[$va shape]})"
nap "mask2 = f32((va == mask1)*$rmis)"
$mask2 set missing $rmis
nap "v1 = va+mask2"
$v1 set missing $rmis

} elseif { $rank == 4 && [lindex $shape 1] == 1 } {

set d1 [lindex $dimen 0]
set d2 [lindex $dimen 2]
set d3 [lindex $dimen 3]

nap "time = f64([nap_get netcdf $mdir/$mfil $d1])"
if {$fix == "yes"} {
nap "lat = f64([nap_get netcdf $ffix $d2 "full_lat_ind"])"
} else {
nap "lat = f64([nap_get netcdf $mdir/$mfil $d2 "full_lat_ind"])"
}
nap "lon = f64([nap_get netcdf $mdir/$mfil $d3 "full_lon_ind"])"

nap "v1 = reshape([nap_get netcdf $mdir/$mfil $ivar],{[lindex $shape 0] [lindex $shape 2]
[lindex $shape 3]})"

$v1 set dim $ldim $ydim $xdim
$v1 set coo $ldim $ydim $xdim
$v1 set missing $rmis

} else {
set d1 [lindex $dimen 0]
set d2 [lindex $dimen 1]
set d3 [lindex $dimen 2]

nap "time = f64([nap_get netcdf $mdir/$mfil $d1])"
if {$fix == "yes"} {
nap "lat = f64([nap_get netcdf $ffix $d2 "full_lat_ind"])"
} else {
nap "lat = f64([nap_get netcdf $mdir/$mfil $d2 "full_lat_ind"])"
}
nap "lon = f64([nap_get netcdf $mdir/$mfil $d3 "full_lon_ind"])"

nap "v1 = [nap_get netcdf $mdir/$mfil $ivar ",full_lat_ind,full_lon_ind"]"

$v1 set dim $ldim $ydim $xdim
$v1 set coo $ldim $ydim $xdim
$v1 set missing $rmis
}

if {$pack == "yes" } {
nap "v1s = i16(v1)"
$v1s netcdf -unlimited ${tdir}/${ofil} $ovar
} else {
$v1 netcdf -unlimited ${tdir}/${ofil} $ovar
}

if {$pack == "yes" } {
```

```
nap "scale_factor = ${scale}f"
nap "add_offset = ${offset}f"
$scale_factor netcdf ${tdir}/$ofil $ovar:scale_factor
$add_offset netcdf ${tdir}/$ofil $ovar:add_offset
nap "valid_min = i16(-32767)"
nap "valid_max = i16(32767)"
$valid_min netcdf $tdir/$ofil $ovar:valid_min
$valid_max netcdf $tdir/$ofil $ovar:valid_max
}

source ~IPCC/processing/tools/ipcc_dims.tcl

source ~IPCC/processing/tools/ipcc_att.tcl

};#tnt=1

if { $subset == "yes" } {
nap "v1 = ([nap_get netcdf $mdir/$mfil $ivar ",,$lat_ind,$lon_ind"])*
$exter_scale-$exter_offset"
} elseif { $rank == 4 && [lindex $shape 1] > 1 } {
set newshape [lreplace $shape 1 1 $lend]
set d1 [lindex $dimen 0]
set d2 [lindex $dimen 1]
set d3 [lindex $dimen 2]
set d4 [lindex $dimen 3]

nap "time = f64([nap_get netcdf $mdir/$mfil $d1])"
nap "depth = f64([nap_get netcdf $mdir/$mfil $d2])"
if {$fix == "yes"} {
nap "lat = f64([nap_get netcdf $ffix $d3 "full_lat_ind"])"
} else {
nap "lat = f64([nap_get netcdf $mdir/$mfil $d3 "full_lat_ind"])"
}
nap "lon = f64([nap_get netcdf $mdir/$mfil $d4 "full_lon_ind"])"

nap "va = ([nap_get netcdf $mdir/$mfil $ivar ",lev_ind,full_lat_ind,full_lon_ind"])*
$exter_scale-$exter_offset"

$va set dim $ldim $zdim $ydim $xdim
$va set coo $ldim $zdim $ydim $xdim

nap "mask1 = f32(va*0)+reshape(FV,{[$va shape]})"
nap "mask2 = f32((va == mask1)*$rmis)"
$mask2 set missing $rmis
nap "v1 = va+mask2"
$v1 set missing $rmis
} elseif { $rank == 4 && [lindex $shape 1] == 1 } {

set d1 [lindex $dimen 0]
set d2 [lindex $dimen 2]
set d3 [lindex $dimen 3]

nap "time = f64([nap_get netcdf $mdir/$mfil $d1])"

if {$fix == "yes"} {
nap "lat = f64([nap_get netcdf $ffix $d2 "full_lat_ind"])"
```

```
} else {
nap "lat = f64([nap_get netcdf $mdir/$mfil $d2 "full_lat_ind"])"
}

nap "lon = f64([nap_get netcdf $mdir/$mfil $d3 "full_lon_ind"])"

if {$ovar == "tos" || $ovar == "sos"} {
nap "v1 = reshape([nap_get netcdf $mdir/$mfil $ivar ",lev_ind,full_lat_ind,full_lon_ind"],
{[lindex $shape 0] [lindex $shape 2] [lindex $shape 3]})*$exter_scale-$exter_offset"
} else {
nap "v1 = reshape([nap_get netcdf $mdir/$mfil $ivar ",,full_lat_ind,full_lon_ind"],
{[lindex $shape 0] [lindex $shape 2] [lindex $shape 3]})*$exter_scale-$exter_offset"
}

$v1 set dim $ldim $ydim $xdim
$v1 set coo $ldim $ydim $xdim
$v1 set missing $rmis

} else {
set d1 [lindex $dimen 0]
set d2 [lindex $dimen 1]
set d3 [lindex $dimen 2]
nap "time = f64([nap_get netcdf $mdir/$mfil $d1])"
if {$fix == "yes"} {
nap "lat = f64([nap_get netcdf $ffix $d2 "full_lat_ind"])"
} else {
nap "lat = f64([nap_get netcdf $mdir/$mfil $d2 "full_lat_ind"])"
}
nap "lon = f64([nap_get netcdf $mdir/$mfil $d3 "full_lon_ind"])"

$v1 set dim $ldim $ydim $xdim
$v1 set coo $ldim $ydim $xdim

nap "v1 = ([nap_get netcdf $mdir/$mfil $ivar ",full_lat_ind,full_lon_ind"])*$exter_scale-
$exter_offset"
}

nap "r = range(v1)"
nap "mi = min(mi//r(0))"
nap "ma = max(ma//r(1))"

$v1 netcdf -unlimited -index "$tntm1[commas [expr [$v1 rank]-1]]" $tdir/$ofil $ovar

incr tnt
incr tot_tnt
};#mnow

};#ynow

$t0 netcdf -unlimited ${tdir}/${ofil} time

[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf $tdir/$ofil $ovar:missing_value

source ~IPCC/processing/tools/ipcc_bnds_ogcm.tcl

puts "Copying $tdir/$ofil to $odir/$ofil..."
```

```
file rename -force  $tdir/$ofil $odir/$ofil

};#dec

if {$pack == "yes" } {
axpc2 [$mi] [$ma] -32750.0 32750.0 scale offset
puts "Use scale factor=$scale"
puts "    offset      =$offset"
}

return
exit
};#
```

14.22 ipcc_cmv

ipcc_calc_4dmon_mk3agcm.tcl, ipcc_calc_day2mon.tcl, ipcc_calc_day2mon_wap.tcl, ipcc_calc_day_mk3agcm.tcl, ipc
ipcc_cmv.tcl is a Tcl-Nap script (procedure) that serves as a key lookup database for defining variable names, units, tables, dimensions and other significant input and output parameter characteristics.

```
proc ipcc_cmv {ivar lon lat ver tim lva pnu lev stn lgn uni ov tid esca eoff lsi time_freq vd

upvar $lon xdim $lat ydim $ver zdim $tim ldim $lva lvars $pnu pnum $lev levs $stn standard_na

#table corresponds to: monthly, daily, 3hourly

catch {unset xdim ydim zdim ldim lvars pnum levs standard_name long_name units ovar table ext

set levs ""

set var_derivation ""

if { \
$ivar == "als" || \
$ivar == "clc" || \
$ivar == "cld" || \
$ivar == "clh" || \
$ivar == "cll" || \
$ivar == "clm" || \
$ivar == "dtm" || \
$ivar == "evp" || \
$ivar == "hfl" || \
$ivar == "icd" || \
$ivar == "ich" || \
$ivar == "ico" || \
$ivar == "ire" || \
$ivar == "isf" || \
$ivar == "itf" || \
$ivar == "iwp" || \
$ivar == "lwp" || \
$ivar == "per" || \
$ivar == "pmc" || \
$ivar == "psl" || \
$ivar == "ps" || \
$ivar == "psf" || \
$ivar == "pwc" || \
$ivar == "rgc" || \
$ivar == "rgd" || \
$ivar == "rgu" || \
$ivar == "rgn" || \
$ivar == "rnc" || \
$ivar == "rnd" || \
$ivar == "pev" || \
$ivar == "rtc" || \
$ivar == "rtu" || \
$ivar == "run" || \
$ivar == "mrros" || \
$ivar == "rsdt" || \
$ivar == "sgc" || \
```

```
$ivar == "sgn" || \
$ivar == "sid" || \
$ivar == "snd" || \
$ivar == "snc" || \
$ivar == "snw" || \
$ivar == "sno" || \
$ivar == "soc" || \
$ivar == "sot" || \
$ivar == "rlus" || \
$ivar == "rsus" || \
$ivar == "rtmt" || \
$ivar == "rsdscs" || \
$ivar == "rsuscs" || \
$ivar == "rldscs" || \
$ivar == "sb2" || \
$ivar == "sb3" || \
$ivar == "tgf" || \
$ivar == "tgg" || \
$ivar == "thd" || \
$ivar == "thm" || \
$ivar == "tld" || \
$ivar == "tlm" || \
$ivar == "tsu" || \
$ivar == "vmo" || \
$ivar == "wfb" || \
$ivar == "sgd" || \
$ivar == "wfg" || \
$ivar == "etr" || \
$ivar == "trav" || \
$ivar == "gsl" || \
$ivar == "hwdi" || \
$ivar == "cwdi" || \
$ivar == "r10" || \
$ivar == "txav" || \
$ivar == "tnav" || \
$ivar == "tav" || \
$ivar == "pav" || \
$ivar == "cdd" || \
$ivar == "cwd" || \
$ivar == "r5d" || \
$ivar == "sdii" || \
$ivar == "r95t" || \
$ivar == "rw95t" || \
$ivar == "tn10" || \
$ivar == "tx10" || \
$ivar == "tn90" || \
$ivar == "tx90" || \
$ivar == "tmaxnorm" || \
$ivar == "tminnorm" || \
$ivar == "txice" || \
$ivar == "ppww" || \
$ivar == "ppdd" || \
$ivar == "pwsav" || \
$ivar == "pwsmed" || \
$ivar == "pwssdv" || \
$ivar == "pdsav" || \
```

```
$ivar == "pdsmed" || \
$ivar == "pdssdv" || \
$ivar == "ppcr" || \
$ivar == "fd" || \
$ivar == "so4" } {
set xdim lon
set ydim lat
set zdim " "
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "sf2" || \
$ivar == "rf2" || \
$ivar == "sc2" || \
$ivar == "rc2" } {
set xdim lon
set ydim lat
set zdim dummy2
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "mrso" || \
$ivar == "mrsos" || \
$ivar == "uas" || \
$ivar == "vas" || \
$ivar == "suas" || \
$ivar == "svas" || \
$ivar == "shus" || \
$ivar == "huss" || \
$ivar == "hus" || \
$ivar == "tsl" || \
$ivar == "tsh" || \
$ivar == "tsca" || \
$ivar == "v10ma" || \
$ivar == "tsc" } {
set xdim lon
set ydim lat
set zdim dummy
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "qflux"} { \
set xdim lon
set ydim lat
set zdim " "
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "pq20" || \
$ivar == "pq40" || \
$ivar == "pq50" || \
$ivar == "pq60" || \
```

```
$ivar == "pq80" || \
$ivar == "pq90" || \
$ivar == "pq95" || \
$ivar == "pqw20" || \
$ivar == "pqw40" || \
$ivar == "pqw50" || \
$ivar == "pqw60" || \
$ivar == "pqw80" || \
$ivar == "pqw90" || \
$ivar == "pqw95" || \
$ivar == "tnq90" || \
$ivar == "tnq10" || \
$ivar == "txq90" || \
$ivar == "txq10" || \
$ivar == "zobt" || \
$ivar == "sftlf" || \
$ivar == "sftgif" || \
$ivar == "orog"} { \
set xdim lon
set ydim lat
set zdim " "
set ldim dummy
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "icu" || \
$ivar == "icv" || \
$ivar == "V10" || \
$ivar == "u01" || \
$ivar == "v01" || \
$ivar == "tax" || \
$ivar == "tay" } {
set xdim lon
set ydim lat
set zdim " "
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "t4d" || \
$ivar == "q4d" || \
$ivar == "r4d" } {
set levs {01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18}
set c [string range $ivar 0 0]
foreach l $levs {
lappend lvars $c$l
}
set xdim lon
set ydim lat
set zdim pressure
set ldim time
if {$ivar == "q4d" || \
$ivar == "r4d"} {
set pnum 4
} else {
set pnum 3
```

```
}

} elseif { \
$ivar == "u4d" || \
$ivar == "v4d" } {
set levs {01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18}
set c [string range $ivar 0 0]
foreach l $levs {
lappend lvars $c$l
}
set xdim lon
set ydim lat
set zdim pressure
set ldim time
set pnum 4
} elseif { \
$ivar == "w4d" } { \
if {$time_freq == "monthly"} {
set levs {1000 925 850 700 600 500 400 300 250 200 150 100 70 50 30 20 10}
} else {
set levs {1000 925 850 700 600 500 400 300 200}
}
set c "wap"
foreach l $levs {
lappend lvars $c[pad10000 $l]
}
set xdim lon
set ydim lat
set zdim plev
set ldim time
set pnum 4
} elseif { \
$ivar == "c4d" } { \
if {$time_freq == "monthly"} {
set levs {01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18}
} else {
set levs {01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18}
}
set c "c"
foreach l $levs {
lappend lvars $c$l
}
set xdim lon
set ydim lat
set zdim lev
set ldim time
set pnum 4
} elseif { \
$ivar == "z4d" } { \
if {$time_freq == "monthly"} {
set levs {1000 925 850 700 600 500 400 300 250 200 150 100 70 50 30 20 10}
} else {
set levs {1000 925 850 700 600 500 400 300 200}
}
set c "z"
foreach l $levs {
```

```
lappend lvars $c[pad10000 $l]
}
set xdim lon
set ydim lat
set zdim plev
set ldim time
set pnum 4
} elseif { \
$ivar == "zg0200" || $ivar == "zg0500"} {\
set xdim lon
set ydim lat
set zdim plev
set ldim time
set pnum 4
set lvars $ivar
} elseif { \
$ivar == "hfogo" || $ivar == "hfoge"} {\
set xdim region
set ydim lat
set zdim " "
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "stfmmc" || $ivar == "stfmme"} {\
set xdim region
set ydim lat
set zdim depth
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "uedd" || \
$ivar == "vedd" || \
$ivar == "wedd" || \
$ivar == "iso20" || \
$ivar == "rhopoto" || \
$ivar == "Temp" || \
$ivar == "uvel" || \
$ivar == "vvel" || \
$ivar == "wvel" || \
$ivar == "Salt" } {
set xdim lon
set ydim lat
set zdim depth
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "rslr" || \
$ivar == "zos" || \
$ivar == "sst" || \
$ivar == "sss" || \
$ivar == "pme" || \
$ivar == "taux" || \
$ivar == "tauy" || \
```

```
$ivar == "hflx" || \
$ivar == "psi" } {
set xdim lon
set ydim lat
set zdim ""
set ldim time
set lvars $ivar
set pnum 0
} elseif { \
$ivar == "mrfso" || \
$ivar == "hfcorr" || \
$ivar == "wfcorr" || \
$ivar == "tauucorr" || \
$ivar == "tauvcorr" || \
$ivar == "snm" } {
puts "these IPCC variables are not available in Mk3.0"
} else {
puts "Don't know that one."
}

if {$ivar == "tnq90"}  {
set standard_name "tnq90"
set long_name "Tmin 90th percentile"
set units "K"
set ovar "tnq90"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tnq90.tcl"

} elseif {$ivar == "tnq10"}  {
set standard_name "tnq10"
set long_name "Tmin 10th percentile"
set units "K"
set ovar "tnq10"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tnq10.tcl"

} elseif {$ivar == "txq90"}  {
set standard_name "txq90"
set long_name "Tmax 90th percentile"
set units "K"
set ovar "txq90"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_txq90.tcl"

} elseif {$ivar == "txq10"}  {
set standard_name "txq10"
set long_name "Tmax 10th percentile"
```

```
set units "K"
set ovar "txq10"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_txq10.tcl"

} elseif {$ivar == "pq20"}  {
set standard_name "pq20"
set long_name "20th percentile of rainday amounts"
set units "mm day-1"
set ovar "pq20"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pq20.tcl"

} elseif {$ivar == "pq40"}  {
set standard_name "pq40"
set long_name "40th percentile of rainday amounts"
set units "mm day-1"
set ovar "pq40"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pq40.tcl"

} elseif {$ivar == "pq50"}  {
set standard_name "pq50"
set long_name "50th percentile of rainday amounts"
set units "mm day-1"
set ovar "pq50"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pq50.tcl"

} elseif {$ivar == "pq60"}  {
set standard_name "pq60"
set long_name "60th percentile of rainday amounts"
set units "mm day-1"
set ovar "pq60"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pq60.tcl"

} elseif {$ivar == "pq80"}  {
set standard_name "pq80"
set long_name "80th percentile of rainday amounts"
set units "mm day-1"
```

```
set ovar "pq80"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pq80.tcl"

} elseif {$ivar == "pq90"}  {
set standard_name "pq90"
set long_name "90th percentile of rainday amounts"
set units "mm day-1"
set ovar "pq90"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw90.tcl"

} elseif {$ivar == "pq95"}  {
set standard_name "pq95"
set long_name "95th percentile of rainday amounts"
set units "mm day-1"
set ovar "pq95"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pq95.tcl"

} elseif {$ivar == "pqw20"}  {
set standard_name "pqw20"
set long_name "20th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw20"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw20.tcl"

} elseif {$ivar == "pqw40"}  {
set standard_name "pqw40"
set long_name "40th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw40"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw40.tcl"

} elseif {$ivar == "pqw50"}  {
set standard_name "pqw50"
set long_name "50th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw50"
```

```
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw50.tcl"

} elseif {$ivar == "pqw60"}  {
set standard_name "pqw60"
set long_name "60th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw60"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw60.tcl"

} elseif {$ivar == "pqw80"}  {
set standard_name "pqw80"
set long_name "80th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw80"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw80.tcl"

} elseif {$ivar == "pqw90"}  {
set standard_name "pqw90"
set long_name "90th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw90"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw90.tcl"

} elseif {$ivar == "pqw95"}  {
set standard_name "pqw95"
set long_name "95th percentile of rainday amounts for wet days"
set units "mm day-1"
set ovar "pqw95"
set table "Z1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pqw95.tcl"

} elseif {$ivar == "sdii" } {
set standard_name "sdii"
set long_name "Simple Daily Precipitation Index"
set units "kg m-2 s-1"
set ovar "sdii"
set table "xxx xxx xxx A4"
```

```
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_sdii.tcl"

} elseif {$ivar == "r5d" } {
set standard_name "r5d"
set long_name "Maximum 5-day Precipitation Total"
set units "kg m-2"
set ovar "r5d"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_r5d.tcl"

} elseif {$ivar == "tmaxnorm" } {
set standard_name "tmaxnorm"
set long_name "Tmax normal"
set units "K"
set ovar "tmaxnorm"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tmaxnorm.tcl"

} elseif {$ivar == "tminnorm" } {
set standard_name "tminnorm"
set long_name "Tmin normal"
set units "K"
set ovar "tminnorm"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tminnorm.tcl"

} elseif {$ivar == "cdd" } {
set standard_name "cdd"
set long_name "Maximum Number of Consecutive Dry Days"
set units "days"
set ovar "cdd"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_cdd.tcl"

} elseif {$ivar == "cwd" } {
set standard_name "cwd"
set long_name "Maximum Number of Consecutive Wet Days"
set units "days"
set ovar "cwd"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
```

```
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_cwd.tcl"

} elseif {$ivar == "hwdi" } {
set standard_name "hwdi"
set long_name "Heat Wave Duration Index"
set units "days"
set ovar "hwdi"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_hwdi.tcl"

} elseif {$ivar == "cwdi" } {
set standard_name "cwdi"
set long_name "Cold Wave Duration Index"
set units "days"
set ovar "cwdi"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_cwdi.tcl"

} elseif {$ivar == "gsl" } {
set standard_name "gsl"
set long_name "Growing Season Length"
set units "days"
set ovar "gsl"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_gsl.tcl"

} elseif {$ivar == "etr" } {
set standard_name "etr"
set long_name "Intra-Annual Extreme Temperature Range"
set units "K"
set ovar "etr"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_etr.tcl"

} elseif {$ivar == "trav" } {
set standard_name "trav"
set long_name "Mean of Diurnal Temperature Range"
set units "K"
set ovar "trav"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
```

```
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_trav.tcl"

} elseif {$ivar == "fd" } {
set standard_name "fd"
set long_name "Total Number of Frost Days in Year"
set units "days"
set ovar "fd"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_fd.tcl"

} elseif {$ivar == "txice" } {
set standard_name "txice"
set long_name "Total Number of Defrost Days in Year"
set units "days"
set ovar "txice"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_txice.tcl"

} elseif {$ivar == "r95t" } {
set standard_name "r95t"
set long_name "Percent of Annual Total Precipitation due to Events Exceeding the 1961-1990
95th Percentile"
set units "percent"
set ovar "r95t"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_r95t.tcl"

} elseif {$ivar == "rw95t" } {
set standard_name "rw95t"
set long_name "Percent of Annual Total Precipitation due to Events Exceeding the 1961-1990
95th Percentile for wet days"
set units "percent"
set ovar "rw95t"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_rw95t.tcl"

} elseif {$ivar == "tn10" } {
set standard_name "tn10"
set long_name "Percent of time Tmin < the 10th percentile value of daily minimum
temperature"
set units "percent"
set ovar "tn10"
set table "xxx xxx xxx A4"
```

```
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tn90.tcl"

} elseif {$ivar == "tx10" } {
set standard_name "tx10"
set long_name "Percent of time Tmax < the 10th percentile value of daily maximum temperature"
set units "percent"
set ovar "tx10"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tx90.tcl"

} elseif {$ivar == "tn90" } {
set standard_name "tn90"
set long_name "Percent of time Tmin > the 90th percentile value of daily minimum
temperature"
set units "percent"
set ovar "tn90"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tn90.tcl"

} elseif {$ivar == "tx90" } {
set standard_name "tx90"
set long_name "Percent of time Tmax > the 90th percentile value of daily maximum
temperature"
set units "percent"
set ovar "tx90"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tx90.tcl"

} elseif {$ivar == "txav" } {
set standard_name "txav"
set long_name "Minimum Surface Air Temperature"
set units "K"
set ovar "txav"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_txav.tcl"

} elseif {$ivar == "tnav" } {
set standard_name "tnav"
set long_name "Maximum Surface Air Temperature"
set units "K"
set ovar "tnav"
```

```
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tnav.tcl"

} elseif {$ivar == "tav" } {
set standard_name "tav"
set long_name ""
set long_name "Surface Air Temperature"
set units "K"
set ovar "tav"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_tav.tcl"

} elseif {$ivar == "pav" } {
set standard_name "pav"
set long_name "Mean Climatological Precipitation"
set units "kg m-2 s-1"
set ovar "pav"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pav.tcl"

} elseif {$ivar == "ppww" } {
set standard_name "ppww"
set long_name "Mean Wet-day Persistence"
set units "percent"
set ovar "ppww"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_ppww.tcl"

} elseif {$ivar == "ppcr" } {
set standard_name "ppcr"
set long_name "Correlation for spell lengths"
set units "percent"
set ovar "ppcr"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_ppcr.tcl"

} elseif {$ivar == "ppdd" } {
set standard_name "ppdd"
set long_name "Mean Dry-day Persistence"
set units "percent"
set ovar "ppdd"
```

```
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_ppdd.tcl"

} elseif {$ivar == "pdsav" } {
set standard_name "pdsav"
set long_name "Mean Dry Spell Length"
set units "percent"
set ovar "pdsav"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pdsav.tcl"

} elseif {$ivar == "pdsmed" } {
set standard_name "pdsmed"
set long_name "Median Dry Spell Length"
set units "percent"
set ovar "pdsmed"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pdsav.tcl"

} elseif {$ivar == "pdssdv" } {
set standard_name "pdsssdv"
set long_name "Dry Spell Length Standard Deviation"
set units "percent"
set ovar "pdssdv"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pdsav.tcl"

} elseif {$ivar == "pwsav" } {
set standard_name "pwsav"
set long_name "Mean Wet Spell Length"
set units "percent"
set ovar "pwsav"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pwsav.tcl"

} elseif {$ivar == "pwsmed" } {
set standard_name "pwsmed"
set long_name "Median Wet Spell Length"
set units "percent"
set ovar "pdsmed"
set table "xxx xxx xxx A4"
```

```
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pwsav.tcl"

} elseif {$ivar == "pwssdv" } {
set standard_name "pwssdv"
set long_name "Wet Spell Length Standard Deviation"
set units "percent"
set ovar "pdssdv"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_pwsav.tcl"

} elseif {$ivar == "r10" } {
set standard_name "r10"
set long_name "Number of Days for which Precipitation Rate Exceeds 10 mm/day"
set units "days"
set ovar "r10"
set table "xxx xxx xxx A4"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/extreme_indices/ipcc_ei_r10.tcl"

} elseif {$ivar == "tsc" } {
set standard_name "air_temperature"
set long_name "Surface Air Temperature"
set units "K"
set ovar "tas"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"

if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "so4" } {
set standard_name "atmosphere_content_of_sulfate_aerosol"
set long_name "Sulfate Burden"
set units "1e-6 kg m-2"
set ovar "trsult"
set table "A1e xxx xxx xxx xxx xxx"
set exter_scale 0.3e1
```

```
set exter_offset 0.0e1
set land_sea_interp "yes"

if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "tld" } {
set standard_name "air_temperature"
set long_name "Minimum Monthly Surface Air Temperature"
set units "K"
set ovar "tasmin"
set table "A1f xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "thd" } {
set standard_name "air_temperature"
set long_name "Maximum Monthly Surface Air Temperature"
set units "K"
set ovar "tasmax"
set table "A1f xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "tsl" } {
```

```
set standard_name "air_temperature"
set long_name "Minimum Daily Surface Air Temperature"
set units "K"
set ovar "tasmin"
set table "xxx A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "tsh" } {
set standard_name "air_temperature"
set long_name "Maximum Daily Surface Air Temperature"
set units "K"
set ovar "tasmax"
set table "xxx A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "huss" } {
set standard_name "specific_humidity"
set long_name "Surface Specific Humidity"
set units "kg kg-1"
set ovar "huss"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_calc_day2mon.tcl (inpu
} elseif {$ivar == "uas" || $ivar == "u01"} {
set standard_name "eastward_wind"
set long_name "Zonal Surface Wind Speed"
set units "m s-1"
set ovar "uas"
set table "A1a A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
```

```
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_calc_day2mon.tcl (inpu
} elseif {$time_freq == "daily"} {
set var_derivation "Computed using daily histb files"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
}
} elseif {$ivar == "vas" || $ivar == "v01"} {
set standard_name "northward_wind"
set long_name "Meridional Surface Wind Speed"
set units "m s-1"
set ovar "vas"
set table "A1a A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_calc_day2mon.tcl
(input files are daily histb files)"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed using daily histb files"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
}
} elseif {$ivar == "suas" } {
# Calculated directly by model
set standard_name "eastward_wind"
set long_name "Zonal Surface Wind Speed"
set units "m s-1"
set ovar "uas"
set table "A1a A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed using daily histb files"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
}
} elseif {$ivar == "svas"} {
set standard_name "northward_wind"
set long_name "Meridional Surface Wind Speed"
set units "m s-1"
set ovar "vas"
set table "A1a A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
```

```
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed using daily histb files"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
}
} elseif {$ivar == "shus" } {
# Calculated directly by model
set standard_name "specific_humidity"
set long_name "Surface Specific Humidity"
set units "kg kg-1"
set ovar "huss"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values of shus"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed using daily histb files of shus"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values of shus"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values of shus"
}
} elseif {$ivar == "mrso"}  {
set standard_name "soil_moisture_content"
set long_name "Total soil moisture"
set units "kg m-2"
set ovar "mrso"
set table "A1a xxx xxx xxx"
set var_derivation ""
} elseif {$ivar == "mrsos"}  {
set standard_name "moisture_content_of_soil_layer"
set long_name "Moisture in top 0.1 m of soil"
set units "kg m-2"
set ovar "mrsos"
set table "A1a xxx xxx xxx"
set var_derivation ""

} elseif {$ivar == "tsca"}  {
set standard_name "air_temperature"
set long_name "Surface Air Temperature"
set units "K"
set ovar "tas"
set table "xxx A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation ""
```

```
} elseif {$ivar == "v10ma"}  {
set standard_name "surface_wind_speed"
set long_name "Surface Wind Speed"
set units "m s-1"
set ovar "vmo"
set table "xxx A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation ""

} elseif {$ivar == "sftgif"}  {
set standard_name "land_ice_area_fraction"
set long_name "Fraction of Land Area Covered with Glacier"
set units "percent"
set ovar "sftgif"
set table "A1b xxx xxx xxx"
set exter_scale 100.0
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_sftgif.tcl (input
files ~col414/ocean/masks/Mk3_agcm.nc ~IPCC/processing/text_directory/glacier.txt)"
} elseif {$ivar == "sftlf"}  {
set standard_name "land_area_fraction"
set long_name "Land Area Fraction"
set units "percent"
set ovar "sftlf"
set table "A1b xxx xxx xxx"
set exter_scale 100.0
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_sftlf.tcl (input
file ~col414/ocean/masks/Mk3_agcm.nc)"
} elseif {$ivar == "orog"}  {
set standard_name "surface_altitude"
set long_name "Surface Altitude"
set units "m"
set ovar "orog"
set table "A1b xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/invariant/ipcc_orog.tcl (input
file ~col414/cec006/CMIP/AGCM/histb.dir/histb.m12.y121.nc)"
} elseif {$ivar == "zobt"}  {
set standard_name "sea_floor_depth_below_geoid"
set long_name "Sea Floor Depth"
set units "m"
set ovar "zobt"
set table "O1d xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/invariant/ipcc_zobt.tcl (input
file ~col414/cec006/CMIP/MOM2/avg.dir/avg.m12.y121.nc)"
} elseif {$ivar == "qflux"}  {
```

```
set standard_name "prescribed_heat_flux_into_slab_ocean"
set long_name "Q-Flux"
set units "W m-2"
set ovar "qflux"
set table "O1d xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/annual/ipcc_qflux.tcl (input
file ~IPCC/data/qflux/*.cif)"
} elseif {$ivar == "w4d"}  {
set standard_name "lagrangian_tendency_of_air_pressure"
set long_name "Omega"
set units "Pa s-1"
set ovar "wap"
set table "A1c xxx xxx xxx A5b A6b"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_calc_day2mon_wap.tcl
(input files are daily histb files, output files are written to raw netcdf directories)"
} elseif {$ivar == "t4d"}  {
set standard_name "air_temperature"
set long_name "temperature"
set units "K"
set ovar "ta"
set table "A1c A2b xxx xxx A5b A6b"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from data on pressure coordinates"
} elseif {$time_freq == "daily" || $time_freq == "12hourly" || $time_freq == "6hourly"} {
set var_derivation "Computed from data on hybrid sigma coordinates"
}
} elseif {$ivar == "u4d"}  {
set standard_name "eastward_wind"
set long_name "Zonal Wind Component"
set units "m s-1"
set ovar "ua"
set table "A1c A2b xxx xxx A5b A6b"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from data on pressure coordinates"
} elseif {$time_freq == "daily" || $time_freq == "12hourly" || $time_freq == "6hourly"} {
set var_derivation "Computed from data on hybrid sigma coordinates"
}
} elseif {$ivar == "v4d"}  {
set standard_name "northward_wind "
set long_name "Meridional Wind Component"
set units "m s-1"
set ovar "va"
set table "A1c A2b xxx xxx A5b A6b"
set exter_scale 0.1e1
```

```
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from data on pressure coordinates"
} elseif {$time_freq == "daily" || $time_freq == "12hourly" || $time_freq == "6hourly"} {
set var_derivation "Computed from data on hybrid sigma coordinates"
}
} elseif {$ivar == "q4d"}  {
set standard_name "specific_humidity"
set long_name "Specific Humidity"
set units "1"
set ovar "hus"
set table "A1c A2b xxx xxx A5b A6b"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from data on pressure coordinates"
} elseif {$time_freq == "daily" || $time_freq == "12hourly" || $time_freq == "6hourly"} {
set var_derivation "Computed from data on hybrid sigma coordinates"
}
} elseif {$ivar == "r4d"}  {
set standard_name "relative_humidity"
set long_name "Relative Humidity"
set units "percent"
set ovar "hur"
set table "A1c A2b xxx xxx A5b A6b"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from data on pressure coordinates"
} elseif {$time_freq == "daily" || $time_freq == "12hourly" || $time_freq == "6hourly"} {
set var_derivation "Computed from data on hybrid sigma coordinates"
}
} elseif {$ivar == "z4d" || $ivar == "zg0200" || $ivar == "zg0500"}  {
set standard_name "geopotential_height"
set long_name "Geopotential Height"
set units "m"
  if {$ivar == "z4d"} {
  set ovar "zg"
  } else {
  set ovar $ivar
  }
set table "A1c A2b xxx xxx A5b A6b"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_zg.tcl (which
itself uses ~dix043/bin/heights_std - input files are daily histb files, output files are
written to raw netcdf directories)"
} elseif {$time_freq == "daily" || $time_freq == "12hourly" || $time_freq == "6hourly"} {
set var_derivation "Computed using ~dix043/bin/histheights"
}
} elseif {$ivar == "sst"}  {
```

```
set standard_name "sea_surface_temperature"
set long_name "Sea Surface Temperature"
set units "K"
set ovar "tos"
set table "O1c xxx xxx xxx"
set exter_offset -273.15
set exter_scale 0.1e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "sss"}  {
set standard_name "sea_surface_salinity"
set long_name "Sea Surface Salinity"
set units "1e-3"
set ovar "sos"
set table "O1c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rslr"}  {
set standard_name "relative_sea_surface_rise"
set long_name "Relative Sea Surface Rise"
set units "m"
set ovar "rslr"
set table "O1c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/miscellaneous/ipcc_rslr.tcl (input
files are created using the method of Jacket et al. 2000 (J. Climate 13(8), 1384-1405)
where pressure term is calculated using geostrophic approximation at a given depth level (793
} elseif {$ivar == "zos"}  {
set standard_name "sea_surface_elevation"
set long_name "Sea Surface Elevation"
set units "m"
set ovar "zos"
```

```
set table "O1c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "pme"}  {
set standard_name "water_flux_into_ocean"
set long_name "Water Flux Into Ocean"
set units "kg m-2 s-1"
set ovar "wfo"
set table "O1c xxx xxx xxx"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "hflx"}  {
set standard_name "heat_flux_into_ocean"
set long_name "Heat Flux Into Ocean"
set units "W m-2"
set ovar "hfo"
set table "O1c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rnd"}  {
```

```
set standard_name "precipitation_flux"
set long_name "Precipitation"
set units "kg m-2 s-1"
set ovar "pr"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "pwc"}  {
set standard_name "atmosphere_water_vapor_content"
set long_name "Preciptable Water"
set units "kg m-2"
set ovar "prw"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "psl"}  {
set standard_name "air_pressure_at_sea_level"
set long_name "Sea Level Pressure"
set units "Pa"
set ovar "psl"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 100.0
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values in histb files"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
```

```
} elseif {$ivar == "ps"}  {
set standard_name "surface_air_pressure"
set long_name "Surface Pressure"
set units "Pa"
set ovar "ps"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e3
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values in histb files"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "vmo"}  {
set standard_name "surface_wind_speed"
set long_name "Surface Wind Speed"
set units "m s-1"
set ovar "vmo"
set table "A1a A2a xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values in histb files"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "wfb"}  {
set standard_name "lower_level_soil_moisture"
set long_name "Lower Level Soil Moisture"
set units "1"
set ovar "wfb"
set table "A1a A2a xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values in histb files"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "wfg"}  {
set standard_name "upper_level_soil_moisture"
set long_name "Upper Level Soil Moisture"
set units "1"
set ovar "wfg"
```

```
set table "A1a A2a xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values in histb files"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "psf"}  {
set standard_name "surface_air_pressure"
set long_name "Surface Pressure"
set units "Pa"
set ovar "ps"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e3
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation ""
} elseif {$ivar == "V10"}  {
set standard_name "wind_speed_at_10m"
set long_name "wind_speed_at_10m"
set units "m s-1"
set ovar "V10"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_calc_day2mon.tcl
(input files are daily histb files)"
} elseif {$ivar == "tax"}  {
set standard_name "surface_downward_eastward_stress"
set long_name "Zonal Surface Wind Stress"
set units "Pa"
set ovar "tauu"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "tay"}  {
set standard_name "surface_downward_northward_stress"
set long_name "Meridional Surface Wind Stress"
set units "Pa"
set ovar "tauv"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
```

```
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "taux"}  {
set standard_name "surface_downward_eastward_stress"
set long_name "Zonal Surface Wind Stress"
set units "Pa"
set ovar "taux"
set table "O1c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "tauy"}  {
set standard_name "surface_downward_northward_stress"
set long_name "Meridional Surface Wind Stress"
set units "Pa"
set ovar "tauy"
set table "O1c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "snd"}  {
set standard_name "surface_snow_thickness"
set long_name "Snow Depth"
set units "m"
set ovar "snd"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e-1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
```

```
}
} elseif {$ivar == "snc"} {
set standard_name "surface_snow_area_fraction_where_land"
set long_name "Snow Area Fraction"
set units "percent"
set ovar "snc"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "snw"}  {
set standard_name "surface_snow_amount_where_land"
set long_name "Snow Amount"
set units "kg m-2"
set ovar "snw"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "evp"}  {
set standard_name "surface_upward_latent_heat_flux"
set long_name "Surface Latent Heat Flux"
set units "W m-2"
set ovar "hfls"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 28.94;#changed 23/11/04
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "pev"}  {
set standard_name "surface_potential_evaporation"
set long_name "Potential Evaporation"
```

```
set units "kg m-2 s-1"
set ovar "pev"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed from monthly means - mention how got from daily files"
} elseif {$ivar == "hfl"}  {
set standard_name "surface_upward_sensible_heat_flux"
set long_name "Surface Sensible Heat Flux"
set units "W m-2"
set ovar "hfss"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1;#changed 23/11/04
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rgd"}  {
set standard_name "surface_downwelling_longwave_flux_in_air"
set long_name "Surface Downwelling Longwave Radiation"
set units "W m-2"
set ovar "rlds"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rgn"}  {
set standard_name "surface_net_longwave_flux"
set long_name "Surface Net Longwave Radiation"
set units "W m-2"
set ovar "rgn"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale -0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
```

```
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rlus"}  {
set standard_name "surface_upwelling_longwave_flux_in_air"
set long_name "Surface Upwelling Longwave Radiation"
set units "W m-2"
set ovar "rlus"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1;#changed 23/11/04
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rlus.tcl (input files
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values (using rgd+rgn)"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "sgd"}  {
set standard_name "surface_downwelling_shortwave_flux_in_air"
set long_name "Surface Downwelling Shortwave Radiation"
set units "W m-2"
set ovar "rsds"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "sgn"}  {
set standard_name "surface_net_shortwave_flux"
set long_name "Surface Net Shortwave Radiation"
set units "W m-2"
set ovar "sgn"
```

```
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rsus"}  {
set standard_name "surface_upwelling_shortwave_flux_in_air"
set long_name "Surface Upwelling Shortwave Radiation"
set units "W m-2"
set ovar "rsus"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1;#changed 23/11/04
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rsus.tcl (input
files are monthly means using sgd-sgn)"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values (using sgd-sgn)"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "tsu"}  {
set standard_name "surface_temperature"
set long_name "Surface Skin Temperature"
set units "K"
set ovar "ts"
set table "A1a A2a A3 xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
```

```
} elseif {$ivar == "sno"}  {
set standard_name "snowfall_flux"
set long_name "Snowfall Flux"
set units "kg m-2 s-1"
set ovar "prsn"
set table "A1a xxx xxx xxx"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rnc"}  {
set standard_name "convective_precipitation_flux"
set long_name "Convective Precipiation Flux"
set units "kg m-2 s-1"
set ovar "prc"
set table "A1a xxx xxx xxx"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "run"}  {
set standard_name "runoff_flux"
set long_name "Surface and Subsurface Runoff"
set units "kg m-2 s-1"
set ovar "mrro"
set table "A1a xxx xxx xxx"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} else {
set var_derivation "Computed from 3-hourly values"
}
```

```
} elseif {$ivar == "mrros"}  {
set standard_name "surface_runoff_flux"
set long_name "Surface Runoff"
set units "kg m-2 s-1"
set ovar "mrros"
set table "A1a xxx xxx xxx"
set exter_scale [expr 1.0/86400.0]
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation ""
} elseif {$ivar == "rtmt"}  {
set standard_name "net_downward_radiative_flux_at_top_of_atmosphere_model"
set long_name "Net Flux at Top of Model"
set units "W m-2"
set ovar "rtmt"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rtmt.tcl (input
files are monthly means using sit-sot-rtu)"
} elseif {$ivar == "rsdt"}  {
set standard_name "toa_incoming_shortwave_flux"
set long_name "TOA Incident Shortwave Radiation"
set units "W m-2"
set ovar "rsdt"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
set var_derivation "Computed using ~IPCC/processing/tools/annual/ipcc_rsdt.tcl (input
file ~IPCC/data/input/sint_t63.nc)"
} elseif {$ivar == "rsdscs"}  {
set standard_name "surface_downwelling_shortwave_flux_in_air_assuming_clear_sky"
set long_name "Incident Surface Clear-Sky Shortwave"
set units "W m-2"
set ovar "rsdscs"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation ""
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rsdscs.tcl (input
files are monthly means using sgd*sgc/sgn)"
} elseif {$ivar == "rsuscs"}  {
set standard_name "surface_upwelling_shortwave_flux_in_air_assuming_clear_sky"
set long_name ""
set units "W m-2"
set ovar "rsuscs"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rsuscs.tcl (input
files are monthly means using sgd*sgc/sgn-sgc)"
} elseif {$ivar == "rldscs"}  {
```

```
set standard_name "surface_downwelling_longwave_flux_in_air_assuming_clear_sky"
set long_name ""
set units "W m-2"
set ovar "rldscs"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rldscs.tcl (input
files are monthly means using rgn+rgd-rgc)"
} elseif {$ivar == "sot"}  {
set standard_name "toa_outgoing_shortwave_flux"
set long_name "TOA Reflected Shortwave Radiation"
set units "W m-2"
set ovar "rsut"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rtc"}  {
set standard_name "toa_outgoing_longwave_flux_assuming_clear_sky"
set long_name "Outgoing Clear-Sky Longwave"
set units "W m-2"
set ovar "rlutcs"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "soc"}  {
set standard_name "toa_outgoing_shortwave_flux_assuming_clear_sky"
set long_name "Reflected Clear-Sky Shortwave"
set units "W m-2"
set ovar "rsutcs"
set table "A1a xxx xxx xxx"
```

```
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rtu"}  {
set standard_name "toa_outgoing_longwave_flux"
set long_name "Outgoing Longwave Radiation"
set units "W m-2"
set ovar "rlut"
set table "A1a A2a xxx xxx A5a A6a"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "lwp"}  {
set standard_name "atmosphere_cloud_condensed_water_content"
set long_name "Column Integrated Cloud Water Content"
set units "kg m-2"
set ovar "clwvi"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "iwp"}  {
set standard_name "atmosphere_cloud_ice_content"
```

```
set long_name "Column Integrated Cloud Ice Content"
set units "kg m-2"
set ovar "clivi"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "cld"}  {
set standard_name "cloud_area_fraction"
set long_name "Total Cloud Fraction"
set units "percent"
set ovar "clt"
set table "A1a xxx xxx xxx"
set exter_scale 100.0
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "ico"}  {
set standard_name "sea_ice_area_fraction"
set long_name "Sea Ice Concentration"
set units "percent"
set ovar "sic"
set table "O1c xxx xxx xxx"
set exter_scale 100.0
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
```

```
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "icd"}  {
set standard_name "sea_ice_thickness"
set long_name "Sea Ice Thickness"
set units "m"
set ovar "sit"
set table "01c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "icu"}  {
set standard_name "eastward_sea_ice_velocity"
set long_name "Eastward Sea Ice Velocity"
set units "m s-1"
set ovar "usi"
set table "01c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "icv"}  {
set standard_name "northward_sea_ice_velocity"
set long_name "Northward Sea Ice Velocity"
set units "m s-1"
set ovar "vsi"
set table "01c xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
set var_derivation "Computed from monthly means"
} elseif {$ivar == "psi"}  {
set standard_name "ocean_barotropic_streamfunction"
set long_name "Ocean Barotropic Streamfunction"
set units "m3 s-1"
```

```
set ovar "stfbarot"
set table "O1c xxx xxx xxx"
set exter_scale 0.1e7
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "Temp"}  {
set standard_name "sea_water_potential_temperature"
set long_name "Potential Temperature"
set units "K"
set ovar "thetao"
set table "O1e xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset -273.15
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rhopoto"}  {
set standard_name "sea_water_potential_density"
set long_name "Potential Density"
set units "kg m-3"
set ovar "rhopoto"
set table "O1e xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 1.0
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_rhopoto.tcl from
IPCC variables thetao and so)"
} elseif {$ivar == "hfogo"}  {
set standard_name "northward_ocean_heat_transport"
set long_name "Northward Ocean Heat Transport"
set units "W"
set ovar "hfogo"
set table "O1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 1.0
set land_sea_interp "yes"
```

```
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_hfogo_ferret.tcl in
ferretv5.70 from IPCC variables vo and thetao. thetao regridded to vo grid. The heat
transports given here are only the RESOLVED ADVECTIVE component. Contributions resulting
from lateral diffusion or eddy-induced transport are not included"
} elseif {$ivar == "stfmmc"}  {
set standard_name "ocean_meridional_overturning_streamfunction"
set long_name "Meridional Overturning Streamfunction"
set units "m3 s-1"
set ovar "stfmmc"
set table "O1b xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 1.0
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_stfmmc_ferret.tcl
in ferretv5.70 from IPCC variables vo. The meridional overturning streamfunctions given
here are for the RESOLVED VELOCITY only. No contribution from eddy-induced transports is
included"
} elseif {$ivar == "hfoge"}  {
set standard_name "northward_ocean_heat_transport"
set long_name "Eddy Induced Northward Ocean Heat Transport"
set units "W"
set ovar "hfoge"
set table "O1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 1.0
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_hfogo_ferret.tcl in
ferretv5.70 from IPCC variables vo and thetao. thetao regridded to vo grid. The heat
transports given here are the eddy-induced transports only."
} elseif {$ivar == "stfmme"}  {
set standard_name "ocean_meridional_overturning_streamfunction"
set long_name "Eddy Induced Meridional Overturning Streamfunction"
set units "m3 s-1"
set ovar "stfmme"
set table "O1b xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 1.0
set land_sea_interp "yes"
set var_derivation "Computed using ~IPCC/processing/tools/derived/ipcc_stfmmc_ferret.tcl
in ferretv5.70 from IPCC variables ve. The meridional overturning streamfunctions given
here are for the eddy-induced transports only."
} elseif {$ivar == "Salt"}  {
set standard_name "sea_water_salinity"
set long_name "Salinity"
set units "1e-3"
set ovar "so"
set table "O1e xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
```

```
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "uvel"}  {
set standard_name "eastward_sea_water_velocity"
set long_name "Eastward Sea Water Velocity"
set units "m s-1"
set ovar "uo"
set table "01e xxx xxx xxx"
set exter_scale 0.01
set exter_offset 0.0
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "vvel"}  {
set standard_name "northward_sea_water_velocity"
set long_name "Northward Sea Water Velocity"
set units "m s-1"
set ovar "vo"
set table "01e xxx xxx xxx"
set exter_scale 0.01
set exter_offset 0.0
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "wvel"}  {
set standard_name "upward_sea_water_velocity"
set long_name "Upward Sea Water Velocity"
set units "m s-1"
set ovar "wo"
set table "01e xxx xxx xxx"
set exter_scale 0.01
set exter_offset 0.0
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
```

```
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "uedd"}  {
set standard_name "eastward_sea_water_eddy_velocity"
set long_name "Eastward Sea Water Eddy Velocity"
set units "m s-1"
set ovar "ue"
set table "O1e xxx xxx xxx"
set exter_scale 0.01
set exter_offset 0.0
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "vedd"}  {
set standard_name "northward_sea_water_eddy_velocity"
set long_name "Northward Sea Water Eddy Velocity"
set units "m s-1"
set ovar "ve"
set table "O1e xxx xxx xxx"
set exter_scale 0.01
set exter_offset 0.0
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "wedd"}  {
set standard_name "upward_sea_water_eddy_velocity"
set long_name "Upward Sea Water Eddy Velocity"
set units "m s-1"
set ovar "we"
set table "O1e xxx xxx xxx"
```

```
set exter_scale 0.01
set exter_offset 0.0
set land_sea_interp "yes"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}

} elseif {$ivar == "sf2"}  {
set standard_name "net_downward_shortwave_flux_in_air"
set long_name "Net Downward Shortwave Flux at 200 hPa"
set units "W m-2"
set ovar "rsntp"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rf2"}  {
set standard_name "net_upward_longwave_flux_in_air"
set long_name "Net Upward Longwave Flux at 200 hPa"
set units "W m-2"
set ovar "rlntp"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "sc2"}  {
```

```
set standard_name "net_downward_shortwave_flux_in_air_assuming_clear_sky"
set long_name "Net Downward Clear-Sky Shortwave Flux at 200 hPa"
set units "W m-2"
set ovar "rsntpcs"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "rc2"}  {
set standard_name "net_upward_longwave_flux_in_air_assuming_clear_sky"
set long_name "Net Upward Clear-Sky Longwave Flux at 200 hPa"
set units "W m-2"
set ovar "rlntpcs"
set table "A1a xxx xxx xxx"
set exter_scale 0.1e1
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from monthly values"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from daily values"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
} else {
set var_derivation "Computed from 3-hourly values"
}
} elseif {$ivar == "c4d"}  {
set standard_name "cloud_area_fraction_in_atmosphere_layer"
set long_name "Total Cloud Fraction"
set units "percent"
set ovar "cl"
set table "A1c xxx xxx xxx"
set exter_scale 100.0
set exter_offset 0.0e1
set land_sea_interp "no"
if {$time_freq == "monthly"} {
set var_derivation "Computed from data on hybrid sigma pressure coordinates"
} elseif {$time_freq == "daily"} {
set var_derivation "Computed from data on hybrid sigma pressure coordinates"
} elseif {$time_freq == "12hourly"} {
set var_derivation "Computed from 12-hourly values"
} elseif {$time_freq == "6hourly"} {
set var_derivation "Computed from 6-hourly values"
```

```
}
}

puts "ivar=$ivar"
puts "time_freq=${time_freq}"

puts "xdim=$xdim"
puts "ydim=$ydim"
puts "zdim=$zdim"
puts "ldim=$ldim"
puts "lvars=$lvars"
puts "pnum=$pnum"
puts "levs=$levs"
puts "standard_name=$standard_name"
puts "long_name=$long_name"
puts "units=$units"
puts "ovar=$ovar"
puts "table=$table"
puts "exter_scale=$exter_scale"
puts "exter_offset=$exter_offset"
puts "land_sea_interp=$land_sea_interp"
puts "var_derivation=$var_derivation"

};#end procedure ipcc_cmv
```

14.23 ipcc_data_final

ipcc_data_final.tcl is an important Tcl-Nap script as it is used to generate almost all of the high priority IPCC parameters in Tables 5- 15. Within in at the Tcl-Nap procecures define in Table 25. Typically only small segments of the script it used at a time by deleting comment (#) marks as required.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September 2004
#Modified     Novemer   2005

source ~/.tclshrc

source [glob ~IPCC/processing/tools/ipcc_scripts.tcl]

source ~IPCC/processing/tools/ipcc_calc_day_mk3agcm_new.tcl

# Ocean monthly mean section

set vars {pme hflx psi sst sss Temp vvel Salt uvel wvel uedd vedd wedd taux tauy}

foreach v $vars {

file mkdir /work/IPCC/$v

#PIcntrl - run1
#ipcc_calc_mon_mk3ogcm uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10 1 491
/cs/datastore/csdar no no 0.0 0.0 2 {1 31} yes no
#ipcc_calc_mon_mk3ogcm uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10 121 491
/cs/datastore/csdar no no 0.0 0.0 2 {1 31} yes no

#PIcntrl - run2
#ipcc_calc_mon_mk3ogcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar no no 0.0 0.0 7 {1 31} yes no

#20C3M
#ipcc_calc_mon_mk3ogcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 121
241 /cs/datastore/csdar no no 0.0 0.0 3 {1 31} yes no

#SRESA2 - run1
#ipcc_calc_mon_mk3ogcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 251
341 /cs/datastore/csdar no no 0.0 0.0 12 {1 31} yes no
#ipcc_calc_mon_mk3ogcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 491
491 /cs/datastore/csdar no no 0.0 0.0 12 {1 31} yes no

#Commit
#ipcc_calc_mon_mk3ogcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
341 /cs/datastore/csdar no no 0.0 0.0 5 {1 31} yes no

#SRESA1B
#ipcc_calc_mon_mk3ogcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
441 /cs/datastore/csdar no no 0.0 0.0 8 {1 31} yes no

#SRESB1
#191 is 1941-1950.
#451 is 2201-2210.
```

```
#491 is 2241-2250.
#ipcc_calc_mon_mk3ogcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 251
541 /cs/datastore/csdar no no 0.0 0.0 1 {1 31} yes no
#ipcc_calc_mon_mk3ogcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 191
541 /cs/datastore/csdar no no 0.0 0.0 1 {1 31} yes no

#1pctto2x
#ipcc_calc_mon_mk3ogcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar no no 0.0 0.0 6 {1 31} yes no

#Slabcntl - not applicable for ocean fields
#ipcc_calc_mon_mk3ogcm qv1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 41
/cs/datastore/csdar no no 0.0 0.0 9 {1 31} yes no

#2xCO2 - not applicable for ocean fields
#ipcc_calc_mon_mk3ogcm qv2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 21
/cs/datastore/csdar no no 0.0 0.0 10 {1 31} yes no

#20C3M
#ipcc_calc_mon_mk3ogcm n20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 131 10 131
251 /cs/datastore/csdar no no 0.0 0.0 15 {1 31} yes no

#Spinup - run1
#ipcc_calc_mon_mk3ogcm uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1741 1701 10
1701 1701 /cs/datastore/csdar no no 0.0 0.0 17 {1 31} yes no

#20C3M
#ipcc_calc_mon_mk3ogcm o20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 141 10 141
261 /cs/datastore/csdar no no 0.0 0.0 19 {1 31} yes no

#PIcntrl - run1 - mk3_5d
#eddy output starts at 501
#81 571
#ipcc_calc_mon_mk3ogcm N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 81
1161 /cs/datastore/csdar no no 0.0 0.0 16 {1 31} yes no

#20C3M - run1 - mk3_5d
#eddy output starts at 346
#171 291
#ipcc_calc_mon_mk3ogcm B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 171
291 /cs/datastore/csdar no no 0.0 0.0 20 {1 31} yes no

#SRESA2 - run1 - mk3_5d
#301 381
#eddy output starts at 346
#ipcc_calc_mon_mk3ogcm B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 301
391 /cs/datastore/csdar no no 0.0 0.0 21 {1 31} yes no

#1pctto2x - run1 - mk3_5d
#no eddy output
#ipcc_calc_mon_mk3ogcm M35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 171 10 171
241 /cs/datastore/csdar no no 0.0 0.0 22 {1 31} yes no

#SRESA1B - run1 - mk3_5d
#301 501
#eddy output starts at 342
```

```
#ipcc_calc_mon_mk3ogcm D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 351 5
91 /cs/datastore/csdar no no 0.0 0.0 23 {1 31} yes no

#SRESB1 - run1 - mk3_5d
#eddy output starts at 317
#301-600
#ipcc_calc_mon_mk3ogcm E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 301
591 /cs/datastore/csdar no no 0.0 0.0 24 {1 31} yes no

#Spinup - run1 - mk3_5d
#ipcc_calc_mon_mk3ogcm uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 8301 8301 5 8301
8301 /cs/datastore/csdar no no 0.0 0.0 25 {1 31} yes no

#Commit - run1 - mk3_5d
#ipcc_calc_mon_mk3ogcm C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 301
391 /cs/datastore/csdar no no 0.0 0.0 26 {1 31} yes no

#20C3M - run2 - mk3_5d
#eddy output starts at nnn
#191 191
#ipcc_calc_mon_mk3ogcm F35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 191 10 191
311 /cs/datastore/csdar no no 0.0 0.0 27 {1 31} yes no

#20C3M - run3 - mk3_5d
#eddy output starts at nnn
#211 211
#ipcc_calc_mon_mk3ogcm G35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 211 10 211
331 /cs/datastore/csdar no no 0.0 0.0 28 {1 31} yes no

#PIcntrl - run1 - mk3_5A
#111 231
#ipcc_calc_mon_mk3ogcm C11 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 111 251
/cs/datastore/csdar no no 0.0 0.0 31 {1 31} yes no

#111 251

#PDcntrl - run1 - mk3_5A
#ipcc_calc_mon_mk3ogcm C12 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 61
/cs/datastore/csdar no no 0.0 0.0 36 {1 31} yes no
};#v

# Atmosphere monthly mean section.
set vars {cld};#clt
set vars {evp};#hfls
set vars {hfl};#hfss iflg=13/14/15/19 only"
set vars {huss};#derived
set vars {icd};#sit
set vars {ico};#sic
set vars {icu};#usi
set vars {icv};#vsi
set vars {iwp};#clivi
set vars {lwp};#clwvi
set vars {mrros};#derived
set vars {pev};#not IPCC variable
set vars {psf};#iflg=13/14/15/16>y261/19 only, preferred surface pressure.
set vars {psl};#psl
```

```
set vars {ps};#not for qv3/qv4/n20/o20
set vars {pwc};#prw
set vars {rc2};#rlntpcs iflg=13/14/15/16>y261/19 only"
set vars {rf2};#rlntp iflg=13/14/15/16>y261/19 only"
set vars {rgd};#rlds
set vars {rgn};#not IPCC variable
set vars {rldscs};#derived
set vars {rlus};#derived
set vars {rnc};#prc
set vars {rnd};#pr
set vars {rsdscs};#derived
set vars {rsuscs};#derived
set vars {rsus};#derived
set vars {rtc};#rlutcs
set vars {rtmt};#derived
set vars {rtu};#rlut
set vars {run};#mrro
set vars {sc2};#rsntpcs
set vars {sc2};#rsntpcs iflg=13/14/15/16>y261/19 only"
set vars {sf2};#rsntp
set vars {sf2};#rsntp iflg=13/14/15/16>y261/19 only"
set vars {sgd};#rsds
set vars {sgn};#not IPCC variable
set vars {shus};#iflg=15/16>y261/19 only
set vars {snc};#derived
set vars {snd};#snd
set vars {sno};#prsn
set vars {snw};#derived
set vars {so4};#trsult
set vars {soc}
set vars {sot};#rsut
set vars {suas};#iflg=16>y261
set vars {svas};#iflg=16>y261
set vars {tax};#tauu
set vars {tay};#tauv
set vars {thd}
set vars {tld}
set vars {tsc};#tas
set vars {tsu};#ts
set vars {uas}
set vars {V10};#not IPCC variable
set vars {vas}
set vars {vmo};#vmo, write out for joseph
set vars {wfb};#wfb, write out for joseph
set vars {wfg};#wfg, write out for joseph
foreach v $vars {

file mkdir /work/IPCC/$v

#PIcntrl - run1
#ipcc_calc_mon_mk3agcm uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10 121 491
/cs/datastore/csdar 2 no yes yes no

#PIcntrl - run2
#ipcc_calc_mon_mk3agcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar 7 no yes yes no
```

```
#20C3M
#ipcc_calc_mon_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 121
241 /cs/datastore/csdar 3 no yes yes no

#SRESA2 - run1
#ipcc_calc_mon_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 251
341 /cs/datastore/csdar 12 no yes yes no

#Commit
#ipcc_calc_mon_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
341 /cs/datastore/csdar 5 no yes yes no

#SRESA1B
#ipcc_calc_mon_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
441 /cs/datastore/csdar 8 no yes yes no

#SRESB1
#191 corresponds to 1941
#ipcc_calc_mon_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 251
541 /cs/datastore/csdar 1 no yes yes no

#1pctto2x
#ipcc_calc_mon_mk3agcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar 6 no yes yes no

#Slabcntl
#ipcc_calc_mon_mk3agcm qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 51
/cs/datastore/csdar 13 no no yes no

#2xCO2
#ipcc_calc_mon_mk3agcm qv4 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 51
/cs/datastore/csdar 14 no no yes no

#Slabcntl
#ipcc_calc_mon_mk3agcm qv1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 31
/cs/datastore/csdar 9 no yes yes no

#2xCO2
#ipcc_calc_mon_mk3agcm qv2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 11
/cs/datastore/csdar 10 no yes yes no

#20C3M
#ipcc_calc_mon_mk3agcm n20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 131 10 131
251 /cs/datastore/csdar 15 no yes yes no

#20C3M
#ipcc_calc_mon_mk3agcm o20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 141 10 141
261 /cs/datastore/csdar 19 no yes yes no

#PIcntrl - run1 - mk3_5d
#81
#cy=1701-2460,my=1-760
#ipcc_calc_mon_mk3agcm N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 81 1161
/cs/datastore/csdar 16 no yes yes no
#ipcc_calc_mon_mk3agcm N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 981
```

```
1161 /cs/datastore/csdar 16 no yes yes no

#20C3M - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 171
291 /cs/datastore/csdar 20 no yes yes no

#SRESA2 - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 301
391 /cs/datastore/csdar 21 no yes yes no

#1pctto2x - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm M35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 171 10 171
241 /cs/datastore/csdar 22 no yes yes no

#SRESA1B - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 301
591 /cs/datastore/csdar 23 no yes yes no

#SRESB1 - run1 - mk3_5d
#my=301-400
#ipcc_calc_mon_mk3agcm E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 301
591 /cs/datastore/csdar 24 no yes yes no

#Commit - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 301
391 /cs/datastore/csdar 26 no yes yes no

#20C3M - run2 - mk3_5d
#ipcc_calc_mon_mk3agcm F35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 191 10 191
311 /cs/datastore/csdar 27 no yes yes no

#20C3M - run3 - mk3_5d
#ipcc_calc_mon_mk3agcm G35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 211 10 211
331 /cs/datastore/csdar 28 no yes yes no

#Slabcntl - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm qa1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
131 /cs/datastore/csdar 29 no yes yes no

#2xCO2 - run1 - mk3_5d
#ipcc_calc_mon_mk3agcm qa2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
131 /cs/datastore/csdar 30 no yes yes no

#PIcntrl - run1 - mk3_5A
#cy=1701-2460,my=1-240
#ipcc_calc_mon_mk3agcm C11 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 111
251 /cs/datastore/csdar 31 no yes yes no

#PDcntrl - run2 - mk3_5A
#ipcc_calc_mon_mk3agcm C12 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 61
/cs/datastore/csdar 36 no yes yes no
};#vars

set vars {t4d q4d r4d u4d v4d w4d z4d}
set vars {c4d};#cl iflg=13/14/15/19/20/21/22/23/24 only" 16
```

```
foreach v $vars {

if {$v == "w4d" || $v == "z4d" || $v == "c4d"} {
set ps ""
} else {
set ps "ps"
};#v

if {$v == "c4d"} {
set mask "no"
} else {
set mask "yes"
};#v

file mkdir /work/IPCC/$v

#PIcntrl - run1
#cy=1751-2250, my=1-500
#cy=1871-2250, my=121-500, but only want
#ipcc_calc_4dmon_mk3agcm_new uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10
351 361 /cs/datastore/csdar 2 no $ps yes yes $mask no

#PIcntrl - run2
#ipcc_calc_4dmon_mk3agcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar 7 no $ps yes yes $mask no

#20C3M
#ipcc_calc_4dmon_mk3agcm_new m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10
121 241 /cs/datastore/csdar 3 no $ps yes yes $mask no

#SRESA2 - run1
#cy=2001-2250, my=251-500
#ipcc_calc_4dmon_mk3agcm_new m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10
351 361 /cs/datastore/csdar 12 no $ps yes yes $mask no

#Commit
#ipcc_calc_4dmon_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
341 /cs/datastore/csdar 5 no $ps yes yes $mask no

#delete later
#ipcc_calc_4dmon_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 351
491 /cs/datastore/csdar 3 no $ps yes yes $mask no

#SRESA1B
#ipcc_calc_4dmon_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
441 /cs/datastore/csdar 8 no $ps yes yes $mask no

#SRESB1
#ipcc_calc_4dmon_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 251
541 /cs/datastore/csdar 1 no $ps yes yes $mask no

#1pctto2x
#ipcc_calc_4dmon_mk3agcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar 6 no $ps yes yes $mask no

#Slabcntl
```

```
#ipcc_calc_4dmon_mk3agcm qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 51
/cs/datastore/csdar 13 no $ps yes yes $mask no
#ipcc_calc_4dmon_mk3agcm_new qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 2 1
1 /cs/datastore/csdar 13 no $ps yes yes $mask no

#2xCO2
#ipcc_calc_4dmon_mk3agcm qv4 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 51
/cs/datastore/csdar 14 no $ps yes yes $mask no

#Slabcntl
#ipcc_calc_4dmon_mk3agcm qv1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 41
/cs/datastore/csdar 9 no $ps yes yes $mask no

#2xCO2
#ipcc_calc_4dmon_mk3agcm qv2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 21
/cs/datastore/csdar 10 no $ps yes yes $mask no

#20C3M - run2 - mk3_0
#ipcc_calc_4dmon_mk3agcm n20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 131 10 131
251 /cs/datastore/csdar 15 no $ps yes yes $mask no

#20C3M - run3 - mk3_0
#ipcc_calc_4dmon_mk3agcm o20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 141 10 141
261 /cs/datastore/csdar 19 no $ps yes yes $mask no

#PIcntrl - run1 - mk3_5d
#spsf starts from year 261, ps/psf before that only...
#cy=1701-2500,my=1-800
#cy=1781-2500,my=81-800
#ipcc_calc_4dmon_mk3agcm_new N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10
81 1161 /cs/datastore/csdar 16 no $ps yes yes $mask no

#20C3M - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171
10 171 291 /cs/datastore/csdar 20 no $ps yes yes $mask no

#SRESA2 - run1 - mk3_5d
#cy=2001-2100,my=301-400
#ipcc_calc_4dmon_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
301 391 /cs/datastore/csdar 21 no $ps yes yes $mask no

#1pctto2x - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new M35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 171 10
171 241 /cs/datastore/csdar 22 no $ps yes yes $mask no

#SRESA1B - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10
301 591 /cs/datastore/csdar 23 no $ps yes yes $mask no

#my=301-591
#SRESB1 - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10
301 591 /cs/datastore/csdar 24 no $ps yes yes $mask no

#Commit - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10
```

```
301 391 /cs/datastore/csdar 26 no $ps yes yes $mask no

#20C3M - run2 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new F35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 191 10
191 311 /cs/datastore/csdar 27 no $ps yes yes $mask no

#20C3M - run3 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new G35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 211 10
211 331 /cs/datastore/csdar 28 no $ps yes yes $mask no

#Slabcntl - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new qa1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1
11 /cs/datastore/csdar 29 no $ps yes yes $mask no

#2xCO2 - run1 - mk3_5d
#ipcc_calc_4dmon_mk3agcm_new qa2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1
11 /cs/datastore/csdar 30 no $ps yes yes $mask no

};#vars

#have a choice of tsu (in histb) or tsua (in hista) files...
#sgu (or IPCC rsus NA).
#need to tell diff. b/w uas for monthly mean and daily...

set vars {tsu};#ts
set vars {uas};#uas iflg=o20
set vars {vas};#vas iflg=o20
set vars {hfl}
set vars {tsl};#tasmin
set vars {tsca};#tas
set vars {rtu};#rlut
set vars {psl};#psl
set vars {evp};#hfls
set vars {sgd};#rsds
set vars {rgd};#rlds iflg=13/14/15/19/27/28 only"
set vars {rsus};#rsus iflg=13/14/15/19/27/28 only based on sgd/sgn"
set vars {rlus};#rlus iflg=13/14/15/19/27/28 only based on rgd/rgn"
set vars {v01};#vas
set vars {u01};#uas
set vars {rnd};#pr
set vars {tsh};#tasmax

set time_freq "12hourly"
set time_freq "6hourly"
set time_freq "daily"

foreach v $vars {

file mkdir /work/IPCC/$v

#PIcntrl - run1
#ipcc_calc_day_mk3agcm_new uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10 211
241 /cs/datastore/csdar 2 no yes yes no $time_freq

####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm_new uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10 121
```

```
491 /cs/datastore/csdar 2 no yes yes no $time_freq

#PIcntrl - run2
#ipcc_calc_day_mk3agcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 181
191 /cs/datastore/csdar 7 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar 7 no yes yes no

#20C3M
#ipcc_calc_day_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 211
241 /cs/datastore/csdar 3 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 121
201 /cs/datastore/csdar 3 no yes yes no

#SRESA2 - run1
#ipcc_calc_day_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 296
306 /cs/datastore/csdar 12 no yes yes no
#ipcc_calc_day_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 331
341 /cs/datastore/csdar 12 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 10 251
491 /cs/datastore/csdar 12 no yes yes no

#Commit
#uncertain about this one:
#ipcc_calc_day_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 281
291 /cs/datastore/csdar 5 no yes yes no
#ipcc_calc_day_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 296
306 /cs/datastore/csdar 5 no yes yes no
#ipcc_calc_day_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 331
341 /cs/datastore/csdar 5 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
341 /cs/datastore/csdar 5 no yes yes no

#SRESA1B
#ipcc_calc_day_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 296
306 /cs/datastore/csdar 8 no yes yes no
#ipcc_calc_day_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 331
341 /cs/datastore/csdar 8 no yes yes no
#ipcc_calc_day_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 431
441 /cs/datastore/csdar 8 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 251
441 /cs/datastore/csdar 8 no yes yes no

#SRESB1
#ipcc_calc_day_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 296
306 /cs/datastore/csdar 1 no yes yes no
#ipcc_calc_day_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 331
341 /cs/datastore/csdar 1 no yes yes no
#ipcc_calc_day_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 431
441 /cs/datastore/csdar 1 no yes yes no
#ipcc_calc_day_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 531
```

```
541 /cs/datastore/csdar 1 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 191 10 191
541 /cs/datastore/csdar 1 no yes yes no

#1pctto2x
#ipcc_calc_day_mk3agcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 181
191 /cs/datastore/csdar 6 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 121
191 /cs/datastore/csdar 6 no yes yes no

#Slabcntl
#ipcc_calc_day_mk3agcm qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 41 51
/cs/datastore/csdar 13 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 51
/cs/datastore/csdar 13 no yes yes no

#2xCO2
#ipcc_calc_day_mk3agcm qv4 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 41 51
/cs/datastore/csdar 14 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm qv4 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 51
/cs/datastore/csdar 14 no yes yes no

#Slabcntl
#ipcc_calc_day_mk3agcm qv1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 41
/cs/datastore/csdar 9 no yes yes no
####this is just for tas/tasmin/tasmax/pr

#2xCO2
#ipcc_calc_day_mk3agcm qv2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 21
/cs/datastore/csdar 10 no yes yes no
####this is just for tas/tasmin/tasmax/pr

#20C3M
#ipcc_calc_day_mk3agcm n20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 131 10 221
251 /cs/datastore/csdar 15 no yes yes no

#20C3M
#ipcc_calc_day_mk3agcm_new o20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 141 10
231 261 /cs/datastore/csdar 19 no yes yes no $time_freq

#PIcntrl - run1 - mk3_5d
#1701-2320, my=81-700, for extremes...
#ipcc_calc_day_mk3agcm_new N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 81
1141 /cs/datastore/csdar 16 no yes yes no $time_freq
#1961-2000, my=261,300
#ipcc_calc_day_mk3agcm_new N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 10 261
291 /cs/datastore/csdar 16 no yes yes no $time_freq

#20C3M - run1 - mk3_5d
#1871-2000, my=171-300, for extremes...
#ipcc_calc_day_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
171 291 /cs/datastore/csdar 20 no yes yes no $time_freq
```

```
#1961-2000, my=261-300
#ipcc_calc_day_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
261 291 /cs/datastore/csdar 20 no yes yes no $time_freq

#SRESA2 - run1 - mk3_5d
#2001-2100, my=301-400, for extremes...
#ipcc_calc_day_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
301 391 /cs/datastore/csdar 21 no yes yes no $time_freq
#2046-2065, my=346-365
#ipcc_calc_day_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
346 356 /cs/datastore/csdar 21 no yes yes no $time_freq
#2081-2100, my=381-400
#ipcc_calc_day_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
381 391 /cs/datastore/csdar 21 no yes yes no $time_freq

#1pctto2x - run1 - mk3_5d
#2001-2100, my=171-250, for extremes...
#ipcc_calc_day_mk3agcm_new M35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 171 10
171 241 /cs/datastore/csdar 22 no yes yes no $time_freq
#2061-2080 + 20 years centred on doubling? 2080 is doubling and as we don't have any data aft
#ipcc_calc_day_mk3agcm_new M35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 171 10
231 241 /cs/datastore/csdar 22 no yes yes no $time_freq

#SRESA1B - run1 - mk3_5d, y414 m11 missing at moment.
#2001-2300, my=301-600, for extremes...
#ipcc_calc_day_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
301 501 /cs/datastore/csdar 23 no yes yes no $time_freq
#2046-2065, my=346-65
#ipcc_calc_day_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
346 356 /cs/datastore/csdar 23 no yes yes no $time_freq
#2081-2100, my=381-400
#ipcc_calc_day_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
381 391 /cs/datastore/csdar 23 no yes yes no $time_freq
#2181-2200, my=481-500
#ipcc_calc_day_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
481 491 /cs/datastore/csdar 23 no yes yes no $time_freq
#2281-2300
#ipcc_calc_day_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
581 591 /cs/datastore/csdar 23 no yes yes no $time_freq

#SRESB1 - run1 - mk3_5d
#2001-2300, my=301-600, for extremes...
#ipcc_calc_day_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
301 591 /cs/datastore/csdar 24 no yes yes no $time_freq
#2046-2065, my=346-365
#ipcc_calc_day_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10
346 356 /cs/datastore/csdar 24 no yes yes no $time_freq
#2081-2100, my=381-400
#ipcc_calc_day_mk3agcm E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 381
391 /cs/datastore/csdar 24 no yes yes no $time_freq
#2181-2200, my=481-500
#ipcc_calc_day_mk3agcm E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 481
491 /cs/datastore/csdar 24 no yes yes no $time_freq
#2281-2300
#ipcc_calc_day_mk3agcm E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 10 581
591 /cs/datastore/csdar 24 no yes yes no $time_freq
```

```
#Commit - run1 - mk3_5d
#2001-2100 my=301-400, for extremes...
#ipcc_calc_day_mk3agcm_new C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10 301
/cs/datastore/csdar 26 no yes yes no $time_freq
#2046-2065, my=346-365
#ipcc_calc_day_mk3agcm_new C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10
346 356 /cs/datastore/csdar 26 no yes yes no $time_freq
#2081-2100, my=381-400
#ipcc_calc_day_mk3agcm_new C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 10
381 391 /cs/datastore/csdar 26 no yes yes no $time_freq

#20C3M - run2 - mk3_5d
#1871-2000, my=191-320, for extremes...
#ipcc_calc_day_mk3agcm_new F35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 191 10
191 311 /cs/datastore/csdar 27 no yes yes no $time_freq
#1961-2000, my=281-320
#ipcc_calc_day_mk3agcm_new F35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 191 10
281 311 /cs/datastore/csdar 27 no yes yes no $time_freq

#20C3M - run3 - mk3_5d
#1871-2000, my=211-340, for extremes...
#ipcc_calc_day_mk3agcm_new G35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 211 10
211 331 /cs/datastore/csdar 28 no yes yes no $time_freq
#1961-2000, my=301-340
#ipcc_calc_day_mk3agcm_new G35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 211 10
301 331 /cs/datastore/csdar 28 no yes yes no $time_freq

#Slabcntl - run1 - mk3_5d
#ipcc_calc_day_mk3agcm_new qa1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 21
31 /cs/datastore/csdar 29 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm_new qa2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1
11 /cs/datastore/csdar 29 no yes yes no

#2xCO2 - run1 - mk3_5d
#ipcc_calc_day_mk3agcm_new qa1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 41
11 /cs/datastore/csdar 30 no yes yes no
####this is just for tas/tasmin/tasmax/pr
#ipcc_calc_day_mk3agcm_new qa2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1
11 /cs/datastore/csdar 30 no yes yes no

};#v

set time_freq "12hourly"
set time_freq "6hourly"
set time_freq "daily"

set vars {z4d};#not ipcc variable!
set vars {t4d q4d u4d v4d}

#PIcntrl - run1
for {set year 211} {$year <= 250} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 1
```

```
$year $year /cs/datastore/csdar 2 no "ps" yes yes yes no
};#v
};#year

#PIcntrl - run2
for {set year 181} {$year <= 200} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 1
$year $year /cs/datastore/csdar 7 no "ps" yes yes yes no
};#v
};#year

#20C3M
for {set year 211} {$year <= 250} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 1
$year $year /cs/datastore/csdar 3 no "ps" yes yes yes no
};#v
};#year

#SRESA2 - run1
for {set year 296} {$year <= 315} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 1
$year $year /cs/datastore/csdar 12 no "ps" yes yes yes no
};#v
};#year
for {set year 331} {$year <= 350} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121
1 $year $year /cs/datastore/csdar 3 no "ps" yes yes yes no
};#v
};#year

#Commit
#used to be 281/300 now 296/306
for {set year 296} {$year <= 315} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251
1 $year $year /cs/datastore/csdar 5 no "ps" yes yes yes no
};#v
};#year
for {set year 331} {$year <= 350} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1
$year $year /cs/datastore/csdar 5 no "ps" yes yes yes no
};#v
};#year

#SRESA1B
```

```
for {set year 296} {$year <= 315} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1
$year $year /cs/datastore/csdar 8 no "ps" yes yes yes no
};#v
};#year
for {set year 331} {$year <= 350} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1
$year $year /cs/datastore/csdar 8 no "ps" yes yes yes no
};#v
};#year
for {set year 431} {$year <= 450} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1
$year $year /cs/datastore/csdar 8 no "ps" yes yes yes no
};#v
};#year

#SRESB1
for {set year 297} {$year <= 316} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 1
$year $year /cs/datastore/csdar 1 no "ps" yes yes yes no
};#v
};#year
for {set year 332} {$year <= 351} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 1
$year $year /cs/datastore/csdar 1 no "ps" yes yes yes no
};#v
};#year
for {set year 432} {$year <= 451} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 1
$year $year /cs/datastore/csdar 1 no "ps" yes yes yes no
};#v
};#year
for {set year 532} {$year <= 551} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 1
$year $year /cs/datastore/csdar 1 no "ps" yes yes yes no
};#v
};#year

#1pctto2x
for {set year 181} {$year <= 200} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
```

```
#ipcc_calc_4dday_mk3agcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 1
$year $year /cs/datastore/csdar 6 no "ps" yes yes yes no
};#v
};#year

#Slabcntl
for {set year 41} {$year <= 60} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 1 $year
$year /cs/datastore/csdar 13 no "ps" yes yes yes no
};#v
};#year

#2xCO2
for {set year 41} {$year <= 60} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm qv4 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 1 $year
$year /cs/datastore/csdar 14 no "ps" yes yes yes no
};#v
};#year

#Slabcntl
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm qv1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 41
/cs/datastore/csdar 9 no "ps" yes yes yes no
};#v

#2xCO2
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm qv2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 21
/cs/datastore/csdar 10 no "ps" yes yes yes no
};#v

#20C3M
for {set year 221} {$year <= 260} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm n20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 131 1
$year $year /cs/datastore/csdar 15 no "ps" yes yes yes no
#return
};#v
};#year

#20C3M
for {set year 231} {$year <= 270} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm o20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 141 1
$year $year /cs/datastore/csdar 19 no "ps" yes yes yes no
#return
};#v
};#year
```

```
#PIcntrl - run1 - mk3_5d
#1961-2000
for {set year 261} {$year <= 300} {incr year} {
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new N35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1701 1 1
$year $year /cs/datastore/csdar 16 no "ps" yes yes yes no $time_freq
};#v
};#year

#20C3M - run1 - mk3_5d
#1961-2000, my=261-300
for {set year 261} {$year <= 300} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 20 no "ps" yes yes yes no $time_freq
};#v
};#year

#SRESA2 - run1 - mk3_5d
#2046-2065, my=346-365
for {set year 346} {$year <= 346} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 21 no "ps" yes yes yes no $time_freq
};#v
};#year
#2081-2100
for {set year 381} {$year <= 400} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new B35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 21 no "ps" yes yes yes no $time_freq
};#v
};#year

#1pctto2x - run1 - mk3_5d
#2061-2080 + 20 years centred on doubling? 2080 is doubling and as we don't have any data aft
#231 250
for {set year 231} {$year <= 250} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new M35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 171 1
$year $year /cs/datastore/csdar 22 no "ps" yes yes yes no $time_freq
};#v
};#year

#SRESA1B - run1 - mk3_5d
#2046-2065
for {set year 346} {$year <= 365} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 23 no "ps" yes yes yes no $time_freq
};#v
};#year
#2081-2100
for {set year 381} {$year <= 400} {incr year} {
```

```
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 23 no "ps" yes yes yes no $time_freq
};#v
};#year
#2181-2200
for {set year 481} {$year <= 500} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 23 no "ps" yes yes yes no $time_freq
};#v
};#year
#cy=2281-2300, my=581-600
for {set year 581} {$year <= 600} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new D35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 23 no "ps" yes yes yes no $time_freq
};#v
};#year

#SRESB1 - run1 - mk3_5d
#2046-2065, my=346-365
for {set year 346} {$year <= 365} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 24 no "ps" yes yes yes no $time_freq
};#v
};#year
#2081-2100
for {set year 381} {$year <= 400} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 24 no "ps" yes yes yes no $time_freq
};#v
};#year
#2181-2200
for {set year 481} {$year <= 500} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 24 no "ps" yes yes yes no $time_freq
};#v
};#year
#2281-2300
for {set year 581} {$year <= 600} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new E35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 171 1
$year $year /cs/datastore/csdar 24 no "ps" yes yes yes no $time_freq
};#v
};#year

#Commit - run1 - mk3_5d
####2031-2050, my=331-350

#2046-2065, my=346-365
for {set year 346} {$year <= 365} {incr year} {
foreach v $vars {
```

```
#ipcc_calc_4dday_mk3agcm_new C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 1
$year $year /cs/datastore/csdar 26 no "ps" yes yes yes no $time_freq
};#v
};#year
#2081-2100, my=381-400
for {set year 381} {$year <= 400} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new C35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 301 1
$year $year /cs/datastore/csdar 26 no "ps" yes yes yes no $time_freq
};#v
};#year

#20C3M - run2 - mk3_5d
#1961-2000, my=281-320
for {set year 281} {$year <= 320} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new F35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 191 1
$year $year /cs/datastore/csdar 27 no "ps" yes yes yes no $time_freq
};#v
};#year

#20C3M - run3 - mk3_5d
#1961-2000, my=301-340
for {set year 301} {$year <= 340} {incr year} {
foreach v $vars {
#ipcc_calc_4dday_mk3agcm_new G35 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 211 1
$year $year /cs/datastore/csdar 28 no "ps" yes yes yes no $time_freq
};#v
};#year

#Slabcntl - run1 - mk3_5d
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new qa1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1
11 /cs/datastore/csdar 29 no "ps" yes yes yes no
};#v

#2xCO2 - run1 - mk3_5d
foreach v $vars {
file mkdir /work/IPCC/$v
#ipcc_calc_4dday_mk3agcm_new qa2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1
11 /cs/datastore/csdar 30 no "ps" yes yes yes no
};#v

set vars {psl tsc rnd}
foreach v $vars {

file mkdir /work/IPCC/$v

#PIcntrl - run1
#ipcc_calc_3hrly_mk3agcm uk1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1751 1 10 211
250 /cs/datastore/csdar 2 no yes yes no

#PIcntrl - run2
#ipcc_calc_3hrly_mk3agcm ct3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10
121 151 /cs/datastore/csdar 7 no yes yes no
```

```
#20C3M
for {set year 211} {$year <= 220} {incr year} {
#ipcc_calc_3hrly_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 1
$year $year /cs/datastore/csdar 3 no yes yes no
};#year

#SRESA2 - run1
#ipcc_calc_3hrly_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1871 121 1 300
300 /cs/datastore/csdar 12 no yes yes no
#ipcc_calc_3hrly_mk3agcm m20 $v /cs/datastore/IPCC/IPCC/data /work/csdar 1871 121 1 350
350 /cs/datastore/csdar 12 no yes yes no

#Commit
#ipcc_calc_3hrly_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10
281 291 /cs/datastore/csdar 5 no yes yes no
#ipcc_calc_3hrly_mk3agcm s20 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 10 331
341 /cs/datastore/csdar 5 no yes yes no

#SRESA1B
#ipcc_calc_3hrly_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1 300
300 /cs/datastore/csdar 8 no yes yes no
#ipcc_calc_3hrly_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1 350
350 /cs/datastore/csdar 8 no yes yes no
#ipcc_calc_3hrly_mk3agcm a1b $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 251 1 400
400 /cs/datastore/csdar 8 no yes yes no

#SRESB1
#ipcc_calc_3hrly_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 10 281
301 /cs/datastore/csdar 1 no yes yes no
#ipcc_calc_3hrly_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 10 321
341 /cs/datastore/csdar 1 no yes yes no
#ipcc_calc_3hrly_mk3agcm st1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 1941 192 10 421
441 /cs/datastore/csdar 1 no yes yes no

#1pctto2x
#ipcc_calc_3hrly_mk3agcm cm3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 121 10 181
191 /cs/datastore/csdar 6 no yes yes no

#Slabcntl
#ipcc_calc_3hrly_mk3agcm qv3 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 1 /
cs/datastore/csdar 13 no yes yes no

#2xCO2
#ipcc_calc_3hrly_mk3agcm qv4 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 1
/cs/datastore/csdar 14 no yes yes no

#Slabcntl
#ipcc_calc_3hrly_mk3agcm qv1 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 41
/cs/datastore/csdar 9 no yes yes no

#2xCO2
#ipcc_calc_3hrly_mk3agcm qv2 $v /cs/datastore/IPCC/IPCC/data /work/IPCC/$v 2001 1 10 1 21
/cs/datastore/csdar 10 no yes yes no
};#v
```

```
return
exit
```

14.24 ipcc_dims

```
if {$ldim == "time"} {
nap "long_name = ‘time‘"
nap "standard_name = ‘time‘"
nap "axis = ‘T‘"

if {$calendar == "leap"} {
#think of ncep/era40/silo at here...
nap "units = ‘hours since 1-1-1 00:00:0.0‘"
nap "calendar = ‘gregorian‘"
} else {
  nap "calendar = ‘noleap‘"
  if {$time_freq == "annual"} {
  nap "units = ‘years since 0000-01-01 00:00:00‘"
  } else {
  nap "units = ‘days since 0000-01-01 00:00:00‘"
  }
}

nap "bounds = ’time_bnds’"
$long_name netcdf ${tdir}/$ofil $ldim:long_name
$standard_name netcdf ${tdir}/$ofil $ldim:standard_name
$axis netcdf ${tdir}/$ofil $ldim:axis
$units netcdf ${tdir}/$ofil $ldim:units
$calendar netcdf ${tdir}/$ofil $ldim:calendar

if {$time_freq == "3hourly"} {
if {$ovar == "pr"} {
$bounds netcdf ${tdir}/$ofil $ldim:bounds
}
} elseif {$time_freq != "invariant"} {
$bounds netcdf ${tdir}/$ofil $ldim:bounds
}

};#ldim

if {$zdim == "pressure" || $zdim == "plev"} {
nap "long_name = ‘plev‘"
nap "standard_name = ‘air_pressure‘"
nap "axis = ‘Z‘"
nap "units = ‘Pa‘"
nap "positive = ‘down‘"

if {$surface_pressure != ""} {
$long_name netcdf ${tdir}/$ofil plev:long_name
$standard_name netcdf ${tdir}/$ofil plev:standard_name
$axis netcdf ${tdir}/$ofil plev:axis
$units netcdf ${tdir}/$ofil plev:units
$positive netcdf ${tdir}/$ofil plev:positive
} else {
$long_name netcdf ${tdir}/$ofil $zdim:long_name
$standard_name netcdf ${tdir}/$ofil $zdim:standard_name
$axis netcdf ${tdir}/$ofil $zdim:axis
$units netcdf ${tdir}/$ofil $zdim:units
$positive netcdf ${tdir}/$ofil $zdim:positive
```

```
};#surface_pressure

} elseif {$zdim == "depth"} {
nap "long_name = ‘depth‘"
nap "standard_name = ‘depth‘"
nap "axis = ‘Z‘"
nap "units = ‘m‘"
nap "bounds = ’depth_bnds’"
nap "positive = ‘down‘"
$long_name netcdf ${tdir}/$ofil $zdim:long_name
$standard_name netcdf ${tdir}/$ofil $zdim:standard_name
$axis netcdf ${tdir}/$ofil $zdim:axis
$units netcdf ${tdir}/$ofil $zdim:units
$bounds netcdf ${tdir}/$ofil $zdim:bounds
$positive netcdf ${tdir}/$ofil $zdim:positive

} elseif {$zdim == "lev"} {
nap "long_name = ‘lev‘"
nap "standard_name = ‘atmosphere_hybrid_sigma_pressure_coordinate‘"
nap "axis = ‘Z‘"
nap "units = ‘1‘"
nap "positive = ‘down‘"
nap "bounds = ’lev_bnds’"
nap "formula = ‘p(n,k,j,i) = a(k)*p0 + b(k)*ps(n,j,i)‘"
nap "formula_terms = ‘p0: p0 a: a b: b ps: ps‘"

$long_name netcdf ${tdir}/$ofil $zdim:long_name
$standard_name netcdf ${tdir}/$ofil $zdim:standard_name
$axis netcdf ${tdir}/$ofil $zdim:axis
$units netcdf ${tdir}/$ofil $zdim:units
$positive netcdf ${tdir}/$ofil $zdim:positive
$bounds netcdf ${tdir}/$ofil $zdim:bounds
$formula netcdf ${tdir}/$ofil $zdim:formula
$formula_terms netcdf ${tdir}/$ofil $zdim:formula_terms
};#zdim

if {$xdim == "lon"} {
nap "long_name = ‘longitude‘"
nap "standard_name = ‘longitude‘"
nap "axis = ‘X‘"
nap "units = ‘degrees_east‘"
nap "bounds = ’lon_bnds’"
$long_name netcdf ${tdir}/$ofil $xdim:long_name
$standard_name netcdf ${tdir}/$ofil $xdim:standard_name
$axis netcdf ${tdir}/$ofil $xdim:axis
$units netcdf ${tdir}/$ofil $xdim:units
$bounds netcdf ${tdir}/$ofil $xdim:bounds
} elseif {$xdim == "region"} {
nap "strlen = f64(0 .. 151)"
nap "geo_region = reshape(
’atlantic_ocean  indian_ocean    pacific_ocean   global_ocean    ’,{4 16})"
$geo_region set dim region strlen
$geo_region netcdf $tdir/$ofil geo_region
nap "standard_name = ‘region‘"
nap "long_name = ‘Ocean Basin‘"
$standard_name netcdf $tdir/$ofil geo_region:standard_name
```

```
$long_name netcdf $tdir/$ofil geo_region:long_name
};#xdim

if {$ydim == "lat"} {
nap "standard_name = ‘latitude‘"
nap "long_name = ‘latitude‘"
nap "axis = ‘Y‘"
nap "units = ‘degrees_north‘"
nap "bounds = ’lat_bnds’"
$long_name netcdf ${tdir}/$ofil $ydim:long_name
$standard_name netcdf ${tdir}/$ofil $ydim:standard_name
$axis netcdf ${tdir}/$ofil $ydim:axis
$units netcdf ${tdir}/$ofil $ydim:units
$bounds netcdf ${tdir}/$ofil $ydim:bounds
};#ydim
```

14.25 ipcc_dinfo_histb_mk3agcm

ipcc_calc_4dday_mk3agcm.tcl

```
if {$iflg == 1} {

if { $ynow <= 385 } {
set mdir "$pdir/col414/cec006/STAB1/AGCM/histb.dir"
} else {
set mdir "$pdir/col414/STAB1_CHECK/AGCM/histb.dir"
}

} elseif {$iflg == 2} {

if { $ynow <= 260 } {
set mdir "$pdir/gor079/C_Mk3a.f0.dir/AGCM/histb.dir"
} else {
set mdir "$pdir/col414/cec006/Mk3.0/AGCM/histb.dir"
}

} elseif {$iflg == 3} {

set mdir "$pdir/col414/cec006/M20th/AGCM/histb.dir"

} elseif {$iflg == 4} {

set mdir "$pdir/cshxr/Ct_Mk3a.f0.dir/AGCM/histb.dir"

} elseif {$iflg == 5} {

set mdir "$pdir/col414/cec006/S20/AGCM/histb.dir"

} elseif {$iflg == 6} {

set mdir "$pdir/col414/cec006/CMIP/AGCM/histb.dir"

} elseif {$iflg == 7} {

set mdir "$pdir/col414/cec006/CMIPC/AGCM/histb.dir"

} elseif {$iflg == 8} {

set mdir "$pdir/hir020/a1b/AGCM/histb.dir"

} elseif {$iflg == 9} {

set mdir "$pdir/wat302/T63qv1/histb.dir"

} elseif {$iflg == 10} {

set mdir "$pdir/wat302/T63qv2/histb.dir"

} elseif {$iflg == 12} {

set mdir "$pdir/col414/cec006/M20th/AGCM/histb.dir"
```

```
} elseif {$iflg == 13} {
set mdir "$pdir/dix043/qv3/histb.dir"
} elseif {$iflg == 14} {
set mdir "$pdir/hir020/qv4/histb.dir"
} elseif {$iflg == 15} {
set mdir "$pdir/hir020/n20/AGCM/histb.dir"
} elseif {$iflg == 16} {
set mdir "$pdir/gor079/C_Mk3.5d.dir/AGCM/histb.dir"
} elseif {$iflg == 19} {
set mdir "$pdir/hir020/o20/AGCM/histb.dir"
} elseif {$iflg == 20} {
set mdir "$pdir/hir020/Mk3.5_DBI.dir/AGCM/histb.dir"
} elseif {$iflg == 21} {
set mdir "$pdir/hir020/Mk3.5_DBI.dir/AGCM/histb.dir"
} elseif {$iflg == 22} {
set mdir "$pdir/dix043/coupled_model/M35/AGCM/histb.dir"
} elseif {$iflg == 23} {
set mdir "$pdir/hir020/Mk3.5_A1B.dir/AGCM/histb.dir"
} elseif {$iflg == 24} {
set mdir "$pdir/dix043/coupled_model/E35/AGCM/histb.dir"
} elseif {$iflg == 26} {
set mdir "$pdir/dix043/coupled_model/C35/AGCM/histb.dir"
} elseif {$iflg == 27} {
set mdir "$pdir/hir020/coupled_model/F35/AGCM/histb.dir"
} elseif {$iflg == 28} {
set mdir "$pdir/col414/coupled_model/G35/AGCM/histb.dir"
} elseif {$iflg == 29} {
set mdir "$pdir/wat302/T63qa1/AGCM/histb.dir"
```

```
} elseif {$iflg == 30} {

set mdir "$pdir/wat302/T63qa2/AGCM/histb.dir"

}
```

14.26 ipcc_dinfo_hist_mk3agcm

ipcc_calc_day2mon.tcl,ipcc_calc_day2mon_wap.tcl, ipcc_calc_day_mk3agcm.tcl

```
if {$iflg == 1} {

if { $ynow <= 385 } {
set hdir "$pdir/col414/cec006/STAB1/AGCM/hist$hist.dir"
} else {
set hdir "$pdir/col414/STAB1_CHECK/AGCM/hist$hist.dir"
}

} elseif {$iflg == 2} {

if { $ynow <= 260 } {
set hdir "$pdir/gor079/C_Mk3a.f0.dir/AGCM/hist$hist.dir"
} else {
set hdir "$pdir/col414/cec006/Mk3.0/AGCM/hist$hist.dir"
}

} elseif {$iflg == 3 || $iflg == 12} {

set hdir "$pdir/col414/cec006/M20th/AGCM/hist$hist.dir"

} elseif {$iflg == 4} {

set hdir "$pdir/cshxr/Ct_Mk3a.f0.dir/AGCM/hist$hist.dir"

} elseif {$iflg == 5} {

set hdir "$pdir/col414/cec006/S20/AGCM/hist$hist.dir"

} elseif {$iflg == 6} {

set hdir "$pdir/col414/cec006/CMIP/AGCM/hist$hist.dir"

} elseif {$iflg == 7} {

set hdir "$pdir/col414/cec006/CMIPC/AGCM/hist$hist.dir"

} elseif {$iflg == 8} {

#this is ok for everything except 3hourly data and
#so testing in diff. directory.
set hdir "$pdir/hir020/a1b/AGCM/hist$hist.dir"

} elseif {$iflg == 9} {

set hdir "$pdir/wat302/T63qv1/hist$hist.dir"

} elseif {$iflg == 10} {

set hdir "$pdir/wat302/T63qv2/hist$hist.dir"

} elseif {$iflg == 11} {
```

```
set hdir "$pdir/ofa001/20c_solar/AGCM/hist$hist.dir"
} elseif {$iflg == 13} {
set hdir "$pdir/dix043/qv3/hist$hist.dir"
} elseif {$iflg == 14} {
set hdir "$pdir/hir020/qv4/hist$hist.dir"
} elseif {$iflg == 15} {
set hdir "$pdir/hir020/n20/AGCM/hist$hist.dir"
} elseif {$iflg == 16} {
set hdir "$pdir/gor079/C_Mk3.5d.dir/AGCM/hist$hist.dir"
} elseif {$iflg == 19} {
set hdir "$pdir/hir020/o20/AGCM/hist$hist.dir"
} elseif {$iflg == 20} {
set hdir "$pdir/hir020/Mk3.5_DBI.dir/AGCM/hist$hist.dir"
} elseif {$iflg == 21} {
set hdir "$pdir/hir020/Mk3.5_DBI.dir/AGCM/hist$hist.dir"
} elseif {$iflg == 22} {
set hdir "$pdir/dix043/coupled_model/M35/AGCM/hist$hist.dir"
} elseif {$iflg == 23} {
set hdir "$pdir/hir020/Mk3.5_A1B.dir/AGCM/hist$hist.dir"
} elseif {$iflg == 24} {
set hdir "$pdir/dix043/coupled_model/E35/AGCM/hist$hist.dir"
} elseif {$iflg == 26} {
set hdir "$pdir/dix043/coupled_model/C35/AGCM/hist$hist.dir"
} elseif {$iflg == 27} {
set hdir "$pdir/hir020/coupled_model/F35/AGCM/hist$hist.dir"
} elseif {$iflg == 28} {
set hdir "$pdir/col414/coupled_model/G35/AGCM/hist$hist.dir"
} elseif {$iflg == 29} {
```

```
set hdir "$pdir/wat302/T63qa1/AGCM/hist$hist.dir"

} elseif {$iflg == 30} {

set hdir "$pdir/wat302/T63qa2/AGCM/hist$hist.dir"

}
```

14.27 ipcc_dinfo_mon_mk3agcm

ipcc_calc_day2mon.tcl,ipcc_calc_day2mon_wap.tcl, ipcc_calc_mon_mk3agcm.tcl,ipcc_calc_4dmon_mk3agcm.tcl

```
if {$iflg == 1} {

if { $ynow <= 385 } {
set mdir "$pdir/col414/cec006/STAB1/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} else {
set mdir "$pdir/col414/STAB1_CHECK/AGCM/netcdf.[expr ($ynow-1)/20].dir"
}

} elseif {$iflg == 2} {

if { $ynow <= 260 } {
set mdir "$pdir/gor079/C_Mk3a.f0.dir/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} else {
set mdir "$pdir/col414/cec006/Mk3.0/AGCM/netcdf.[expr ($ynow-1)/20].dir"
}

} elseif {$iflg == 3} {

set mdir "$pdir/col414/cec006/M20th/AGCM/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 4} {

set mdir "$pdir/cshxr/Ct_Mk3a.f0.dir/AGCM/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 5} {

set mdir "$pdir/col414/cec006/S20/AGCM/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 6} {

set mdir "$pdir/col414/cec006/CMIP/AGCM/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 7} {

set mdir "$pdir/col414/cec006/CMIPC/AGCM/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 8} {

set mdir "$pdir/hir020/a1b/AGCM/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 9} {

set mdir "$pdir/wat302/T63qv1/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 10} {

set mdir "$pdir/wat302/T63qv2/netcdf.[expr ($ynow-1)/20].dir"

} elseif {$iflg == 11} {

set mdir "$pdir/ofa001/20c_solar/AGCM/netcdf.[expr ($ynow-1)/20].dir"
```

```
} elseif {$iflg == 12} {
set mdir "$pdir/col414/cec006/M20th/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 13} {
set mdir "$pdir/dix043/qv3/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 14} {
set mdir "$pdir/hir020/qv4/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 15} {
set mdir "$pdir/hir020/n20/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 16} {
set mdir "$pdir/gor079/C_Mk3.5d.dir/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 19} {
set mdir "$pdir/hir020/o20/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 20} {
set mdir "$pdir/hir020/Mk3.5_DBI.dir/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 21} {
set mdir "$pdir/hir020/Mk3.5_DBI.dir/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 22} {
set mdir "$pdir/dix043/coupled_model/M35/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 23} {
set mdir "$pdir/hir020/Mk3.5_A1B.dir/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 24} {
set mdir "$pdir/dix043/coupled_model/E35/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 26} {
set mdir "$pdir/dix043/coupled_model/C35/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 27} {
set mdir "$pdir/hir020/coupled_model/F35/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 28} {
set mdir "$pdir/col414/coupled_model/G35/AGCM/netcdf.[expr ($ynow-1)/20].dir"
```

```
} elseif {$iflg == 29} {
set mdir "$pdir/wat302/T63qa1/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 30} {
set mdir "$pdir/wat302/T63qa2/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 31} {
set mdir "$pdir/rot032/c63/C11/AGCM/netcdf.[expr ($ynow-1)/20].dir"
} elseif {$iflg == 36} {
set mdir "$pdir/rot032/c63/C12/AGCM/netcdf.[expr ($ynow-1)/20].dir"
}
```

14.28 ipcc_dinfo_mon_mk3ogcm

ipcc_calc_mon_mk3ogcm.tcl

```
if {$iflg == 1} {

  if { $ynow <= 385 } {
  set mdir "$pdir/col414/cec006/STAB1/MOM2/avg.dir"
  } else {
  set mdir "$pdir/col414/STAB1_CHECK/MOM2/avg.dir"
  #delete this change later...
  #set mdir "$pdir/col414/cec006/STAB1/MOM2/avg.dir"
  }

} elseif {$iflg == 2} {

  if { $ynow <= 260 } {
  set mdir "$pdir/gor079/C_Mk3a.f0.dir/MOM2/avg.dir"
  } else {
  set mdir "$pdir/col414/cec006/Mk3.0/MOM2/avg.dir"
  }

} elseif {$iflg == 3} {

set mdir "$pdir/col414/cec006/M20th/MOM2/avg.dir"

} elseif {$iflg == 4} {

set mdir "$pdir/cshxr/Ct_Mk3a.f0.dir/MOM2/avg.dir"

} elseif {$iflg == 5} {

set mdir "$pdir/col414/cec006/S20/MOM2/avg.dir"

} elseif {$iflg == 6} {

set mdir "$pdir/col414/cec006/CMIP/MOM2/avg.dir"

} elseif {$iflg == 7} {

set mdir "$pdir/col414/cec006/CMIPC/MOM2/avg.dir"

} elseif {$iflg == 8} {

set mdir "$pdir/hir020/a1b/MOM2/avg.dir"

} elseif {$iflg == 9} {

puts "Ocean fields do not exist for this run."
return
set mdir "$pdir/wat302/T63qv1/MOM2/avg.dir"

} elseif {$iflg == 10} {

puts "Ocean fields do not exist for this run."
return
```

```
set mdir "$pdir/wat302/T63qv2/MOM2/avg.dir"

} elseif {$iflg == 11} {

set mdir "$pdir/ofa001/20c_solar/MOM2/avg.dir"

} elseif {$iflg == 12} {

set mdir "$pdir/col414/cec006/M20th/MOM2/avg.dir"

} elseif {$iflg == 15} {

set mdir "$pdir/hir020/n20/MOM2/avg.dir"

} elseif {$iflg == 16} {

  if { $ynow <= 80 } {
  set mdir "$pdir/gor079/C_Mk3.5a.dir/MOM2/avg.dir"
  } else {
  set mdir "$pdir/gor079/C_Mk3.5d.dir/MOM2/avg.dir"
  }

} elseif {$iflg == 17} {

  set mdir "$pdir/gor079/C_Mk3oca.f0.dir/MOM2/avg.dir"

} elseif {$iflg == 19} {

set mdir "$pdir/hir020/o20/MOM2/avg.dir"

} elseif {$iflg == 20} {

set mdir "$pdir/hir020/Mk3.5_DBI.dir/MOM2/avg.dir"

} elseif {$iflg == 21} {

set mdir "$pdir/hir020/Mk3.5_DBI.dir/MOM2/avg.dir"

} elseif {$iflg == 22} {

set mdir "$pdir/dix043/coupled_model/M35/MOM2/avg.dir"

} elseif {$iflg == 23} {

set mdir "$pdir/hir020/Mk3.5_A1B.dir/MOM2/avg.dir"

} elseif {$iflg == 24} {

set mdir "$pdir/dix043/coupled_model/E35/MOM2/avg.dir"

} elseif {$iflg == 25} {

set mdir "$pdir/gor079/cshxr/NEWoc.3.dir/MOM2/avg.dir"

} elseif {$iflg == 26} {
```

```
set mdir "$pdir/dix043/coupled_model/C35/MOM2/avg.dir"

} elseif {$iflg == 27} {

set mdir "$pdir/hir020/coupled_model/F35/MOM2/avg.dir"

} elseif {$iflg == 28} {

set mdir "$pdir/col414/coupled_model/G35/MOM2/avg.dir"

} elseif {$iflg == 31} {

set mdir "$pdir/rot032/c63/C11/MOM2/avg.dir"

} elseif {$iflg == 36} {

set mdir "$pdir/rot032/c63/C12/MOM2/avg.dir"

}
```

14.29 intro

ipcc_calc_4dday_mk3agcm.tcl, ipcc_calc_4dmon_mk3agcm.tcl, ipcc_calc_day2mon.tcl, ipcc_calc_day2mon_wap.tcl, ipcc_calc_day_mk3agcm.tcl, ipcc_calc_mon_mk3agcm.tcl, ipcc_calc_mon_mk3ogcm.tcl, ipcc_ei_cdd.tcl, ipcc_ei_etr.tcl, ipcc_ei_fd.tcl, ipcc_ei_gsl.tcl, ipcc_ei_hwdi.tcl, ipcc_ei_pq95.tcl, ipcc_ei_r10.tcl, ipcc_ei_r5d.tcl, ipcc_ei_r95t.tcl, ipcc_ei_sdii.tcl, ipcc_ei_tmaxnorm.tcl, ipcc_ei_tn90.tcl, ipcc_ei_tnq90.tcl, ipcc_hfogo_ferret.tcl, ipcc_hfogo.tcl, ipcc_intro.tcl, ipcc_mrros.tcl, ipcc_orog.tcl, ipcc_psf.tcl, ipcc_qflux.tcl, ipcc_rhopoto.tcl, ipcc_rldscs.tcl, ipcc_rlus.tcl, ipcc_rsdscs.tcl, ipcc_rsdt.tcl, ipcc_rsuscs.tcl, ipcc_rsus.tcl, ipcc_rtmt.tcl, ipcc_sftgif.tcl, ipcc_sftlf.tcl, ipcc_snc.tcl, ipcc_snw.tcl, ipcc_stfmmc_ferret.tcl, ipcc_stfmmc.tcl, ipcc_zg.tcl, ipcc_zobt.tcl

```
if {$iflg < 1 && $iflg > 26} {
puts "           model experiment          run  model           calendar"
puts "                 name                name years           years"

puts "iflg=1  => Mk3.0 SRESB1              st1  190-392/386-501 1949-2260"
puts "iflg=2  => Mk3.0 PIcntrl             uk1  1-500           1751-2250 (nominal)"
puts "iflg=3  => Mk3.0 20C3M 1871 (SRESA2) m20  121-250         1871-2000"
puts "iflg=4  => Mk3.0 20C3M 1961 (SRESA2) uk2  121-260         1961-2100"
puts "iflg=5  => Mk3.0 Commit              s20  251-353         2001-2102"
puts "iflg=6  => Mk3.0 1pctto2x            cm3  121-200         2001-2080"
puts "iflg=7  => Mk3.0 PIcntrl             ct3  121-200         2001-2080 (nominal)"
puts "iflg=8  => Mk3.0 SRESA1B             a1b  251-450         2001-2200"
puts "iflg=9  => Mk3.0 Slabcntl            qv1  1-50            2001-2050"
puts "iflg=10 => Mk3.0 2xCO2               qv2  1-37            2001-2037"
puts "iflg=11 => Mk3.0 SRESB2              sol  151-312         1871-2032"
puts "iflg=12 => Mk3.0 SRESA2              m20  251-500         2001-2250"
puts "iflg=13 => Mk3.0 Slabcntl            qv3  1-60            2001-2060"
puts "iflg=14 => Mk3.0 2xCO2               qv4  1-60            2001-2060"
puts "iflg=15 => Mk3.0 20C3M 1871          n20  131-260         1871-2000"
puts "iflg=16 => Mk3.5d PIcntrl            N35  1-1150          1701-2850"
puts "iflg=17 => Mk3.0 Synch. Spinup       uk1  1701-1710       1741-1750"
puts "iflg=18 => Mk3.0 ASynch. Spinup      uk1  1210-1700       1241-1740"
puts "iflg=19 => Mk3.0 20C3M 1871          o20  141-270         1871-2000"
puts "iflg=20 => Mk3.5d 20C3M              B35  171-300         1871-2000"
puts "iflg=21 => Mk3.5d SRESA2             B35  301-400         2000-2100"
puts "iflg=22 => Mk3.5d 1pctto2x           M35  171-250         2001-2080"
puts "iflg=23 => Mk3.5d SRESA1B            D35  301-600         2001-2300"
puts "iflg=24 => Mk3.5d SRESB1             E35  301-500         2001-2200"
puts "iflg=25 => Mk3.5d Synch. Spinup      N35  1701-1710       8301-8305"
puts "iflg=26 => Mk3.5d Commit             C35  301-400         2001-2100"
puts "iflg=27 => Mk3.5d 20C3M              F35  191-320         1871-2000"
puts "iflg=28 => Mk3.5d 20C3M              G35  211-340         1871-2000"
puts "iflg=29 => Mk3.5d Slabcntl           qa1  21-40           2001-2020"
puts "iflg=30 => Mk3.5d 2xCO2              qa2  21-40           2001-2020"
puts "iflg=31 => Mk3.5A PIcntrl            C11  111-251         1811-1950"
puts "iflg=32 => Mk3.5d SRESA2             H35  321-450         2000-2100"
puts "iflg=33 => Mk3.5d SRESA2             I35  341-470         2000-2100"
puts "iflg=34 => Mk3.5d SRESB1             J35  321-450         2001-2100"
puts "iflg=35 => Mk3.5d SRESB1             K35  341-470         2001-2100"
puts "iflg=36 => Mk3.5A PDcntrl            C12  1-70            2001-2070"

return
}

set calendar noleap;#365 day years
```

```
if {$hostname == "oldfarrer"} {
set mach_pref "cherax/"
} else {
set mach_pref ""
}

if {$iflg == 1} {
set model_version mk3.0
set runname st1
set expname SRESB1
set ensnum run1
set fil1_idir "[glob ~col414/${mach_pref}STAB1_CHECK/AGCM]"
set fil1_year 190
set ybeg 192;set yend 542;set yinc 10;#1941 onwards
set ybas_year 1939
nap "realization = 1"

} elseif {$iflg == 2} {
set model_version mk3.0
set runname uk1
set expname PIcntrl
set ensnum run1
set fil1_idir "[glob ~col414/cec006/${mach_pref}Mk3.0/AGCM]"

set fil1_year 21;#AGCM avg files.
set ybeg 21;set yend 491;set yinc 10;#AGCM avg files.

set fil1_year 1;#MOM2 avg files.
set ybeg 1;set yend 491;set yinc 10;#MOM2 avg files.

set ybas_year 1751
nap "realization = 1"

} elseif {$iflg == 3} {
set model_version mk3.0
set runname m20
set expname 20C3M
set ensnum run1
set fil1_idir "[glob ~col414/cec006/${mach_pref}M20th/AGCM]"
set fil1_year 121
set ybas_year 1871
set ybeg 121;set yend 491;set yinc 10
nap "realization = 1"
} elseif {$iflg == 4} {
set model_version mk3.0
set runname uk2
set expname SRESA2
set ensnum run1
set fil1_idir "[glob ~cshxr/${mach_pref}Ct_Mk3a.f0.dir/AGCM]"
set fil1_year 121
set ybeg 121;set yend 251;set yinc 10
set ybas_year 1961
nap "realization = 2"
} elseif {$iflg == 5} {
set model_version mk3.0
```

```
set runname s20
set expname Commit
set ensnum run1
set fil1_idir "[glob ~col414/cec006/${mach_pref}S20/AGCM]"
set fil1_year 251
set ybeg 251;set yend 341;set yinc 10
set ybas_year 2001
nap "realization = 1"
} elseif {$iflg == 6} {
set model_version mk3.0
set runname cm3
set expname 1pctto2x
set ensnum run1
set fil1_idir "[glob ~col414/cec006/${mach_pref}CMIP/AGCM]"
set fil1_year 121
set ybeg 121;set yend 191;set yinc 10
set ybas_year 2001
nap "realization = 1"
} elseif {$iflg == 7} {
set model_version mk3.0
set runname ct3
set expname PIcntrl
set ensnum run2
set fil1_idir "[glob ~col414/cec006/${mach_pref}CMIP/AGCM]"
set fil1_year 121
set ybeg 121;set yend 191;set yinc 10
set ybas_year 2001
nap "realization = 2"
} elseif {$iflg == 8} {
set model_version mk3.0
set runname a1b
set expname SRESA1B
set ensnum run1
set fil1_idir "[glob ~hir020/${mach_pref}a1b/AGCM]"
set fil1_year 251
set ybeg 251;set yend 441;set yinc 10
set ybas_year 2001
nap "realization = 1"
} elseif {$iflg == 9} {
set model_version mk3.0
set runname qv1
set expname Slabcntl
set ensnum run1
set fil1_idir "[glob ~wat302/${mach_pref}T63qv1]"
set fil1_year 1
puts "Ocean fields do not exist for this run."
set ybeg 1;set yend 41;set yinc 10
set ybas_year 2001
nap "realization = 1"
} elseif {$iflg == 10} {
set model_version mk3.0
set runname qv2
set expname 2xCO2
set ensnum run1
set fil1_idir "[glob ~wat302/${mach_pref}T63qv2]"
set fil1_year 1
```

```
puts "Ocean fields do not exist for this run."
set ybeg 1;set yend 21;set yinc 10
set ybas_year 2001
nap "realization = 1"
} elseif {$iflg == 12} {
set model_version mk3.0
set runname m20
set expname SRESA2
set ensnum run1
set fil1_idir "[glob ~col414/cec006/${mach_pref}M20th/AGCM]"
set fil1_year 251
set ybas_year 2001
set ybeg 251;set yend 491;set yinc 10
nap "realization = 1"

} elseif {$iflg == 13} {
set model_version mk3.0
set runname qv3
set expname Slabcntl
set ensnum run1
set fil1_idir "[glob ~dix043/${mach_pref}qv3]"
set fil1_year 1
puts "Ocean fields do not exist for this run."
set ybeg 1;set yend 51;set yinc 10
set ybas_year 2001
nap "realization = 1"
} elseif {$iflg == 14} {
set model_version mk3.0
set runname qv4
set expname 2xCO2
set ensnum run1
set fil1_idir "[glob ~hir020/${mach_pref}qv4]"
set fil1_year 1
puts "Ocean fields do not exist for this run."
set ybeg 1;set yend 51;set yinc 10
set ybas_year 2001
nap "realization = 1"

} elseif {$iflg == 15} {
set model_version mk3.0
set runname n20
set expname 20C3M
set ensnum run2
set fil1_idir "[glob ~hir020/${mach_pref}n20/AGCM]"
set fil1_year 131
set ybas_year 1871
set ybeg 131;set yend 251;set yinc 10
nap "realization = 2"

} elseif {$iflg == 16} {
set model_version mk3.5
set runname N35
set expname PIcntrl
set ensnum run1
set fil1_idir "[glob ~gor079/${mach_pref}C_Mk3.5d.dir/AGCM]"
set fil1_year 1
```

```
set ybas_year 1701
set ybeg 81;set yend 1161;set yinc 10
nap "realization = 1"

} elseif {$iflg == 17} {
set model_version mk3.0
set runname uk1
set expname Spinup
set ensnum run1
set fil1_idir "[glob ~gor079/${mach_pref}C_Mk3oca.f0.dir/MOM2]"
set fil1_year 1701
set ybeg 1701;set yend 1710;set yinc 10
set ybas_year 1741
nap "realization = 1"
} elseif {$iflg == 18} {
set model_version mk3.0
set runname uk1
set expname Spinup
set ensnum run1
set fil1_idir "[glob ~gor079/${mach_pref}C_Mk3oc.f0.dir/MOM2]"
set fil1_year 1241
set ybeg 1241;set yend 1241;set yinc 500
set ybas_year 1241
nap "realization = 1"
} elseif {$iflg == 19} {
set model_version mk3.0
set runname o20
set expname 20C3M
set ensnum run3
set fil1_idir "[glob ~hir020/${mach_pref}o20/AGCM]"
set fil1_year 141
set ybas_year 1871
set ybeg 141;set yend 261;set yinc 10
nap "realization = 3"

} elseif {$iflg == 20} {
set model_version mk3.5
set runname B35
set expname 20C3M
set ensnum run1
set fil1_idir "[glob ~hir020/Mk3.5_DBI.dir/AGCM]"
set fil1_year 171
set ybas_year 1871
set ybeg 171;set yend 291;set yinc 10
nap "realization = 1"

} elseif {$iflg == 21} {
set model_version mk3.5
set runname B35
set expname SRESA2
set ensnum run1
set fil1_idir "[glob ~hir020/Mk3.5_DBI.dir/AGCM]"
set fil1_year 301
set ybas_year 2001
set ybeg 301;set yend 391;set yinc 10
nap "realization = 1"
```

```
} elseif {$iflg == 22} {
set model_version mk3.5
set runname M35
set expname 1pctto2x
set ensnum run1
set fil1_idir "[glob ~dix043/coupled_model/M35/AGCM]"
set fil1_year 171
set ybas_year 2001
set ybeg 171;set yend 241;set yinc 10
nap "realization = 1"

} elseif {$iflg == 23} {
set model_version mk3.5
set runname D35
set expname SRESA1B
set ensnum run1
set fil1_idir "[glob ~hir020/Mk3.5_A1B.dir/AGCM]"
set fil1_year 301
set ybas_year 2001
set ybeg 301;set yend 591;set yinc 10
nap "realization = 1"

} elseif {$iflg == 24} {
set model_version mk3.5
set runname E35
set expname SRESB1
set ensnum run1
set fil1_idir "[glob ~dix043/coupled_model/E35/AGCM]"
set fil1_year 301
set ybas_year 2001
set ybeg 301;set yend 591;set yinc 10
nap "realization = 1"

} elseif {$iflg == 25} {
set model_version mk3.5
set runname N35
set expname Spinup
set ensnum run1
set fil1_idir "[glob ~gor079/${mach_pref}cshxr/NEWoc.3.dir/MOM2]"
set fil1_year 8301
set ybeg 8301;set yend 8305;set yinc 5
set ybas_year 8301
nap "realization = 1"

} elseif {$iflg == 26} {
set model_version mk3.5
set runname C35
set expname Commit
set ensnum run1
set fil1_idir "[glob ~dix043/coupled_model/C35/AGCM]"
set fil1_year 301
set ybas_year 2001
set ybeg 301;set yend 391;set yinc 10
nap "realization = 1"
```

```
} elseif {$iflg == 27} {
set model_version mk3.5
set runname F35
set expname 20C3M
set ensnum run2
set fil1_idir "[glob ~hir020/coupled_model/F35/AGCM]"
set fil1_year 191
set ybas_year 1871
set ybeg 191;set yend 311;set yinc 10
nap "realization = 2"

} elseif {$iflg == 28} {
set model_version mk3.5
set runname G35
set expname 20C3M
set ensnum run3
set fil1_idir "[glob ~col414/coupled_model/G35/AGCM]"
set fil1_year 211
set ybas_year 1871
set ybeg 211;set yend 331;set yinc 10
nap "realization = 3"

} elseif {$iflg == 29} {
set model_version mk3.5
set runname qa1
set expname Slabcntl
set ensnum run1
set fil1_idir "[glob ~wat302/${mach_pref}T63qa1]"
set fil1_year 21
puts "Ocean fields do not exist for this run."
set ybeg 21;set yend 31;set yinc 10
set ybas_year 2001
nap "realization = 1"

} elseif {$iflg == 30} {
set model_version mk3.5
set runname qa2
set expname 2xCO2
set ensnum run1
set fil1_idir "[glob ~wat302/${mach_pref}T63qa2]"
set fil1_year 1
puts "Ocean fields do not exist for this run."
set ybeg 21;set yend 31;set yinc 10
set ybas_year 2001
nap "realization = 1"

} elseif {$iflg == 31} {
set model_version mk3.5A
set runname C11
set expname PIcntrl
set ensnum run1
set fil1_idir "[glob ~rot032/${mach_pref}/c63/C11/AGCM]"
set fil1_year 1
set ybas_year 1701
set ybeg 111;set yend 251;set yinc 10
nap "realization = 1"
```

```
} elseif {$iflg == 32} {
set model_version mk3.5
set runname H35
set expname SRESA2
set ensnum run2
set fil1_idir "[glob ~col414/coupled_model/H35/AGCM]"
set fil1_year 321
set ybas_year 1871
set ybeg 321;set yend 441;set yinc 10
nap "realization = 2"

} elseif {$iflg == 33} {
set model_version mk3.5
set runname I35
set expname SRESA2
set ensnum run3
set fil1_idir "[glob ~col414/coupled_model/I35/AGCM]"
set fil1_year 341
set ybas_year 1871
set ybeg 341;set yend 461;set yinc 10
nap "realization = 3"

} elseif {$iflg == 34} {
set model_version mk3.5
set runname J35
set expname SRESB1
set ensnum run2
set fil1_idir "[glob ~col414/coupled_model/J35/AGCM]"
set fil1_year 321
set ybas_year 1871
set ybeg 321;set yend 441;set yinc 10
nap "realization = 2"

} elseif {$iflg == 35} {
set model_version mk3.5
set runname K35
set expname SRESB1
set ensnum run3
set fil1_idir "[glob ~col414/coupled_model/J35/AGCM]"
set fil1_year 341
set ybas_year 1871
set ybeg 341;set yend 461;set yinc 10
nap "realization = 3"

} elseif {$iflg == 36} {
set model_version mk3.5A
set runname C12
set expname PDcntrl
set ensnum run1
set fil1_idir "[glob ~rot032/${mach_pref}/c63/C12/AGCM]"
set fil1_year 1
set ybas_year 2001
set ybeg 1;set yend 61;set yinc 10
nap "realization = 1"
```

```
} elseif {$iflg == 100} {
set calendar "leap"
set model_version ncep
set runname ncep
set expname ncep
set ensnum run1
} elseif {$iflg == 101} {
set calendar "leap"
set model_version era40
set runname era40
set expname era40
set ensnum run1
} elseif {$iflg == 102} {
set calendar "leap"
set model_version silo
set runname silo
set expname silo
set ensnum run1
} elseif {$iflg == 103} {
set calendar "leap"
set model_version era40hr
set runname era40hr
set expname era40hr
set ensnum run1
}
```

14.30 ipcc_mask

ipcc_calc_4dday_mk3agcm.tcl, ipcc_calc_4dmon_mk3agcm.tcl, ipcc_calc_day2mon_wap.tcl, ipcc_calc_day_mk3agcm.tcl, ipcc_calc_mon_mk3agcm.tcl, ipcc_rlus.tcl, ipcc_rsus.tcl

```
set m96x48 [glob ~col414/ocean/masks/Mk3_agcm96x48.nc]
set m64x32 [glob ~col414/ocean/masks/Mk3_agcm64x32.nc]
set m192x96 [glob ~col414/ocean/masks/Mk3_agcm.nc]

puts "Mask files= $m192x96"
puts "            $m96x48"
puts "            $m64x32"

dmget_tcl "$m192x96 $m96x48 $m64x32"

nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "xfine=cv(l1,1)"
nap "yfine=cv(l1,0)"

set l1s [$l1 shape]
set l2s [$l2 shape]
set l3s [$l3 shape]

set o1s [$o1 shape]
set o2s [$o2 shape]
set o3s [$o3 shape]
```

14.31 ipcc_scripts

ipcc_data_final.tcl

```
source ~IPCC/processing/tools/ipcc_calc_mon_mk3agcm.tcl
source ~IPCC/processing/tools/ipcc_calc_4dmon_mk3agcm.tcl
source ~IPCC/processing/tools/ipcc_calc_4dmon_mk3agcm_new.tcl
source ~IPCC/processing/tools/ipcc_calc_mon_mk3ogcm.tcl
source ~IPCC/processing/tools/ipcc_calc_day_mk3agcm.tcl
source ~IPCC/processing/tools/ipcc_calc_4dday_mk3agcm.tcl
source ~IPCC/processing/tools/ipcc_calc_4dday_mk3agcm_new.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~col414/tcl_fun/pad10000.tcl
source ~col414/tcl_fun/pad10.tcl
source ~col414/tcl_fun/dmget_tcl.tcl
source ~col414/tcl_fun/del_items_from_list.tcl
source ~col414/tcl_fun/axpc2.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl
source ~IPCC/processing/tools/fill_holesA.tcl
source ~IPCC/processing/tools/ipcc_calc_3hrly_mk3agcm.tcl
source ~IPCC/processing/tools/hostname.tcl
```

14.32 ipcc_sigma

ipcc_bnds_agcm.tcl

```
nap "p0 = f32(101320.0)"

nap "a = f64({
    0.002250904520
    0.010908229599
    0.027445109782
    0.051161329294
    0.081499498157
    0.117432065629
    0.156602006617
    0.194782429690
    0.226158613316
    0.244560019861
    0.245284186804
    0.226803046532
    0.191604108558
    0.145736687815
    0.097191431038
    0.053798826780
    0.021604938272
    0.004458161866
})";#a

nap "b = f64({
    0.993290933614
    0.967486832130
    0.918371076775
    0.848701496495
    0.761093094436
    0.658631034508
    0.545729954974
    0.428674360434
    0.315336585586
    0.213944781236
    0.131259023073
    0.070864991877
    0.032332791305
    0.011670719592
    0.002945743174
    0.000384986663
    0.000000000000
    0.000000000000
})";#b

nap "a_bnds_a = f64({
    0.000000000000
    0.004501809041
    0.017314650157
    0.037575569408
    0.064747089179
    0.098251907134
    0.136612224124
```

```
    0.176591789110
    0.212973070269
    0.239344156364
    0.249775883358
    0.240792490250
    0.212813602813
    0.170394614303
    0.121078761327
    0.073304100748
    0.034293552812
    0.008916323731
})";#a_bnds_a

nap "a_bnds_b = f64({
0.004501809041
0.017314650157
0.037575569408
0.064747089179
0.098251907134
0.136612224124
0.176591789110
0.212973070269
0.239344156364
0.249775883358
0.240792490250
0.212813602813
0.170394614303
0.121078761327
0.073304100748
0.034293552812
0.008916323731
0.000000000000
})";#a_bnds_b

nap "b_bnds_a = f64({
    1.000000000000
    0.986581867228
    0.948391797031
    0.888350356518
    0.809052636472
    0.713133552399
    0.604128516617
    0.487331393331
    0.370017327536
    0.260655843636
    0.167233718837
    0.095284327309
    0.046445656446
    0.018219926164
    0.005121513021
    0.000769973326
    0.000000000000
    0.000000000000
})";#a_bnds_a

nap "b_bnds_b = f64({
```

```
0.986581867228
0.948391797031
0.888350356518
0.809052636472
0.713133552399
0.604128516617
0.487331393331
0.370017327536
0.260655843636
0.167233718837
0.095284327309
0.046445656446
0.018219926164
0.005121513021
0.000769973326
0.000000000000
0.000000000000
0.000000000000
})";#a_bnds_b

nap "lev_bnds_a = f64({1.0 0.991084 0.965706 0.925926 0.8738 0.811385 0.740741 0.663923
0.58299 0.5 0.41701 0.336077 0.259259 0.188615 0.1262 0.0740741 0.0342936 0.00891632})"
;#lev_bnds_a

nap "lev_bnds_b = f64({0.991084 0.965706 0.925926 0.8738 0.811385 0.740741 0.663923
0.58299 0.5 0.41701 0.336077 0.259259 0.188615 0.1262 0.0740741 0.0342936 0.00891632 0.0})"
;#lev_bnds_b

nap "a_bnds = transpose(a_bnds_a /// a_bnds_b,{1 0})"
nap "b_bnds = transpose(b_bnds_a /// b_bnds_b,{1 0})"
nap "lev_bnds = transpose(lev_bnds_a /// lev_bnds_b,{1 0})"

$p0 set missing
$p0 set label "reference pressure for hybrid sigma"
$p0 set unit Pa

$a set missing
$a set label "hybrid sigma coordinate A coefficient for layer"
$a set dim lev

$b set missing
$b set label "hybrid sigma coordinate B coefficient for layer"
$b set dim lev

$a_bnds set missing
$a_bnds set label "hybrid sigma coordinate A coefficient for layer bounds"
$a_bnds set dim lev bnds

$b_bnds set missing
$b_bnds set label "hybrid sigma coordinate B coefficient for layer bounds"
$b_bnds set dim lev bnds

$lev_bnds set missing
$lev_bnds set label "atmosphere_hybrid_sigma_pressure_coordinate"
$lev_bnds set dim lev bnds
```

```
$p0 netcdf $tdir/$ofil p0
$a netcdf $tdir/$ofil a
$b netcdf $tdir/$ofil b
$a_bnds netcdf $tdir/$ofil a_bnds
$b_bnds netcdf $tdir/$ofil b_bnds
$lev_bnds netcdf $tdir/$ofil lev_bnds

nap "formula = ‘p(n,k,j,i) = a(k)*p0 + b(k)*ps(n,j,i)‘"
nap "formula_terms = ‘p0: p0 a: a_bnds b: b_bnds ps: ps‘"
$formula netcdf $tdir/$ofil lev_bnds:formula
$formula_terms netcdf $tdir/$ofil lev_bnds:formula_terms

nap "lev = f64{0.995542 0.978395 0.945816 0.899863 0.842593 0.776063 0.702332 0.623457
0.541495 0.458505 0.376543 0.297668 0.223937 0.157407 0.100137 0.0541838 0.0216049
0.00445816}"
$lev netcdf $tdir/$ofil lev
```

14.33 ipcc_tspace

ipcc_calc_day_mk3agcm.tcl,ipcc_calc_4dday_mk3agcm.tcl

```
if {$time_freq == "daily"} {
set tdenom 1
} elseif {$time_freq == "3hourly"} {
set tdenom 8
} elseif {$time_freq == "6hourly"} {
set tdenom 4
} elseif {$time_freq == "12hourly"} {
set tdenom 2
} else {
puts "Don't know that interval."
return
}

set tincre [expr 1.0/$tdenom]
```

14.34 xyzt_edges

ipcc_bnds_ogcm.tcl, ipcc_ei_cdd.tcl, ipcc_ei_etr.tcl, ipcc_ei_fd.tcl, ipcc_ei_gsl.tcl, ipcc_ei_hwdi.tcl, ipcc_ei_pq95.tcl, ipcc_ei_r10.tcl, ipcc_ei_r5d.tcl, ipcc_ei_r95t.tcl, ipcc_ei_sdii.tcl, ipcc_ei_tmaxnorm.tcl, ipcc_ei_tn90.tcl, ipcc_ei_tnq90.tcl, ipcc_hfogo.tcl, ipcc_orog.tcl, ipcc_qflux.tcl, ipcc_rhopoto.tcl, ipcc_rsdt.tcl, ipcc_scripts.tcl, ipcc_sftgif.tcl, ipcc_sftlf.tcl, ipcc_stfmmc.tcl, ipcc_zobt.tcl

```
proc xyzt_edges \
{d fix Ed_bnds Edlo Edhi Edb} {
upvar $Ed_bnds d_bnds $Edlo dlo $Edhi dhi $Edb db
nap "numd = nels(d)"
nap "numdm1 = numd-1"
nap "numdm2 = numd-2"
nap "midd = (d(0 .. (numd-2)) + d(1 .. (numd-1)))/2"
nap "d_bnds = reshape( d(0)-(midd(0)-d(0)) // reshape(transpose(midd///midd),
{[expr [$numdm1]*2]}) // d(numdm1)+(d(numdm1)-midd(numdm2)) ,{[$numd] 2})"
nap "dlo = d_bnds(,0)"
nap "dhi = d_bnds(,1)"
nap "db = dhi-dlo"

if {$fix != ""} {
$d_bnds set value [lindex $fix 0] "0,0"
$d_bnds set value [lindex $fix 1] "[expr [lindex [$d_bnds shape] 0]-1],1"
}

return
}
```

14.34.1 Tools/Annual

14.35 ipcc_orog

ipcc_orog.tcl is a Tcl-Nap script used to generate parameter *orog* defined in Table 16.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_orog"
set time_freq "invariant"

puts "Entering procedure $proc_name"

set ipccout "no"
set ipccout "yes"

source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set nbeg 0;set nend 0

set tdir /work/IPCC

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

set ivar orog
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/tools/ipcc_intro.tcl

set odir [glob ~IPCC/data]/${model_version}/$expname/[lindex $table 0]/$ensnum
;file mkdir $odir

set ifil histb.m12.y121.nc

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${runname}_${model_version}_${ensnum}.nc
}

set idir "[glob ~col414/cec006/CMIP/AGCM/histb.dir]"

puts "Input file=$idir/$ifil"
puts "Output file=$odir/$ofil"

file delete $tdir/$ofil

nap "va = [nap_get netcdf $idir/$ifil zht ","]"

nap "$ydim = f64([nap_get netcdf $idir/$ifil lat])"
nap "$xdim  = f64([nap_get netcdf $idir/$ifil lon])"

$va set dim $ydim $xdim
$va set coo $ydim $xdim
$va set label "$olgn"
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

source ~IPCC/processing/tools/ipcc_att.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

file copy -force $tdir/$ofil $odir/$ofil;file delete $tdir/$ofil

};#iflg

return
exit
```

14.36 ipcc_sftgif

ipcc_sftgif.tcl is a Tcl-Nap script used to generate parameter *sftgif* defined in Table 16.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_sftgif"
set time_freq "invariant"

puts "Entering procedure $proc_name"

set ipccout "no"
set ipccout "yes"

source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set nbeg 0;set nend 0

set tdir /work/IPCC

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

set ivar sftgif
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/tools/ipcc_intro.tcl

set odir [glob ~IPCC/data]/$model_version/$expname/[lindex $table 0]/$ensnum
;file mkdir $odir

set ifil Mk3_agcm.nc
set gfil glacier.txt

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${runname}_${model_version}_${ensnum}.nc
}

puts "Input file=[glob ~col414/ocean/masks]/$ifil"
puts "Output file=$odir/$ofil"

if {$iflg == [lindex $iflgs 0]} {
nap "v1 = [nap_get netcdf [glob ~col414/ocean/masks]/$ifil land "0,0,,"]"
nap "v1 = reshape(i16(0),{[$v1 shape]})"

set inf [open [glob ~IPCC/processing/text_directory/$gfil]]
set line [gets $inf]
set nx [string range [lindex $line 0] 0 end-1];set ny [lindex $line 1]
set line [gets $inf]
set nxs [string range [lindex $line 0] 0 end-1];set nys [lindex $line 1]
if {$nx != $nxs || $ny != $nys} {
puts "Dimension mismatch."
return
}
for {set j [expr $ny-1]} {$j >= 0} {incr j -1} {
set line [gets $inf]
for {set i 0} {$i < $nx} {incr i} {
$v1 set value [string range $line $i $i] "$j,$i"
};#i
};#j

nap "va = (v1==2)*$exter_scale-$exter_offset"

nap "$ydim = f64([nap_get netcdf [glob ~col414/ocean/masks]/$ifil ytav_j])"
nap "$xdim  = f64([nap_get netcdf [glob ~col414/ocean/masks]/$ifil xtav_i])"

$va set dim $ydim $xdim
$va set coo $ydim $xdim
$va set label "$olgn"
};#iflg

$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
```

```
source ~IPCC/processing/tools/ipcc_att.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl
file copy -force $tdir/$ofil $odir/$ofil;file delete $tdir/$ofil

};#iflg

return
exit
```

14.37 ipcc_sftlf

ipcc_sftlf.tcl is a Tcl-Nap script used to generate parameter *sftlf* defined in Table 16.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_sftlf"
set time_freq "invariant"

puts "Entering procedure $proc_name"

set ipccout "no"
set ipccout "yes"

source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set nbeg 0;set nend 0

set tdir /work/IPCC

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

set ivar sftlf
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/tools/ipcc_intro.tcl

set odir [glob ~IPCC/data]/${model_version}/$expname/[lindex $table 0]/$ensnum;file mkdir $od

set ifil Mk3_agcm.nc

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${runname}_${model_version}_${ensnum}.nc
}

puts "Input file=[glob ~col414/ocean/masks]/$ifil"
puts "Output file=$odir/$ofil"

file delete $tdir/$ofil $odir/$ofil

nap "va = isPresent([nap_get netcdf [glob ~col414/ocean/masks]/$ifil land "0,0,,"])*
$exter_scale-$exter_offset"
nap "va = f32(va)"

nap "$ydim = f64([nap_get netcdf [glob ~col414/ocean/masks]/$ifil ytav_j])"
nap "$xdim  = f64([nap_get netcdf [glob ~col414/ocean/masks]/$ifil xtav_i])"

$va set missing 1e20
$va set dim $ydim $xdim
$va set coo $ydim $xdim
$va set label "$olgn"
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

source ~IPCC/processing/tools/ipcc_att.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl
file rename -force $tdir/$ofil $odir/$ofil

};#iflg

return
exit
```

14.38 ipcc_zobt

ipcc_zobt.tcl is a Tcl-Nap script used to generate parameter *zobt* defined in Table 17.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~IPCC/processing/tools/hostname.tcl

set nbeg 0;set nend 0

set hostname [hostname]

set fix "yes"

if {$fix == "yes"} {
set ffix [glob ~gor079/cshxr/NEWoc.3.dir/MOM2/avg.dir/avg.m12.y5610.nc]
}

set proc_name "ipcc_zobt"
set time_freq "invariant"

puts "Entering procedure $proc_name"

set ipccout "no"
set ipccout "yes"

source ~/.tclshrc
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl
source ~col414/tcl_fun/dmget_tcl.tcl

set tdir /work/IPCC

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
```

```
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

if {$iflg == 13 || $iflg == 14} {
set fix "no";set ffix ""
nap "full_lat_ind = 0 .. 95";nap "full_lon_ind = 0 .. 191"
} else {
nap "full_lat_ind = 0 .. 188";nap "full_lon_ind = 1 .. 192"
}

set ivar zobt
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/tools/ipcc_intro.tcl

set odir [glob ~IPCC/data]/${model_version}/$expname/[lindex $table 0]/$ensnum;
file mkdir $odir

if {$iflg == 13 || $iflg == 14} {
set ifil stsu_cm3.y0121_y0140.nc
set idir "[glob ~cec006/CMIP/AGCM/netcdf.6.dir]"
} else {
set ifil avg.m12.y121.nc
set idir "[glob ~cec006/CMIP/MOM2/avg.dir]"
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${runname}_${model_version}_${ensnum}.nc
}

puts "Input file=$idir/$ifil"
puts "Output file=$odir/$ofil"

dmget_tcl "$idir/$ifil $ffix"

file delete $tdir/$ofil

if {$iflg == 13 || $iflg == 14} {
nap "va = [get_cif -m 1e20 [glob ~IPCC/data/input/mlod.cif]]"
} else {
nap "z1 = (0 // [nap_get netcdf $idir/$ifil zwav_k])"
nap "va = z1([nap_get netcdf $idir/$ifil kmt "0,full_lat_ind,full_lon_ind"])"
nap "tvar=(va!=0)/(va!=0)"
nap "va=va*tvar"
}

nap "va = f32(va >>> 0.0)"
```

```
if {$iflg == 13 || $iflg == 14} {
nap "$ydim = f64([nap_get netcdf $idir/$ifil latitude "full_lat_ind"])"
nap "$xdim  = f64([nap_get netcdf $idir/$ifil longitude "full_lon_ind"])"

} else {
if {$fix == "yes"} {
nap "$ydim = f64([nap_get netcdf $ffix ytav_j "full_lat_ind"])"
} else {
nap "$ydim = f64([nap_get netcdf $idir/$ifil ytav_j "full_lat_ind"])"
}
nap "$xdim  = f64([nap_get netcdf $idir/$ifil xtav_i "full_lon_ind"])"
}

$va set dim $ydim $xdim
$va set coo $ydim $xdim
$va set missing 1e20
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

source ~IPCC/processing/tools/ipcc_att.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

file rename -force $tdir/$ofil $odir/$ofil

};#iflg

return
exit
```

14.38.1 Tools/Invariant

14.39 ipcc_qflux

ipcc_qflux.tcl is a Tcl-Nap script used to generate parameter *qflux* defined in Table 17.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier January 2005

source ~/.tclshrc
source ~col414/tcl_fun/pad10.tcl
source ~col414/tcl_fun/dmget_tcl.tcl

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

#qflux is constant for all experiments (function of month).

set proc_name "ipcc_qflux"
set time_freq "monthly"

puts "Entering procedure $proc_name"

source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set nbeg 0;set nend 0

set tdir /work/IPCC

set iflgs {13 14}

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set ivar qflux
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set odir [glob ~IPCC/data]/$model_version/$expname/[lindex $table 0]/$ensnum;
file mkdir $odir
set idir [glob ~IPCC/data]

set ofil ${ovar}_[lindex ${table} 0]_${expname}_${model_version}_${ensnum}.nc

puts "Output file=$odir/$ofil"

file delete -force $tdir/$ofil

set ifils [glob ~IPCC/data/qflux/qf??av1.cif]

dmget_tcl $ifils

foreach month {1 2 3 4 5 6 7 8 9 10 11 12} {
set monthm1 [expr $month-1]
set ifil qf[pad10 ${month}]av1.cif
```

```
nap "vb = [get_cif -m 1e20 $idir/qflux/$ifil]"
$va set value $vb "$monthm1,,"
};#month


nap "$ldim = f64({15.5 45.0 74.5 105.0 135.5 166.0 196.5 227.5 258.0 288.5 319.0 349.5})"

nap "$ydim = f64([nap_get netcdf $idir/sint_t63.nc lat])"
nap "$xdim  = f64([nap_get netcdf $idir/sint_t63.nc lon])"

$va set dim $ldim $ydim $xdim
$va set coo $ldim $ydim $xdim
$va set label "$olgn"
$va set missing f32(1e20)
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl
file rename $tdir/$ofil $odir/$ofil

};#iflg

return
exit
```

14.40 ipcc_rsdt

ipcc_rsdt.tcl is a Tcl-Nap script used to generate parameter *rsdt* defined in Table 6.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

#rsdt is constant for all experiments (function of month) except s20.

set proc_name "ipcc_rsdt"
set time_freq "monthly"

puts "Entering procedure $proc_name"

source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set nbeg 0;set nend 0

set tdir /work/IPCC

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {
```

```
source ~IPCC/processing/tools/ipcc_intro.tcl

set ivar rsdt
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set odir [glob ~IPCC/data]/${model_version}/$expname/[lindex $table 0]/$ensnum;
file mkdir $odir
set idir [glob ~IPCC/data/input]

set ifil sint_t63.nc
set ofil ${ovar}_[lindex ${table} 0]_${expname}_${model_version}_${ensnum}.nc

puts "Output file=$odir/$ofil"

file delete -force $tdir/$ofil

nap "va = [nap_get netcdf $idir/$ifil sit ",,"]"

nap "$ldim = f64({15.5 45.0 74.5 105.0 135.5 166.0 196.5 227.5 258.0 288.5 319.0 349.5})"

nap "$ydim = f64([nap_get netcdf $idir/$ifil lat])"
nap "$xdim  = f64([nap_get netcdf $idir/$ifil lon])"

$va set dim $ldim $ydim $xdim
$va set coo $ldim $ydim $xdim
$va set label "$olgn"
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl

source ~IPCC/processing/tools/ipcc_att.tcl

catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

file rename -force $tdir/$ofil $odir/$ofil

};#iflg

return
exit
```

14.40.1 Tools/Derived

14.41 ipcc_3dmask

ipcc_3dmask.tcl is a Tcl-Nap script which defines basin masks (Atlantic, Pacific, Indian and Global) for use in the scripts ipcc_stfmmc_ferret.tcl in Section 14.57 and ipcc_hfogo_ferret.tcl in Section 14.44 for both tracer and velocity grids on both the resolved scale and eddy induced origins. It can be easily modified to define masks of other sub-regions.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004
#Modified     May 6 2006

set proc_name "ipcc_3dmask"

source ~/.tclshrc
source ~col414/tcl_fun/dmget_tcl.tcl

#each number corresponds to an ocean region, by chosing
#the number you keep that basin as "wet".
#1 North Atlantic basin and Weddell sea.
#2 South Atlantic basin.
#3 North Indian basin.
#4 South Indian basin.
#5 North Pacific basin and Ross sea.
#6 South Pacific basin.
#7 Mediterranean sea.
#8 corresponds to no region.
#9 Various lakes.

set grid_top "uv"
set grid_top "ts"
set grid_top "uve"

if {$grid_top != "uv" && $grid_top == "ts" && $grid_top == "uve"} {
puts "Unknown output grid_top."
return
}

set type "stfmmc";#based on resolved scale velocity
set type "hfogo";#based on resolved scale velocity

set type "hfoge";#based on eddy induced velocity
set type "stfmme";#based on eddy induced velocity

if {$type == "stfmmc" && $grid_top == "uve" || $type == "hfogo" && $grid_top == "uve"} {
puts "Not allowed 1."
}

if {$type == "stfmme" && $grid_top == "uv" || $type == "stfmme" && $grid_top == "uv"
|| $type == "hfoge" && $grid_top == "uv" || $type == "hfoge" && $grid_top == "uv"} {
puts "Not allowed 2."
}

if {$type == "stfmme" && $grid_top == "ts" || $type == "stfmme" && $grid_top == "ts"
|| $type == "hfoge" && $grid_top == "ts" || $type == "hfoge" && $grid_top == "ts"} {
puts "Not allowed 3."
}
```

```
if {$grid_top == "uve"} {
set grid "uv"
}

if {$type == "stfmmc" || $type == "stfmme"} {
set ind {3}
set pac {5}
set atl {1 7}
} elseif {$type == "hfogo" || $type == "hfoge"} {
set ind {3 4}
set pac {5 6}
set atl {1 2 7}
}
set glo {1 2 3 4 5 6 7}

set mstr $glo;set nfil "[string toupper $grid]M3d.nc";set ofil
"${grid}_M3DR_GlobNoLak_${type}.nc";#globe with no lakes

nap "full_lat_ind = 0 .. 188";nap "full_lon_ind = 1 .. 192"

set idir [glob ~col414/ocean/masks]
set odir [glob ~IPCC/data/masks]
set fdir [glob ~gor079/cshxr/NEWoc.3.dir/MOM2/avg.dir]

set ffil avg.m12.y5610.nc
set mfil ${grid}_M3D.txt

puts "Output file=$odir/$ofil"

set mhan [open $odir/${mfil}]

puts "Input files=$odir/$mfil"
puts "            $idir/$nfil"
puts "            $fdir/$ffil"

dmget_tcl "$idir/$mfil $idir/$nfil $fdir/$ffil"

set ovar mask

set line [gets $mhan]
set nx [string range [lindex $line 0] 0 end-1];set ny [lindex $line 1]
set line [gets $mhan]
set nxs [string range [lindex $line 0] 0 end-1];set nys [lindex $line 1]

if {$nx != $nxs || $ny != $nys} {
puts "Dimension mismatch."
return
}

nap "land_tmp = reshape(i16(_),{$ny $nx})"

for {set j [expr $ny-1]} {$j >= 0} {incr j -1} {

set line [gets $mhan]
```

```
for {set i 0} {$i < $nx} {incr i} {

$land_tmp set value [string range $line $i $i] "$j,$i"

};#i

};#j

close $mhan

nap "land = land_tmp(full_lat_ind,full_lon_ind)"

nap "ocean_mask = land*0"

foreach m $mstr {
nap "ocean_mask = ocean_mask + (land == $m)"
};#m

file delete $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$nfil "mask" ",,full_lat_ind,full_lon_ind"]"

nap "time = f64([nap_get netcdf $fdir/$ffil Time])"
nap "time=time*0"
nap "depth = f64([nap_get netcdf $fdir/$ffil ztav_k])"

if {$grid_top == "uv"} {
nap "lat = f64([nap_get netcdf $fdir/$ffil yuav_j "full_lat_ind"])"
nap "lon = f64([nap_get netcdf $fdir/$ffil xuav_i "full_lon_ind"])"
} elseif {$grid_top == "ts"} {
nap "lat = f64([nap_get netcdf $fdir/$ffil ytav_j "full_lat_ind"])"
nap "lon = f64([nap_get netcdf $fdir/$ffil xtav_i "full_lon_ind"])"
} elseif {$grid_top == "uve"} {
nap "lat = f64([nap_get netcdf $fdir/$ffil yuav_j "full_lat_ind"])"
nap "lon = f64([nap_get netcdf $fdir/$ffil xtav_i "full_lon_ind"])"
}

$v1 set dim time depth lat lon
$v1 set coo time depth lat lon

$v1 set missing 1e20
$v1 set label "land-sea mask, 0=wet _=dry"
$v1 set unit ""

for {set k 0} {$k < 31} {incr k} {

#1=land/0=ocean
nap "layer = f32([nap_get netcdf $idir/$nfil mask "0,$k,,"])"

nap "mask = (layer==0 && ocean_mask==1)"

nap "mask = (mask/mask)-1"

$v1 set value mask ",$k,,"

};#k
```

```
$v1 netcdf -unlimited $odir/$ofil $ovar

nap "long_name = ‘longitude‘"
nap "standard_name = ‘longitude‘"
nap "axis = ‘X‘"
nap "units = ‘degrees_east‘"
$long_name netcdf $odir/$ofil lon:long_name
$standard_name netcdf $odir/$ofil lon:standard_name
$axis netcdf $odir/$ofil lon:axis
$units netcdf $odir/$ofil lon:units

nap "standard_name = ‘latitude‘"
nap "long_name = ‘latitude‘"
nap "axis = ‘Y‘"
nap "units = ‘degrees_north‘"
$long_name netcdf $odir/$ofil lat:long_name
$standard_name netcdf $odir/$ofil lat:standard_name
$axis netcdf $odir/$ofil lat:axis
$units netcdf $odir/$ofil lat:units

nap "long_name = ‘depth‘"
nap "standard_name = ‘depth‘"
nap "axis = ‘Z‘"
nap "units = ‘m‘"
nap "positive = ‘down‘"
$long_name netcdf $odir/$ofil depth:long_name
$standard_name netcdf $odir/$ofil depth:standard_name
$axis netcdf $odir/$ofil depth:axis
$units netcdf $odir/$ofil depth:units
$positive netcdf $odir/$ofil depth:positive

nap "long_name = ‘time‘"
nap "standard_name = ‘time‘"
nap "axis = ‘T‘"
nap "units = ‘days since 0000-01-01 00:00:00‘"
nap "calendar = ‘noleap‘"
$long_name netcdf $odir/$ofil time:long_name
$standard_name netcdf $odir/$ofil time:standard_name
$axis netcdf $odir/$ofil time:axis
$units netcdf $odir/$ofil time:units
$calendar netcdf $odir/$ofil time:calendar

for {set k 0} {$k < 31} {incr k} {

#1=land/0=ocean
nap "layer = f32([nap_get netcdf $idir/$nfil mask "0,$k,,"])"

nap "mask = (layer==0 && ocean_mask==1)"

nap "mask = (mask/mask)-1"

$mask set missing 1e20
$mask netcdf -index ",$k,," $odir/$ofil $ovar

$mask netcdf j.nc $ovar
```

```
};#k

return
exit
```

14.42 ipcc_calc_day2mon

ipcc_calc_day2mon.tcl is a Tcl-Nap script used to generate parameter *uas* and *vas* defined in Table 6. It is used to calculate various monthly variables from daily input.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set time_freq "monthly"

source ~col414/tcl_fun/pad10000.tcl
source ~col414/tcl_fun/pad10.tcl
source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

set clobber "yes"
set clobber "no"

set pdir /cs/datastore/csdar

set nmy 12

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

set ivars {v10ma}
set ivars {pev}
set ivars {pev v10ma q01 u01 v01 psf}

set ivars {psf}
set ivars {v01}
set ivars {u01}
set ivars {q01}

foreach ivar $ivars {

if {$ivar == "pev"} {
set ovar pev;set ofs "";set hist a;set long_name_tcl "Potential Evaporation";
set units_tcl "mm/day"
} elseif {$ivar == "v10ma"} {
set ovar V10;set ofs "";set hist a;set long_name_tcl "Surface Wind Speed";
set units_tcl "m/s"
} elseif {$ivar == "q01"} {
set ovar huss;set ofs "";set hist b;set long_name_tcl "Surface Specific Humidity";
set units_tcl "1"
} elseif {$ivar == "u01"} {
set ovar uas;set ofs "";set hist b;set long_name_tcl "Zonal Surface Wind Speed";
set units_tcl "m/s"
} elseif {$ivar == "v01"} {
set ovar vas;set ofs "";set hist b;set long_name_tcl "Meridional Surface Wind Speed";
set units_tcl "m/s"
} elseif {$ivar == "psf"} {
set ovar ps;set ofs "";set hist b;set long_name_tcl "Surface Pressure";
set units_tcl "Pa"
} else {
puts "Don't know that variable."
return
}

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

if {$ivar == "pev"} {
set exter_scale 0.1e1
} elseif {$ivar == "psf"} {
set exter_scale 0.1e1
}

source ~IPCC/processing/tools/ipcc_intro.tcl

set ndir_now -1

set ofil ${ofs}${ovar}_${runname}.nc
set opre ${ofs}${ovar}_${runname}

set ycnt 0
for {set ynow $ybeg} {$ynow <= [expr $yend+$yinc-1]} {incr ynow} {

source ~IPCC/processing/tools/ipcc_dinfo_hist_mk3agcm.tcl
```

```
source ~IPCC/processing/tools/ipcc_dinfo_mon_mk3agcm.tcl

set odir $mdir

set ofil_tmp "[glob $mdir/stsu_*]"
if {[llength $ofil_tmp] != 1} {
puts "Should only be one stsu file in output directory $mdir."
return
}
set y2 "[string last y $ofil_tmp]"
set y1 "[string last y $ofil_tmp [expr $y2-2]]"
if {$y1 != -1 && $y2 != -1} {
set ofil "${ofs}${ovar}_${runname}.[string range $ofil_tmp $y1 [expr $y2+4]].nc"
}

set check "[glob -nocomplain $odir/${opre}_*]"

if {[llength $check] == 2} {
puts "Something wrong as two similar files exist: $check"
return
}

if {[file exists $odir/${ofil}] == 1} {
puts "$odir/$ofil exists"
}

if {[file owned $odir] != 1} {
continue
}

if {[llength $check] == 1 && "$odir/$ofil" != $check} {
file copy $check $odir/$ofil
file delete $check
}

catch {unset hfils}
for {set mnow 1} {$mnow <= $nmy} {incr mnow} {

if {$runname == "st1"} {

if {$ynow > 385 } {
lappend hfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
} else {
lappend hfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.nc
}

} elseif {$runname == "C35" || $runname == "D35" || $runname == "E35" || $runname == "B35"
|| $runname == "N35" || $runname == "a1b" || $runname == "s20" || $runname == "qv3"
|| $runname == "qv4"} {
lappend hfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
} elseif {$runname == "qv1" || $runname == "qv2"} {
lappend hfils $hdir/hist$hist[pad10000 $ynow][pad10 $mnow].$runname.nc
} else {
lappend hfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.nc
}
}
```

```
for {set mnow 1} {$mnow <= $nmy} {incr mnow} {
set mnowm1 [expr $mnow-1]

if {$runname == "st1"} {

if {$ynow > 385 } {
set ifil hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
} else {
set ifil hist${hist}.m[pad10 $mnow].y$ynow.nc
}

} elseif {$runname == "C35" || $runname == "D35" || $runname == "E35" || $runname == "B35"
|| $runname == "N35" || $runname == "a1b" || $runname == "s20" || $runname == "qv3"
|| $runname == "qv4"} {
set ifil hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
} elseif {$runname == "qv1" || $runname == "qv2"} {
set ifil hist$hist[pad10000 $ynow][pad10 $mnow].$runname.nc
} else {
set ifil hist${hist}.m[pad10 $mnow].y$ynow.nc
}

set ndir_new [expr ($ynow-1)/20]
set ndir_newp1 [expr $ndir_new+1]

puts "ynow=$ynow mnow=$mnow"
puts "Input file=$ifil"
puts "Output file=$odir/$ofil"

if {[file owned $odir] != 1} {
continue
}

if {$ndir_new != $ndir_now && [file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

puts $hfils
dmget_tcl $hfils

if {$ndir_new != $ndir_now} {
file delete -force $odir/$ofil
set ndir_now $ndir_new
set ycnt 0

if {[expr $yend+$yinc-1-$ynow] < 20} {
set ylast [expr $yend+$yinc-1]
} else {
set ylast [expr $ndir_newp1*20]
}

set ydif [expr $ylast-$ynow+1]

set dimen [nap_get netcdf -dimension $hdir/$ifil $ivar]
set shape [nap_get netcdf -shape $hdir/$ifil $ivar]
set rank [nap_get netcdf -rank $hdir/$ifil $ivar]
```

```
puts $shape

set newshape "$ydif $nmy [lindex $shape 1] [lindex $shape 2]"

nap "va = reshape(f32(0),{$newshape})"

nap "month = i32(1..$nmy)"
nap "year = i32(1..$ydif)"
nap "lon = f32([nap_get netcdf $hdir/$ifil lon])"
nap "lat = f32([nap_get netcdf $hdir/$ifil lat])"

$va set dim year month lat lon
$va set coo year month lat lon

$va netcdf -unlimited $odir/$ofil $ovar

nap "units = 'degrees_north'"
$units netcdf $odir/$ofil lat:units

nap "units = 'degrees_east'"
$units netcdf $odir/$ofil lon:units

nap "units = 'months'"
$units netcdf $odir/$ofil month:units

nap "units = 'years'"
$units netcdf $odir/$ofil year:units

nap "long_name = '$long_name_tcl'"
nap "units = '$units_tcl'"
$long_name netcdf $odir/$ofil $ovar:long_name
$units netcdf $odir/$ofil $ovar:units

nap "history = `created by Mark Collier CSIRO [date_time_now]`"
$history netcdf $odir/$ofil :history

};#ndir_new

nap "va = [nap_get netcdf $hdir/$ifil $ivar]"
nap "va2 = am(va)*$exter_scale-$exter_offset"

$va2 set coo

$va2 set missing 1e20

$va2 netcdf -unlimited -index "$ycnt,$mnowm1,," $odir/$ofil $ovar

nap "y=$ynow"

$y netcdf -unlimited -index "$ycnt" $odir/$ofil "year"

};#mnow

incr ycnt
};#ynow
```

```
};#ivar

};#iflg

return
exit
```

14.43 ipcc_calc_day2mon_wap

ipcc_calc_day2mon$_{w}ap.tclisaTcl-NapscriptusedtogenerateparameterwapdefinedinTable 9. Itisusedtocalculatemonthlywapfromdailyinput.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier 20 October 2006

source ~IPCC/processing/tools/hostname.tcl
set hostname [hostname]

set proc_name "ipcc_calc_day2mon_wap"
puts "Entering procedure $proc_name"

set time_freq "monthly"

set clobber "yes"
set clobber "no"

#omega01 (in the calc_omega output file) is near earth's surface
#omega18 (in the calc_omega output file) is near top of atmosphere

source ~/.tclshrc
source ~col414/tcl_fun/pad10.tcl
source ~col414/tcl_fun/pad10000.tcl
source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

set hfine "yes"
set hfine "no";#to match z (put on fine grid during next step)

set tdir $::env(TMPDIR)

set number 0
while {$number < 100} {
set tfil omega$number.nc
puts $number
if {[file exists $tdir/$tfil] != 1} {
break
}
incr number
};#number
set tfil "omega$number.nc"
if {$tfil == "omega100.nc"} {
puts "Perhaps need for a cleanup of temporary omega files."
return
}
puts "Output temporary file: $tdir/$tfil"
set file [open $tdir/$tfil w]
puts $file
close $file

set tvar omega
set nmy 12

set pdir /cs/datastore/csdar
```

```
set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set ivar w4d;set ovar wap;set ofs "";set hist b;set long_name_tcl "Vertical Motion";
set units_tcl "Pa s-1"

nap "plev={1000.0 925.0 850.0 700.0 600.0 500.0 400.0 300.0 250.0 200.0 150.0 100.0 70.0
50.0 30.0 20.0 10.0}"

set iplev_nels [$plev nels]

nap "twelvezeros=12#0"

nap "twelvemonths=1..12"

nap "iplev=i16(plev)"

nap "yunits = 'degrees_north'"
nap "xunits = 'degrees_east'"
nap "munits = 'months'"
nap "yunits = 'years'"
nap "units = '$units_tcl'"
nap "history = `created by Mark Collier CSIRO [date_time_now]`"
nap "month = i32(1..$nmy)"

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
```

```
exter_offset land_sea_interp $time_freq var_derivation

if {$hfine == "yes"} {
source ~IPCC/processing/tools/ipcc_mask.tcl
}

set ycnt 0
for {set ynow $ybeg} {$ynow <= [expr $yend+$yinc-1]} {incr ynow} {
nap "yval=$ynow"
set ynowp1 [expr $ynow+1]

source ~IPCC/processing/tools/ipcc_dinfo_hist_mk3agcm.tcl
source ~IPCC/processing/tools/ipcc_dinfo_mon_mk3agcm.tcl

set odir $mdir

set ndir [expr ($ynow-1)/20]
set ndirp1 [expr $ndir+1]

set month_go 0

#month_go=1 means to process months for current year
#month_go=0 means to skip months for current year and move
#to next year.

set icheck 0
for {set ppp 0} {$ppp < $iplev_nels} {incr ppp} {

nap "siplev=iplev($ppp)";set siplevt "[$siplev v]"

set ofil_prefix ${ofs}${ovar}[pad10000 ${siplevt}]_${runname}

set ofil ${ofil_prefix}.nc

set check "[glob -nocomplain $odir/${ofil_prefix}*]"

if {[llength $check] == 2} {
puts "Something wrong as two similar files exist: $check"
return
}

if {[file exists $odir/$ofil] == 1} {
puts "$odir/$ofil exists"
incr icheck

nap "years=[nap_get netcdf $odir/$ofil year]"
set years_tcl [$years v]

if {$ppp == 0} {

set year_position [lsearch $years_tcl $ynow]

  if {$year_position != [expr [llength $years_tcl]-1]} {
#means that not last year element in years_tcl list
  set year_positionp1 [lsearch $years_tcl $ynowp1]
  }
```

```
  if {$year_position < 0} {
  set month_go 1
  }

#ppp!=0
} else {

  if {$year_position != [lsearch $years_tcl $ynow]} {
  puts "problem year_position=$year_position, [lsearch $years_tcl $ynow]"
  set month_go 1
  }

  if {$year_position != [expr [llength $years_tcl]-1]} {
  if {$year_positionp1 != [lsearch $years_tcl $ynowp1]} {
  puts "problem year_positionp1=$year_positionp1, [lsearch $years_tcl $ynow]"
  set month_go 1
  }
  }

};#ppp

nap "months=[nap_get netcdf $odir/$ofil month]"
nap "months_sum=sum(months)"
set months_sum_tcl [$months_sum v]

if {$months_sum_tcl != 78} {
#in here do not allow months_sum_tcl not to be anything other
#than 78.

  if {$ynowp1 == [lindex $years_tcl [lsearch $years_tcl $ynowp1]]} {
  set month_go 0
  } else {
  set month_go 1
  }
}

};#ofil exists

};#ppp

if {[file owned $odir] != 1} {
continue
}

if {$icheck == 0} {
#don't exist and so start writing..."
} elseif {$icheck > 0 && $icheck < $iplev_nels} {
#only some of them exist and so problem...
puts "icheck=$icheck, need to resolve problem."
return
} elseif {$icheck == $iplev_nels} {
#puts "must be 17 files, all ok."
} else {
puts "problem, should never get here."
return
```

```
}

if {$month_go == 0 && $icheck == $iplev_nels} {
incr ycnt;#skip this current year ynow
continue
}

catch {unset mfiles}
for {set mnow 1} {$mnow <= $nmy} {incr mnow} {
if {$runname == "B35" || $runname == "C35" || $runname == "D35" || $runname == "E35"
|| $runname == "F35" || $runname == "G35" || $runname == "M35" || $runname == "N35"
|| $runname == "a1b" || $runname == "s20" || $runname == "qv3" || $runname == "qv4"
|| $runname == "o20"} {
lappend mfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
} elseif {$runname == "st1"} {
  if {$ynow <= 385} {
  lappend mfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.nc
  } else {
  lappend mfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
  }
} elseif {$runname == "qv1" || $runname == "qv2"} {
lappend mfils $hdir/hist$hist[pad10000 $ynow][pad10 $mnow].$runname.nc
} else {
lappend mfils $hdir/hist${hist}.m[pad10 $mnow].y$ynow.nc
}
}

lappend mfils [glob ~dix043/bin/calc_omega]

for {set mnow 1} {$mnow <= $nmy} {incr mnow} {
set mnowm1 [expr $mnow-1]
nap "mval=$mnow"

if {$runname == "B35" || $runname == "C35" || $runname == "D35" || $runname == "E35"
|| $runname == "F35" || $runname == "G35" || $runname == "M35" || $runname == "N35"
|| $runname == "a1b" || $runname == "s20" || $runname == "qv3" || $runname == "qv4"
|| $runname == "o20"} {
set ifil hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
} elseif {$runname == "st1"} {
  if {$ynow <= 385} {
  set ifil hist${hist}.m[pad10 $mnow].y$ynow.nc
  } else {
  set ifil hist${hist}.m[pad10 $mnow].y$ynow.$runname.nc
  }
} elseif {$runname == "qv1" || $runname == "qv2"} {
set ifil hist$hist[pad10000 $ynow][pad10 $mnow].$runname.nc
} else {
set ifil hist${hist}.m[pad10 $mnow].y$ynow.nc
}

exec /cs/datastore/csdar/dix043/bin/calc_omega --oldformat -i $hdir/$ifil -o $tdir/$tfil
set dimen [nap_get netcdf -dimension $tdir/$tfil ${tvar}01]
set shape [nap_get netcdf -shape $tdir/$tfil ${tvar}01]
set rank [nap_get netcdf -rank $tdir/$tfil ${tvar}01]

if {$hfine == "yes"} {
```

```
nap "lon = f32(xfine)"
nap "lat = f32(yfine)"
} else {
nap "lon = f32([nap_get netcdf $tdir/$tfil lon])"
nap "lat = f32([nap_get netcdf $tdir/$tfil lat])"
}

if {[expr $yend+$yinc-1-$ynow] < 20} {
set ylast [expr $yend+$yinc-1]
} else {
set ylast [expr $ndirp1*20]
}

set ydif [expr $ylast-$ynow+1]

set newshape "$ydif $nmy [lindex $shape 1] [lindex $shape 2]"

if {$hfine == "yes"} {
set newshape [lreplace $newshape 2 3 [lindex $l1s 0] [lindex $l1s 1]]
}
set newshapeB "18 [lindex $shape 1] [lindex $shape 2]"

if {$icheck == 0} {
set ycnt 0
nap "year = i32(-1..-$ydif...-1)";#give it an abstract sequence...
nap "month = i32(-1..-$nmy...-1)";#give it an astract sequence...

nap "va = reshape(f32(0),{$newshape})"
$va set missing 1e20
$va set dim year month lat lon
$va set coo year month lat lon

for {set ppp 0} {$ppp < $iplev_nels} {incr ppp} {

nap "siplev=iplev($ppp)";set siplevt "[$siplev v]"

set ofil ${ofs}${ovar}[pad10000 ${siplevt}]_${runname}.nc

puts "ofil=$odir/$ofil"

nap "long_name = '$long_name_tcl at $siplevt'"

$va netcdf -unlimited $odir/$ofil "${ovar}[pad10000 ${siplevt}]"

[nap_get netcdf $odir/$ofil ${ovar}[pad10000 ${siplevt}]:_FillValue] netcdf $odir/$ofil ${ova

$yunits netcdf $odir/$ofil lat:units
$xunits netcdf $odir/$ofil lon:units
$munits netcdf $odir/$ofil month:units
$yunits netcdf $odir/$ofil year:units
$long_name netcdf $odir/$ofil ${ovar}[pad10000 ${siplevt}]:long_name
$units netcdf $odir/$ofil ${ovar}[pad10000 ${siplevt}]:units
$history netcdf $odir/$ofil :history
};#ppp

#icheck=0
```

```
} else {

if {$mnow == 1} {
for {set ppp 0} {$ppp < $iplev_nels} {incr ppp} {
nap "siplev=iplev($ppp)";set siplevt "[$siplev v]"
$twelvezeros netcdf $odir/${ofs}${ovar}[pad10000 ${siplevt}]_${runname}.nc month
};#ppp
}

};#icheck!=0

nap "vax = f64(reshape(0,{$newshapeB}))"

nap "psf = [nap_get netcdf $tdir/$tfil psf ",,"]"
nap "ps = am(psf)"

set lnt 0
foreach lv $lvars {
nap "v1 = [nap_get netcdf $tdir/$tfil ${tvar}[pad10 [expr $lnt+1]] ",,"]"
$v1 set missing 1e20
nap "v1a = am(v1)"

$vax set value $v1a "$lnt,,"

incr lnt
};#lnt

$vax set missing 1e20

nap "vax=vax*$exter_scale-$exter_offset"
$vax set missing 1e20

puts "plev=[$plev a]"
puts "ps=[$ps a]"

nap "va = mk3atPressure(plev,ps,vax)"
$va set missing 1e20

puts "va=[$va a]"

if {$hfine == "yes"} {
nap "va = va(,@yfine,@xfine)"
}

$va set dim plev lat lon
$va set coo plev lat lon

for {set ppp 0} {$ppp < $iplev_nels} {incr ppp} {
nap "siplev=iplev($ppp)";set siplevt "[$siplev v]"
set ofil ${ofs}${ovar}[pad10000 ${siplevt}]_${runname}.nc
puts "ofil=$odir/$ofil"
nap "vay = va($ppp,,)"

$vay netcdf -unlimited -index "$ycnt,$mnowm1,," $odir/$ofil "${ovar}[pad10000 ${siplevt}]"

if {$mnow == 1} {
```

```
$yval netcdf -unlimited -index "$ycnt" $odir/$ofil "year"
}

$mval netcdf -index "$mnowm1" $odir/$ofil "month"

$mval netcdf -index "$mnowm1" $odir/$ofil month
};#ppp

};#mnow

#now outside month loop, reset month vector...
$twelvemonths netcdf $odir/$ofil month

incr ycnt
};#ynow

file delete -force $tdir/$tfil

};#iflg

return
exit
```

14.44 ipcc_grid9to1

ipcc_mrros.tcl, ipcc_rldscs.tcl, ipcc_rlus.tcl, ipcc_rsdscs.tcl, ipcc_rsus.tcl, ipcc_rtmt.tcl

```
proc ipcc_grid9to1 {array} {

nap "year=cv(array,0)"
nap "month=cv(array,1)"

nap "lat=cv(array,2)"
nap "lon=cv(array,3)"

nap "area=area_on_globe(lat,lon)"

nap "array_shape = shape(array)"

nap "AREA=reshape(area,array_shape)"

nap "AREA_ma = moving_average(AREA,3,-1)"

nap "array_ma = moving_average(array*AREA,3,-1)"

nap "ratio = array_ma/AREA_ma"

nap "xred = 1 .. 190 ... 3"
nap "yred = 1 .. 94 ... 3"

nap "latitude = lat(yred)"
nap "longitude = lon(xred)"

nap "array_reduced = ratio(,,yred,xred)"

$array_reduced set dim year month latitude longitude
$array_reduced set coo year month latitude longitude

nap "result = array_reduced(,,@lat,@lon)"

};#ipcc_grid9to1
```

14.45 ipcc_hfogo_ferret

ipcc_hfogo_ferret.tcl is a Tcl-Nap script used to generate parameter *hfogo* defined in Table 10 using FERRET.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier July 2004

source ~/.tclshrc

set user $::env(USER)

set time_freq "monthly"
set proc_name "ipcc_hfogo_ferret"

set ferret_exec /tools/ferret_581_ia32/bin/ferret

puts "Entering procedure $proc_name"

source [glob ~col414/tcl_fun/dmget_tcl.tcl]
source [glob ~col414/tcl_fun/pad10000.tcl]
source [glob ~IPCC/processing/tools/ipcc_cmv.tcl]
source [glob ~IPCC/processing/tools/hostname.tcl]

set hostname [hostname]

set type "eddy";#eddy
set type "";#non-eddy

set ipccout "no"
set ipccout "yes"
set ipccin "no"
set ipccin "yes"

set clobber "yes"
set clobber "no"

if {$type == "eddy"} {
set ivar vedd
} else {
set ivar vvel
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set table_backup $table

if {$type == "eddy"} {
set ivar hfoge
} else {
set ivar hfogo
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation
```

```
set predir /cs/datastore/csdar/$user/ocean/ipcc/data
set predir /cs/datastore/IPCC/IPCC/data

#doesn't exist for 9/10

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source [glob ~IPCC/processing/tools/ipcc_intro.tcl]

if {$type == "eddy" && $iflg == 16 && $ybeg < 501} {
set ybeg 501
}

set odir $predir/$ovar
set odir [glob ~IPCC/data/$model_version/$ovar]

for {set ynow $ybeg} {$ynow <= $yend} {incr ynow $yinc} {

if {$ynow > 250 && $runname == "m20"} {
set expname "SRESA2"
} elseif {$ynow > 260 && $runname == "n20"} {
set expname "SRESA2"
} elseif {$ynow > 300 && $runname == "B35"} {
set expname "SRESA2"
}

set tdir /work/$user/$expname/[lindex $table_backup 0]/$ensnum
```

```
file mkdir $tdir

set idir $predir/$model_version/$expname/[lindex $table_backup 0]/$ensnum

if {$ipccin == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_IN [pad10000 $cbeg]
set YEND_IN [pad10000 $cend]
} else {
set YBEG_IN [pad10000 $ynow]
set YEND_IN [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_IN == $YEND_IN} {
set YSTR_IN "$YBEG_IN"
} else {
set YSTR_IN "$YBEG_IN-$YEND_IN"
}

if {$ipccout == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_OUT [pad10000 $cbeg]
set YEND_OUT [pad10000 $cend]
} else {
set YBEG_OUT [pad10000 $ynow]
set YEND_OUT [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_OUT == $YEND_OUT} {
set YSTR_OUT "$YBEG_OUT"
} else {
set YSTR_OUT "$YBEG_OUT-$YEND_OUT"
}

if {$ipccin == "yes" } {
  if {$type == "eddy"} {
  set vfil ve_[lindex ${table_backup} 0]_${YSTR_IN}_${expname}_${model_version}_
${ensnum}.nc
  } else {
  set vfil vo_[lindex ${table_backup} 0]_${YSTR_IN}_${expname}_${model_version}_
${ensnum}.nc
  }
  set tfil thetao_[lindex ${table_backup} 0]_${YSTR_IN}_${expname}_${model_version}_
${ensnum}.nc
} else {
  if {$type == "eddy"} {
  set vfil ve_[lindex ${table_backup} 0]_${YSTR_IN}_${runname}_${model_version}_
${ensnum}.nc
  } else {
  set vfil vo_[lindex ${table_backup} 0]_${YSTR_IN}_${runname}_${model_version}_
${ensnum}.nc
  }
  set tfil thetao_[lindex ${table_backup} 0]_${YSTR_IN}_${runname}_${model_version}_
${ensnum}.nc
```

```
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${runname}_${model_version}_${ensnum}.nc
}

puts "Input V    =$idir/$vfil"
puts "Input T    =$idir/$tfil"
puts "Output NHT =$odir/$ofil"
if {$type == "eddy"} {
puts "Temporary NHT =$tdir/nhte.nc"
} else {
puts "Temporary NHT =$tdir/nht.nc"
}

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

if {[file exists $idir/$vfil] == 0} {
puts "$idir/$vfil doesn't exist."
return
}

if {[file exists $idir/$tfil] == 0} {
puts "$idir/$tfil doesn't exist."
return
}

puts "Getting files..."
dmget_tcl "$idir/$vfil $idir/$tfil"
puts "Done."

if {$type == "eddy"} {
file delete -force $tdir/nhte.nc $odir/$ofil
catch {exec $ferret_exec -script nht.jnl $idir/$vfil $idir/$tfil $tdir ve}
puts "Renaming $tdir/nhte.nc to $odir/$ofil ..."
file rename -force $tdir/nhte.nc $odir/$ofil
} else {
file delete -force $tdir/nht.nc $odir/$ofil
catch {exec $ferret_exec -script nht.jnl $idir/$vfil $idir/$tfil $tdir vo}
puts "Renaming $tdir/nht.nc to $odir/$ofil ..."
file rename -force $tdir/nht.nc $odir/$ofil
}

};#ynow

};#iflg

return
exit
```

14.46 ipcc_hfogo

ipcc_hfogo.tcl is a Tcl-Nap script used to generate parameter *hfogo* defined in Table 10 computed in a previous step using ipcc$_hfogo_ferretdescribedinSection$ 14.44.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier August 2004

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_hfogo"
set time_freq "monthly"

puts "Entering procedure $proc_name"

set type "eddy";#eddy
set type "";#non-eddy

set ipccin "no"
set ipccin "yes"
set ipccout "no"
set ipccout "yes"

set clobber "yes"
set clobber "no"

source [glob ~col414/tcl_fun/pad10000.tcl]
source [glob ~IPCC/processing/tools/ipcc_cmv.tcl]
source [glob ~IPCC/processing/tools/xyzt_edges.tcl]

source ~/.tclshrc

set tdir /work/IPCC

set rank 4

set tregion {ATLANTIC INDIAN PACIFIC GLOBAL}

if {$type == "eddy"} {
set ivar vedd
} else {
set ivar vvel
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set table_backup $table

if {$type == "eddy"} {
set ivar hfoge
} else {
set ivar hfogo
}
```

```
ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set predir /cs/datastore/IPCC/IPCC/data/mk3.5

set idir $predir/$ovar

for {set ynow $ybeg} {$ynow <= $yend} {incr ynow $yinc} {

set nbeg $ybeg;set nend $yend

if {$ynow > 250 && $runname == "m20"} {
set expname "SRESA2"
} elseif {$ynow > 260 && $runname == "n20"} {
set expname "SRESA2"
} elseif {$ynow > 300 && $runname == "B35"} {
set expname "SRESA2"
}

set odir $predir/$expname/[lindex $table 0]/$ensnum;file mkdir $odir

if {$ipccin == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
```

```
set cend [expr $cbeg+$yinc-1]
set YBEG_IN [pad10000 $cbeg]
set YEND_IN [pad10000 $cend]
} else {
set YBEG_IN [pad10000 $ynow]
set YEND_IN [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_IN == $YEND_IN} {
set YSTR_IN "$YBEG_IN"
} else {
set YSTR_IN "$YBEG_IN-$YEND_IN"
}

if {$ipccout == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_OUT [pad10000 $cbeg]
set YEND_OUT [pad10000 $cend]
} else {
set YBEG_OUT [pad10000 $ynow]
set YEND_OUT [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_OUT == $YEND_OUT} {
set YSTR_OUT "$YBEG_OUT"
} else {
set YSTR_OUT "$YBEG_OUT-$YEND_OUT"
}

if {$ipccin == "yes" } {
set ifil ${ovar}_[lindex ${table} 0]_${YSTR_IN}_${expname}_${model_version}_${ensnum}.nc
} else {
set ifil ${ovar}_[lindex ${table} 0]_${YSTR_IN}_${runname}_${model_version}_${ensnum}.nc
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${runname}_${model_version}_${ensnum}.nc
}

puts "Input NHT  =$idir/$ifil"
puts "Output NHT =$odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

if {[file exists $idir/$ifil] == 0} {
puts "$idir/$ifil doesn't exist."
continue
}

file delete -force $tdir/$ofil $odir/$ofil
```

```
set shape [nap_get netcdf -shape $idir/$ifil [string toupper $ovar]_[lindex $tregion 0]]

for {set tnow 1} {$tnow <= [lindex $shape 0]} {incr tnow} {

set tnowm1 [expr $tnow-1]

puts "ynow=$ynow tnow=$tnow"

nap "t0 = [nap_get netcdf $idir/$ifil TIME "{$tnowm1}"]"

if {$tnow == 1} {
nap "hfogo = [nap_get netcdf $idir/$ifil [string toupper $ovar]_[lindex $tregion 0]
"{$tnowm1},"]"
nap "region=f64({0 1 2 3})"
$region set label global_ocean

set oshape "[lindex [$hfogo shape] 0] [$region shape] [lindex [$hfogo shape] end]"
nap "time = [nap_get netcdf $idir/$ifil [string toupper $ldim] "{$tnowm1}"]"
nap "lat = [nap_get netcdf $idir/$ifil [string toupper $ydim]]"
};#tnow=1

nap "hfogo = reshape(f32(_),{$oshape})"

$hfogo set dim $ldim region $ydim
$hfogo set coo $ldim region $ydim
$hfogo set missing f32(1e20)

if {$tnow == 1} {
$hfogo netcdf -unlimited -index "$tnowm1,," $tdir/$ofil "$ovar"
};#tnow

$t0 netcdf -unlimited -index "$tnowm1" $tdir/$ofil "time"

set rnt 0
foreach r $tregion {
nap "hfogo = [nap_get netcdf $idir/$ifil [string toupper $ovar]_$r "{$tnowm1},"]"

$hfogo netcdf -unlimited -index "$tnowm1,$rnt," $tdir/$ofil "$ovar"
incr rnt
};#r

if {$tnow == 1} {
catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}
source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
};#tnow=1

}

source ~IPCC/processing/tools/ipcc_bnds_ogcm.tcl

file rename -force $tdir/$ofil $odir/$ofil

};#ynow
```

```
};#iflg

return
exit
```

14.47 ipcc_mrros

ipcc_mrros.tcl is a Tcl-Nap script used to generate parameter *mrros* defined in Table 6.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

set proc_name "ipcc_mrros"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 per
set ivar2 run
set ovar mrros

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set m96x48 [glob ~col414/ocean/masks/Mk3_agcm96x48.nc]
set m64x32 [glob ~col414/ocean/masks/Mk3_agcm64x32.nc]
set m192x96 [glob ~col414/ocean/masks/Mk3_agcm.nc]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "Mask files= $m192x96"
puts "            $m96x48"
puts "            $m64x32"
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
```

```
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  #exit
  continue
  } else {
  #bit like noclobber - it exists already therefoer don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2 $m192x96 $m96x48 $m64x32"
file delete $tdir/$ofil $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]

set v1x [lindex $v1s 3]
set v1y [lindex $v1s 2]

set v2x [lindex $v2s 3]
set v2y [lindex $v2s 2]

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
}
if {$v2x == 64 && $v2y == 32} {
```

```
incr ires
}

if {$ires == 0} {
#all on full grid.
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {
#any one on reduced grid.

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "v2_full=ipcc_grid9to1(v2)"
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full=ipcc_grid9to1(v1)"
} else {
puts "Don't know that resolution."
return
}

};#ires

nap "v3 = f32(v2_full-v1_full)"
# Ensure there are no negative values from roundiff error
nap "v3 = v3 >>> 0.0"

$v3 set missing 1e20
$v3 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.48 ipcc_psf

ipcc_ps.tcl is a Tcl-Nap script used to generate a surface pressure parameter *ps* described in Table 19. It is used for masking out below surface values of parameters derived using ipcc_calc_4dmon_mk3agcm.tcl. The surface pressure files must exist before ipcc_calc_4dmon_mk3agcm.tcl is executed if masking is required.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_psf"
set time_freq "monthly"

puts "Entering procedure $proc_name"

set clobber "yes"
set clobber "no"

catch {unset land_sea_horiz_interp}
set pass 0

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
```

```
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set zfil zht_t63.nc
set zdir [glob ~IPCC/data/input]

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 psl
set ivar2 tsu

set ovar psf

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "Input files=$idir/$mfil2"
puts "Output file=$odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2 $zdir/$zfil"

file delete -force $tdir/$ofil $odir/$ofil

exec /cs/datastore/csdar/dix043/bin/calc_psf -p $idir/$mfil1 -t $idir/$mfil2 -z $zdir/$zfil
-o $tdir/$ofil
```

```
puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.49 ipcc_rhopoto

ipcc_rhopoto.tcl is a Tcl-Nap script used to generate parameter *rhopto* defined in Table 13.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier July 2004
#Last Modified August 19 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_rhopoto"
set time_freq "monthly"

set ipccout "no"
set ipccout "yes"
set ipccin "no"
set ipccin "yes"

set clobber "yes"
set clobber "no"

source ~/.tclshrc
source ~IPCC/processing/miscellaneous/unesco.tcl
source ~col414/tcl_fun/pad10000.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl
source ~col414/tcl_fun/dmget_tcl.tcl

set tdir /work/IPCC

set rank 4

set ivar rhopoto

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

#not appropriate as slab ocean.

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
```

```
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set predir /cs/datastore/IPCC/IPCC/data

for {set ynow $ybeg} {$ynow <= $yend} {incr ynow $yinc} {

set nbeg $ybeg;set nend $yend

if {$nbeg > 250 && $runname == "m20"} {
set expname "SRESA2"
} elseif {$nbeg > 260 && $runname == "n20"} {
set expname "SRESA2"
} elseif {$nbeg > 300 && $runname == "B35"} {
set expname "SRESA2"
}

set idir $predir/${model_version}/$expname/[lindex $table 0]/$ensnum
set odir $predir/${model_version}/$expname/[lindex $table 0]/$ensnum

if {$ipccin == "yes" } {
set cbeg [expr ${ybas_year}+$ynow-${fil1_year}]
set cend [expr $cbeg+$yinc-1]
set YBEG_IN [pad10000 $cbeg]
set YEND_IN [pad10000 $cend]
} else {
set YBEG_IN [pad10000 $ynow]
set YEND_IN [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_IN == $YEND_IN} {
set YSTR_IN "$YBEG_IN"
} else {
set YSTR_IN "$YBEG_IN-$YEND_IN"
}

if {$ipccout == "yes" } {
set cbeg [expr ${ybas_year}+$ynow-${fil1_year}]
set cend [expr $cbeg+$yinc-1]
set YBEG_OUT [pad10000 $cbeg]
set YEND_OUT [pad10000 $cend]
} else {
```

```
set YBEG_OUT [pad10000 $ynow]
set YEND_OUT [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_OUT == $YEND_OUT} {
set YSTR_OUT "$YBEG_OUT"
} else {
set YSTR_OUT "$YBEG_OUT-$YEND_OUT"
}

if {$ipccin == "yes" } {
set tfil thetao_[lindex ${table} 0]_${YSTR_IN}_${expname}_${model_version}_${ensnum}.nc
set sfil so_[lindex ${table} 0]_${YSTR_IN}_${expname}_${model_version}_${ensnum}.nc
} else {
set tfil thetao_[lindex ${table} 0]_[pad10000 $ynow]-[pad10000 [expr $ynow+$yinc-1]]_
${runname}_${ensnum}.nc
set sfil so_[lindex ${table} 0]_[pad10000 $ynow]-[pad10000 [expr $ynow+$yinc-1]]_
${runname}_${ensnum}.nc
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${expname}_${ensnum}.nc
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${runname}_${ensnum}.nc
}

puts "Input temperature=$idir/$tfil"
puts "      salinity   =$idir/$sfil"
puts "Output density   =$odir/$ofil"
puts "Temporary file   =$tdir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

if {[file exists $idir/$tfil] == 0} {
puts "$idir/$tfil doesn't exist."
return
} elseif {[file exists $idir/$sfil] == 0} {
puts "$idir/$sfil doesn't exist."
return
}

dmget_tcl "$idir/$tfil $idir/$sfil"

file delete -force $tdir/$ofil $odir/$ofil

set shape [nap_get netcdf -shape $idir/$tfil thetao]
set tshape [nap_get netcdf -shape $idir/$tfil thetao]
set sshape [nap_get netcdf -shape $idir/$sfil so]

if {$tshape != $sshape} {
puts "shape of input files $tshape/$sshape should match."
return
}
```

```
for {set tnow 1} {$tnow <= [lindex $tshape 0]} {incr tnow} {

set tnowm1 [expr $tnow-1]

puts "ynow=$ynow tnow=$tnow"

nap "t0 = [nap_get netcdf $idir/$tfil time "{$tnowm1}"]"

nap "t = f64([nap_get netcdf $idir/$tfil thetao "{$tnowm1},,,"])-273.15"
nap "s = f64([nap_get netcdf $idir/$sfil so "{$tnowm1},,,"])"

if {$tnow == 1} {
nap "time = [nap_get netcdf $idir/$tfil $ldim "{$tnowm1}"]"
nap "depth = f64([nap_get netcdf $idir/$tfil $zdim])"
nap "lat = [nap_get netcdf $idir/$tfil $ydim]"
nap "lon = [nap_get netcdf $idir/$tfil $xdim]"

nap "mis1=isPresent(t)+isPresent(s)"
nap "mis2=(mis1/mis1)"

nap "p1=reshape(0.1,{1 189 192 31})*depth"
nap "p=transpose(p1,{0 3 1 2})*mis2"
#as temperature is already potential, depth should just be p at surface and
#not a function.
nap "p=p*0.0"

nap "t=t*mis2"
nap "s=s*mis2"

};#tnow=1

$p set coo $ldim $zdim $ydim $xdim
$p set dim $ldim $zdim $ydim $xdim

nap "d = f32(unesco(t,s,p)-1000.0)"

$d set missing f32(1e20)
$d netcdf -unlimited -index "$tnowm1,,," $tdir/$ofil "$ovar"
$t0 netcdf -unlimited -index "$tnowm1" $tdir/$ofil "time"

if {$tnow == 1} {
catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}

source ~IPCC/processing/tools/ipcc_dims.tcl

source ~IPCC/processing/tools/ipcc_att.tcl
};#tnow=1

};#tnow

source ~IPCC/processing/tools/ipcc_bnds_ogcm.tcl

puts "Copying $tdir/$ofil $odir/$ofil..."
```

```
file rename -force $tdir/$ofil $odir/$ofil

};#ynow

};#iflg

return
exit
```

14.50 ipcc_rldscs

ipcc_rldscs.tcl is a Tcl-Nap script used to generate parameter *rldscs* defined in Table 7.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl

set proc_name "ipcc_rldscs"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir
puts "Input dir $idir"

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 rgn
set ivar2 rgd
set ivar3 rgc
set ovar rldscs

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}
set ipre3 s${ivar3}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre3}*]
set mfil3 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set m96x48 [glob ~col414/ocean/masks/Mk3_agcm96x48.nc]
set m64x32 [glob ~col414/ocean/masks/Mk3_agcm64x32.nc]
set m192x96 [glob ~col414/ocean/masks/Mk3_agcm.nc]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "            $idir/$mfil3"
puts "Mask files= $m192x96"
puts "            $m96x48"
puts "            $m64x32"
```

```
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefore don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2 $idir/$mfil3 $m192x96 $m96x48 $m64x32"
file delete $tdir/$ofil $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"
nap "v3 = [nap_get netcdf $idir/$mfil3 $ivar3]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]
set v3s [$v3 shape]

set v1x [lindex $v1s 3]
set v1y [lindex $v1s 2]

set v2x [lindex $v2s 3]
```

```
set v2y [lindex $v2s 2]

set v3x [lindex $v3s 3]
set v3y [lindex $v3s 2]

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
}
if {$v2x == 64 && $v2y == 32} {
incr ires
}
if {$v3x == 64 && $v3y == 32} {
incr ires
}

if {$ires == 0} {
#all on full grid.
  nap "v3_full = v3"
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {
#any one on reduced grid.

if {$v3x == 64 && $v3y == 32} {
  nap "v3_full = v3(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar3,$v3x,$v3y'"
} elseif {$v3x == 192 && $v3y == 96} {
  nap "v3_full=ipcc_grid9to1(v3)"
} else {
puts "Don't know that resolution."
return
}

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "v2_full=ipcc_grid9to1(v2)"
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full=ipcc_grid9to1(v1)"
} else {
puts "Don't know that resolution."
return
}

};#ires
```

```
nap "v4 = f32(v1_full+v2_full-v3_full)"

# Make sure it's non-negative
#nap "v4 = v4 >>> 0.0"

$v4 set missing 1e20
$v4 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil


};#dnow

};#area

};#iflg

return
exit
```

14.51 ipcc_rlus

ipcc_rldscs.tcl is a Tcl-Nap script used to generate parameter *rlus* defined in Table 14.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl

set proc_name "ipcc_rlus"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
source ~IPCC/processing/tools/ipcc_mask.tcl

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 rgn
set ivar2 rgd
set ovar rlus

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1] [expr [string length $mfil1]-1]

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefoer don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
```

```
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2"
file delete $tdir/$ofil $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]

set v1x [lindex $v1s 3]
set v1y [lindex $v1s 2]

set v2x [lindex $v2s 3]
set v2y [lindex $v2s 2]

puts "v1s=$v1s"
puts "v2s=$v2s"

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
}
if {$v2x == 64 && $v2y == 32} {
incr ires
}

if {$ires == 0} {
```

```
#all on full grid.
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {
#any one on reduced grid.

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "v2_full=ipcc_grid9to1(v2)"
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full=ipcc_grid9to1(v1)"
} else {
puts "Don't know that resolution."
return
}

};#ires

nap "v3 = f32(v2_full+v1_full)"

$v3 set missing 1e20
$v3 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.52 ipcc_rsdscs

ipcc_rsdscs.tcl is a Tcl-Nap script used to generate parameter *rsdscs* defined in Table 7.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl
source ~col414/tcl_fun/axpc2.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl

set proc_name "ipcc_rsdscs"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
set rdir /cs/datastore/IPCC/IPCC/processing/tools/derived

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

incr pass

set ivar1 sgn
set ivar2 sgc
set ivar3 sgd
set ovar rsdscs

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}
set ipre3 s${ivar3}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre3}*]
set mfil3 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set m96x48 [glob ~col414/ocean/masks/Mk3_agcm96x48.nc]
set m64x32 [glob ~col414/ocean/masks/Mk3_agcm64x32.nc]
set m192x96 [glob ~col414/ocean/masks/Mk3_agcm.nc]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "            $idir/$mfil3"
puts "Mask files= $m192x96"
puts "            $m96x48"
puts "            $m64x32"
puts "Output file=$odir/$ofil"
```

```
set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefoer don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2 $idir/$mfil3 $m192x96 $m96x48 $m64x32"
file delete $tdir/$ofil $odir/$ofil

cd $rdir

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"
nap "v3 = [nap_get netcdf $idir/$mfil3 $ivar3]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]
set v3s [$v3 shape]

set v1x [lindex $v1s 3]
set v1y [lindex $v1s 2]

set v2x [lindex $v2s 3]
```

```
set v2y [lindex $v2s 2]

set v3x [lindex $v3s 3]
set v3y [lindex $v3s 2]

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
}
if {$v2x == 64 && $v2y == 32} {
incr ires
}
if {$v3x == 64 && $v3y == 32} {
incr ires
}

if {$ires == 0} {
#all on full grid.
  nap "v3_full = v3"
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {
#any one on reduced grid.

if {$v3x == 64 && $v3y == 32} {
  nap "v3_full = v3(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar3,$v3x,$v3y'"
} elseif {$v3x == 192 && $v3y == 96} {
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar3,$v3x,$v3y'"
set mfil3_pack sAAA[string range $mfil3 [string first "_" $mfil3]
[expr [string last ".nc" $mfil3]-1]]_pack.nc
set mfil3_red sAAA[string range $mfil3 [string first "_" $mfil3]
[expr [string last ".nc" $mfil3]-1]]_red.nc
file delete -force $idir/${mfil3_pack}
nap "AAA = [nap_get netcdf $idir/$mfil3 $ivar1]"
nap "mi = min(min(min(min(AAA))))"
nap "ma = max(max(max(max(AAA))))"
axpc2 [$mi] [$ma] -32500.0 32500.0 scale offset
nap "scale_factor = ${scale}f"
nap "add_offset = ${offset}f"
nap "valid_min = i16(-32500)"
nap "valid_max = i16(32500)"
nap "BBB = i16((AAA-$add_offset)/scale_factor)"
$BBB set label "$ivar1"
$BBB set unit "dummy"
$BBB netcdf $tdir/${mfil3_pack} AAA
$scale_factor netcdf $tdir/${mfil3_pack} AAA:scale_factor
$add_offset netcdf $tdir/${mfil3_pack} AAA:add_offset
$valid_min netcdf $tdir/${mfil3_pack} AAA:valid_min
$valid_max netcdf $tdir/${mfil3_pack} AAA:valid_max
nap "valid_min = i32([nap_get netcdf $idir/$mfil3 year "0"])"
$valid_min netcdf $tdir/${mfil3_pack} year:valid_min
file delete -force $tdir/${mfil3_red}
cd $tdir
exec [glob ~dix043/bin/ncreduce] -i ${mfil3_pack} -o $tdir/${mfil3_red}
```

```
cd $rdir
nap "v3 = [nap_get netcdf $tdir/${mfil3_red} AAA]"
nap "v3_full = v3(,,@lat,@lon)"
file delete -force $tdir/${mfil3_pack} $tdir/${mfil3_red}
} else {
puts "Don't know that resolution."
return
}

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
set mfil2_pack sAAA[string range $mfil2 [string first "_" $mfil2]
[expr [string last ".nc" $mfil2]-1]]_pack.nc
set mfil2_red sAAA[string range $mfil2 [string first "_" $mfil2]
[expr [string last ".nc" $mfil2]-1]]_red.nc
file delete -force $idir/${mfil2_pack}
nap "AAA = [nap_get netcdf $idir/$mfil2 $ivar2]"
nap "mi = min(min(min(min(AAA))))"
nap "ma = max(max(max(max(AAA))))"
axpc2 [$mi] [$ma] -32500.0 32500.0 scale offset
nap "scale_factor = ${scale}f"
nap "add_offset = ${offset}f"
nap "valid_min = i16(-32500)"
nap "valid_max = i16(32500)"
nap "BBB = i16((AAA-$add_offset)/scale_factor)"
$BBB set label "$ivar2"
$BBB set unit "dummy"
$BBB netcdf $tdir/${mfil2_pack} AAA
$scale_factor netcdf $tdir/${mfil2_pack} AAA:scale_factor
$add_offset netcdf $tdir/${mfil2_pack} AAA:add_offset
$valid_min netcdf $tdir/${mfil2_pack} AAA:valid_min
$valid_max netcdf $tdir/${mfil2_pack} AAA:valid_max
nap "valid_min = i32([nap_get netcdf $idir/$mfil2 year "0"])"
$valid_min netcdf $tdir/${mfil2_pack} year:valid_min
file delete -force $tdir/${mfil2_red}
cd $tdir
exec [glob ~dix043/bin/ncreduce] -i ${mfil2_pack} -o $tdir/${mfil2_red}
cd $rdir
nap "v2 = [nap_get netcdf $tdir/${mfil2_red} AAA]"
#sgd should never be less than 0.
nap "v2_full = v2(,,@lat,@lon)"
file delete -force $tdir/${mfil2_pack} $tdir/${mfil2_red}
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
set mfil1_pack sAAA[string range $mfil1 [string first "_" $mfil1]
```

```
[expr [string last ".nc" $mfil1]-1]]_pack.nc
set mfil1_red sAAA[string range $mfil1 [string first "_" $mfil1]
[expr [string last ".nc" $mfil1]-1]]_red.nc
file delete -force $idir/${mfil1_pack}
nap "AAA = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "mi = min(min(min(min(AAA))))"
nap "ma = max(max(max(max(AAA))))"
axpc2 [$mi] [$ma] -32500.0 32500.0 scale offset
nap "scale_factor = ${scale}f"
nap "add_offset = ${offset}f"
nap "valid_min = i16(-32500)"
nap "valid_max = i16(32500)"
nap "BBB = i16((AAA-$add_offset)/scale_factor)"
$BBB set label "$ivar1"
$BBB set unit "dummy"
$BBB netcdf $tdir/${mfil1_pack} AAA
$scale_factor netcdf $tdir/${mfil1_pack} AAA:scale_factor
$add_offset netcdf $tdir/${mfil1_pack} AAA:add_offset
$valid_min netcdf $tdir/${mfil1_pack} AAA:valid_min
$valid_max netcdf $tdir/${mfil1_pack} AAA:valid_max
nap "valid_min = i32([nap_get netcdf $idir/$mfil1 year "0"])"
$valid_min netcdf $tdir/${mfil1_pack} year:valid_min
file delete -force $tdir/${mfil1_red}
cd $tdir
exec [glob ~dix043/bin/ncreduce] -i ${mfil1_pack} -o $tdir/${mfil1_red}
cd $rdir
nap "v1 = [nap_get netcdf $tdir/${mfil1_red} AAA]"
nap "v1_full = v1(,,@lat,@lon)"
file delete -force $tdir/${mfil1_pack} $tdir/${mfil1_red}
} else {
puts "Don't know that resolution."
return
}

};#ires

puts "v1=[$v1_full a]"
puts "v2=[$v2_full a]"
puts "v3=[$v3_full a]"

nap "v4 = f32(v3_full*v2_full/v1_full)"

$v4 set missing 1e20
$v4 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg
```

```
return
exit
```

14.53 ipcc_rsuscs

ipcc_rsuscs.tcl is a Tcl-Nap script used to generate parameter *rsuscs* defined in Table 7.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl

set proc_name "ipcc_rsuscs"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir
puts "Input dir $idir"

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 sgn
set ivar2 sgc
set ivar3 sgd
set ovar rsuscs

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}
set ipre3 s${ivar3}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre3}*]
set mfil3 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set m96x48 [glob ~col414/ocean/masks/Mk3_agcm96x48.nc]
set m64x32 [glob ~col414/ocean/masks/Mk3_agcm64x32.nc]
set m192x96 [glob ~col414/ocean/masks/Mk3_agcm.nc]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1] [expr [string length $mfil1]-1]

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "            $idir/$mfil3"
puts "Mask files= $m192x96"
puts "            $m96x48"
puts "            $m64x32"
puts "Output file=$odir/$ofil"
```

```
set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefore don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2 $idir/$mfil3 $m192x96 $m96x48 $m64x32"
file delete $tdir/$ofil $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"
nap "v3 = [nap_get netcdf $idir/$mfil3 $ivar3]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]
set v3s [$v3 shape]

set v1x [lindex $v1s 3]
set v1y [lindex $v1s 2]

set v2x [lindex $v2s 3]
set v2y [lindex $v2s 2]
```

```
set v3x [lindex $v3s 3]
set v3y [lindex $v3s 2]

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
}
if {$v2x == 64 && $v2y == 32} {
incr ires
}
if {$v3x == 64 && $v3y == 32} {
incr ires
}

if {$ires == 0} {
#all on full grid.
  nap "v3_full = v3"
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {
#any one on reduced grid.

if {$v3x == 64 && $v3y == 32} {
  nap "v3_full = v3(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar3,$v3x,$v3y'"
} elseif {$v3x == 192 && $v3y == 96} {
  nap "v3_full=ipcc_grid9to1(v3)"
} else {
puts "Don't know that resolution."
return
}

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "v2_full=ipcc_grid9to1(v2)"
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full=ipcc_grid9to1(v1)"
} else {
puts "Don't know that resolution."
return
}

};#ires
```

```
nap "v4 = f32(v3_full*v2_full/v1_full-v2_full)"

# Make sure it's non-negative
nap "v4 = v4 >>> 0.0"

$v4 set missing 1e20
$v4 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.54 ipcc_rsus

ipcc_rsus.tcl is a Tcl-Nap script used to generate parameter *rsus* defined in Table 5.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(WORKDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl
source ~col414/tcl_fun/axpc2.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl

set proc_name "ipcc_rsus"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
source ~IPCC/processing/tools/ipcc_mask.tcl

set rdir /cs/datastore/IPCC/IPCC/processing/tools/derived

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 sgn
set ivar2 sgd
set ovar rsus

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
```

```
  continue
  } else {
  #bit like noclobber - it exists already therefoer don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2"
file delete $tdir/$ofil $odir/$ofil

cd $rdir

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]

set v1x [lindex $v1s 3]
set v1y [lindex $v1s 2]

set v2x [lindex $v2s 3]
set v2y [lindex $v2s 2]

puts "v1s=$v1s"
puts "v2s=$v2s"

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
```

```
}
if {$v2x == 64 && $v2y == 32} {
incr ires
}

if {$ires == 0} {
#all on full grid.
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
set mfil2_pack sAAA[string range $mfil2 [string first "_" $mfil2]
[expr [string last ".nc" $mfil2]-1]]_pack.nc
set mfil2_red sAAA[string range $mfil2 [string first "_" $mfil2]
[expr [string last ".nc" $mfil2]-1]]_red.nc
file delete -force $idir/${mfil2_pack}
nap "AAA = [nap_get netcdf $idir/$mfil2 $ivar2]"
nap "mi = min(min(min(min(AAA))))"
nap "ma = max(max(max(max(AAA))))"
axpc2 [$mi] [$ma] -32500.0 32500.0 scale offset
nap "scale_factor = ${scale}f"
nap "add_offset = ${offset}f"
nap "valid_min = i16(-32500)"
nap "valid_max = i16(32500)"
nap "BBB = i16((AAA-$add_offset)/scale_factor)"
$BBB set label "$ivar2"
$BBB set unit "dummy"
$BBB netcdf $tdir/${mfil2_pack} AAA
$scale_factor netcdf $tdir/${mfil2_pack} AAA:scale_factor
$add_offset netcdf $tdir/${mfil2_pack} AAA:add_offset
$valid_min netcdf $tdir/${mfil2_pack} AAA:valid_min
$valid_max netcdf $tdir/${mfil2_pack} AAA:valid_max
nap "valid_min = i32([nap_get netcdf $idir/$mfil2 year "0"])"
$valid_min netcdf $tdir/${mfil2_pack} year:valid_min
file delete -force $tdir/${mfil2_red}
cd $tdir
exec [glob ~dix043/bin/ncreduce] -i ${mfil2_pack} -o $tdir/${mfil2_red}
cd $rdir
nap "v2 = [nap_get netcdf $tdir/${mfil2_red} AAA]"
#sgd should never be less than 0.
nap "v2_full = v2(,,@lat,@lon)"
file delete -force $tdir/${mfil2_pack} $tdir/${mfil2_red}
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
```

```
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ’,’ // ’$ivar1,$v1x,$v1y’"
set mfil1_pack sAAA[string range $mfil1 [string first "_" $mfil1]
[expr [string last ".nc" $mfil1]-1]]_pack.nc
set mfil1_red sAAA[string range $mfil1 [string first "_" $mfil1]
[expr [string last ".nc" $mfil1]-1]]_red.nc
file delete -force $idir/${mfil1_pack}
nap "AAA = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "mi = min(min(min(min(AAA))))"
nap "ma = max(max(max(max(AAA))))"
axpc2 [$mi] [$ma] -32500.0 32500.0 scale offset
nap "scale_factor = ${scale}f"
nap "add_offset = ${offset}f"
nap "valid_min = i16(-32500)"
nap "valid_max = i16(32500)"
nap "BBB = i16((AAA-$add_offset)/scale_factor)"
$BBB set label "$ivar1"
$BBB set unit "dummy"
$BBB netcdf $tdir/${mfil1_pack} AAA
$scale_factor netcdf $tdir/${mfil1_pack} AAA:scale_factor
$add_offset netcdf $tdir/${mfil1_pack} AAA:add_offset
$valid_min netcdf $tdir/${mfil1_pack} AAA:valid_min
$valid_max netcdf $tdir/${mfil1_pack} AAA:valid_max
nap "valid_min = i32([nap_get netcdf $idir/$mfil1 year "0"])"
$valid_min netcdf $tdir/${mfil1_pack} year:valid_min
file delete -force $tdir/${mfil1_red}
cd $tdir
exec [glob ~dix043/bin/ncreduce] -i ${mfil1_pack} -o $tdir/${mfil1_red}
cd $rdir
nap "v1 = [nap_get netcdf $tdir/${mfil1_red} AAA]"
nap "v1_full = v1(,,@lat,@lon)"
file delete -force $tdir/${mfil1_pack} $tdir/${mfil1_red}
} else {
puts "Don’t know that resolution."
return
}

};#ires

#should never be less than 0.
nap "v3 = f32(v2_full-v1_full) >>> 0"

$v3 set missing 1e20
$v3 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg
```

```
return
exit
```

14.55 ipcc_rtmt

ipcc_rtmt.tcl is a Tcl-Nap script used to generate parameter *rtmt* defined in Table 7.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl
source ~IPCC/processing/tools/derived/ipcc_grid9to1.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl

set proc_name "ipcc_rtmt"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0
set setdim "no"

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 rtu
set ivar2 sot
set ivar3 sit
set ovar rtmt

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

nap "land_sea_horiz_interp = '$land_sea_interp'"

set ipre1 s${ivar1}_${run}
set ipre2 s${ivar2}_${run}

set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set m96x48 [glob ~col414/ocean/masks/Mk3_agcm96x48.nc]
set m64x32 [glob ~col414/ocean/masks/Mk3_agcm64x32.nc]
set m192x96 [glob ~col414/ocean/masks/Mk3_agcm.nc]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

set mfil3 [glob ~IPCC/data/input/sint_t63.nc]

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "            $mfil3"
puts "Mask files= $m192x96"
puts "            $m96x48"
puts "            $m64x32"
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
```

```
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefore don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2 $mfil3 $m192x96 $m96x48 $m64x32"
file delete $tdir/$ofil $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "v2 = [nap_get netcdf $idir/$mfil2 $ivar2]"
nap "v3 = [nap_get netcdf $mfil3 $ivar3]"

if {$pass >  0 && $setdim == "no"} {
set setdim "yes"
nap "l1 = [nap_get netcdf $m192x96 land "0,0,,"]"
nap "l2 = [nap_get netcdf $m96x48 land "0,0,,"]"
nap "l3 = [nap_get netcdf $m64x32 land "0,0,,"]"

nap "l1p = isPresent(l1)"
nap "l2p = isPresent(l2)"
nap "l3p = isPresent(l3)"

nap "o1 = isMissing(l1)/isMissing(l1)"
nap "o2 = isMissing(l2)/isMissing(l2)"
nap "o3 = isMissing(l3)/isMissing(l3)"

nap "o1p = isPresent(o1)"
nap "o2p = isPresent(o2)"
nap "o3p = isPresent(o3)"

nap "lon=cv(l1,1)"
nap "lat=cv(l1,0)"
}

set v1s [$v1 shape]
set v2s [$v2 shape]
set v3s [$v3 shape]

set v1x [lindex $v1s end]
set v1y [lindex $v1s end-1]

set v2x [lindex $v2s end]
set v2y [lindex $v2s end-1]

set v3x [lindex $v3s end]
```

```
set v3y [lindex $v3s end-1]

set ires 0

if {$v1x == 64 && $v1y == 32} {
incr ires
}
if {$v2x == 64 && $v2y == 32} {
incr ires
}
if {$v3x == 64 && $v3y == 32} {
incr ires
}

if {$ires == 0} {
#all on full grid.
  nap "v3_full = v3"
  nap "v2_full = v2"
  nap "v1_full = v1"
} elseif {$ires > 0} {
#any one on reduced grid.

if {$v2x == 64 && $v2y == 32} {
  nap "v2_full = v2(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar2,$v2x,$v2y'"
} elseif {$v2x == 192 && $v2y == 96} {
  nap "v2_full=ipcc_grid9to1(v2)"
} else {
puts "Don't know that resolution."
return
}

if {$v1x == 64 && $v1y == 32} {
  nap "v1_full = v1(,,@lat,@lon)"
  nap "land_sea_horiz_interp = land_sea_horiz_interp // ',' // '$ivar1,$v1x,$v1y'"
} elseif {$v1x == 192 && $v1y == 96} {
  nap "v1_full=ipcc_grid9to1(v1)"
} else {
puts "Don't know that resolution."
return
}

};#ires

set newshape [lreplace $v1s 2 3 [$lat nels] [$lon nels]]

nap "v3_full = reshape(v3,{$newshape})"

nap "latitude=lat"
nap "longitude=lon"
nap "year=cv(v1,0)"
nap "month=cv(v1,1)"

$v3_full set dim year month latitude longitude
$v3_full set coo year month latitude longitude
```

```
nap "v4 = f32(v3_full-v2_full-v1_full)"

$v4 set missing 1e20
$v4 netcdf $tdir/$ofil $ovar
if {[info vars land_sea_horiz_interp] != ""} {
$land_sea_horiz_interp netcdf $tdir/$ofil $ovar:land_sea_horiz_interp
}

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.56 ipcc_snc

ipcc_snc.tcl is a Tcl-Nap script used to generate parameter *snc* defined in Table 6.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

#I believe snd and therefore snc will be on 192x96 grid for all IPCC experiments.

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl

set proc_name "ipcc_snc"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {
```

```
source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

puts "$odir"
puts "bbb"

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 snd
set ivar2 ico
set ovar snc

set ipre1 s${ivar1}_${run}
set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ipre2 s${ivar2}_${run}
set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefore don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}
```

```
dmget_tcl "$idir/$mfil1 $idir/$mfil2"
file delete $tdir/$ofil $odir/$ofil

nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "va = [nap_get netcdf $idir/$mfil2 $ivar2]"

nap "v2 = f32(v1>0.0038565)*100";#as percentage
nap "vb = f32(va<=0.0038565)*1";#sea-ice

nap "v2 = v2*vb";#wipe out sea-ice areas

$v2 set missing 1e20

$v2 netcdf $tdir/$ofil $ovar

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.57 ipcc_snw

ipcc_snw.tcl is a Tcl-Nap script used to generate parameter *snw* defined in Table 6.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier June 2004

#I believe snd and therefore snw will be on 192x96 grid for all IPCC experiments.

source ~/.tclshrc

set tdir $::env(TMPDIR)

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl

set proc_name "ipcc_snw"
set time_freq "monthly"

puts "Entering procedure $proc_name"

catch {unset land_sea_horiz_interp}
set pass 0

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {
```

```
source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar1 snd
set ivar2 ico
set ovar snw

set ipre1 s${ivar1}_${run}
set aaa [glob $idir/${ipre1}*]
set mfil1 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ipre2 s${ivar2}_${run}
set aaa [glob $idir/${ipre2}*]
set mfil2 [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ofil "$ovar[string range $mfil1 [string first "_" $mfil1]
[expr [string length $mfil1]-1]]"

puts "Input files=$idir/$mfil1"
puts "            $idir/$mfil2"
puts "Output file=$odir/$ofil"

set check "[glob -nocomplain $odir/${ovar}_*]"
if {[llength $check] == 2} {
puts "Two files in output directory resembling output files: $check"
continue
} elseif {[llength $check] == 1} {
  if {$check != "$odir/$ofil"} {
  puts "Actual output file and intended output file don't match: $check $odir/$ofil"
  continue
  } else {
  #bit like noclobber - it exists already therefore don't make it
  puts "Actual output file and intended output file match: $check $odir/$ofil"
  continue
  }
} elseif {[llength $check] == 0} {
puts "No output file, therefore creating."
}

dmget_tcl "$idir/$mfil1 $idir/$mfil2"
file delete $tdir/$ofil $odir/$ofil
```

```
nap "v1 = [nap_get netcdf $idir/$mfil1 $ivar1]"
nap "va = [nap_get netcdf $idir/$mfil2 $ivar2]"

nap "v2 = f32(v1>0.0038565)*v1";#as kg m-2
nap "vb = f32(va<=0.0038565)*1";#sea-ice

nap "v2 = v2*vb";#wipe out sea-ice areas

$v2 set missing 1e20

$v2 netcdf $tdir/$ofil $ovar

puts "Copying $tdir/$ofil to $odir/$ofil..."
file rename -force $tdir/$ofil $odir/$ofil

};#dnow

};#area

};#iflg

return
exit
```

14.58 ipcc_stfmmc_ferret

ipcc_stfmmc_ferret.tcl is a Tcl-Nap script used to generate parameter *stfmmc* defined in Table 11 using FERRET.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier July 2004

source ~/.tclshrc

set user $::env(USER)

set time_freq "monthly"
set proc_name "ipcc_stfmmc_ferret"

set ferret_exec /tools/ferret_581_ia32/bin/ferret

puts "Entering procedure $proc_name"

source [glob ~col414/tcl_fun/dmget_tcl.tcl]
source [glob ~col414/tcl_fun/pad10000.tcl]
source [glob ~IPCC/processing/tools/ipcc_cmv.tcl]
source [glob ~IPCC/processing/tools/hostname.tcl]

set hostname [hostname]

set type "eddy";#eddy
set type "";#non-eddy

set ipccout "no"
set ipccout "yes"
set ipccin "no"
set ipccin "yes"

set clobber "yes"
set clobber "no"

if {$type == "eddy"} {
set ivar vedd
} else {
set ivar vvel
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set table_backup $table

if {$type == "eddy"} {
set ivar stfmme
} else {
set ivar stfmmc
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation
```

```
set predir /cs/datastore/csdar/$user/ocean/ipcc/data
set predir /cs/datastore/IPCC/IPCC/data

#doesn't exist for 9/10
set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source [glob ~IPCC/processing/tools/ipcc_intro.tcl]

if {$type == "eddy" && $iflg == 16 && $ybeg < 501} {
set ybeg 501
}

set odir $predir/$ovar
set odir [glob ~IPCC/data/$model_version/$ovar]

for {set ynow $ybeg} {$ynow <= $yend} {incr ynow $yinc} {

if {$ynow > 250 && $runname == "m20"} {
set expname "SRESA2"
} elseif {$ynow > 260 && $runname == "n20"} {
set expname "SRESA2"
} elseif {$ynow > 300 && $runname == "B35"} {
set expname "SRESA2"
}

set tdir /work/$user/$expname/[lindex $table_backup 0]/$ensnum
file mkdir $tdir
```

```
set idir $predir/$model_version/$expname/[lindex $table_backup 0]/$ensnum

if {$ipccin == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_IN [pad10000 $cbeg]
set YEND_IN [pad10000 $cend]
} else {
set YBEG_IN [pad10000 $ynow]
set YEND_IN [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_IN == $YEND_IN} {
set YSTR_IN "$YBEG_IN"
} else {
set YSTR_IN "$YBEG_IN-$YEND_IN"
}

if {$ipccout == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_OUT [pad10000 $cbeg]
set YEND_OUT [pad10000 $cend]
} else {
set YBEG_OUT [pad10000 $ynow]
set YEND_OUT [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_OUT == $YEND_OUT} {
set YSTR_OUT "$YBEG_OUT"
} else {
set YSTR_OUT "$YBEG_OUT-$YEND_OUT"
}

if {$ipccin == "yes" } {
  if {$type == "eddy"} {
  set vfil ve_[lindex ${table_backup} 0]_${YSTR_IN}_${expname}_${model_version}_
${ensnum}.nc
  } else {
  set vfil vo_[lindex ${table_backup} 0]_${YSTR_IN}_${expname}_${model_version}_
${ensnum}.nc
  }
} else {
  if {$type == "eddy"} {
  set vfil ve_[lindex ${table_backup} 0]_${YSTR_IN}_${runname}_${model_version}_
${ensnum}.nc
  } else {
  set vfil vo_[lindex ${table_backup} 0]_${YSTR_IN}_${runname}_${model_version}_
${ensnum}.nc
  }
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${expname}_${model_version}_${ensnum}.nc
} else {
```

```
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${runname}_${model_version}_${ensnum}.nc
}

puts "Input V    =$idir/$vfil"
puts "Output MOC =$odir/$ofil"
if {$type == "eddy"} {
puts "Temporary MOC=$tdir/moce.nc"
} else {
puts "Temporary MOC=$tdir/moc.nc"
}

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

if {[file exists $idir/$vfil] == 0} {
puts "$idir/$vfil doesn't exist."
return
}

puts "Getting files..."
dmget_tcl $idir/$vfil
puts "Done."

if {$type == "eddy"} {
file delete -force $tdir/moce.nc $odir/$ofil
catch {exec $ferret_exec -script moc.jnl $idir/$vfil $tdir ve}

puts "Renaming $tdir/moce.nc to $odir/$ofil ..."
file rename -force $tdir/moce.nc $odir/$ofil
} else {

file delete -force $tdir/moc.nc $odir/$ofil

catch {exec $ferret_exec -script moc.jnl $idir/$vfil $tdir vo}

puts "Renaming $tdir/moc.nc to $odir/$ofil ..."

file rename -force $tdir/moc.nc $odir/$ofil
}

};#ynow

};#iflg

return
exit
```

14.59 ipcc_stfmmc

ipcc_stfmmc.tcl is a Tcl-Nap script used to generate parameter *stfmmc* defined in Table 11 computed in a previous step using ipcc$_stfmmc_ferretdescribedinSection$ 14.57.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier August 2004

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_stfmmc"
set time_freq "annual"
set time_freq "monthly"

puts "Entering procedure $proc_name"

set type "eddy";#eddy
set type "";#non-eddy

set ipccin "no"
set ipccin "yes"
set ipccout "no"
set ipccout "yes"

set clobber "yes"
set clobber "no"

source [glob ~col414/tcl_fun/pad10000.tcl]
source [glob ~IPCC/processing/tools/ipcc_cmv.tcl]
source [glob ~IPCC/processing/tools/xyzt_edges.tcl]

source ~/.tclshrc

set tdir /work/IPCC

set rank 4

set tregion {ATLANTIC INDIAN PACIFIC GLOBAL}

if {$type == "eddy"} {
set ivar vedd
} else {
set ivar vvel
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set table_backup $table

if {$type == "eddy"} {
set ivar stfmme
} else {
set ivar stfmmc
```

```
}

ipcc_cmv $ivar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set predir /cs/datastore/IPCC/IPCC/data/${model_version}

set idir $predir/$ovar

for {set ynow $ybeg} {$ynow <= $yend} {incr ynow $yinc} {

set nbeg $ybeg;set nend $yend

if {$ynow > 250 && $runname == "m20"} {
set expname "SRESA2"
} elseif {$ynow > 260 && $runname == "n20"} {
set expname "SRESA2"
} elseif {$ynow > 300 && $runname == "B35"} {
set expname "SRESA2"
}

set odir $predir/$expname/[lindex $table 0]/$ensnum;file mkdir $odir

if {$ipccin == "yes" } {
```

```
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_IN [pad10000 $cbeg]
set YEND_IN [pad10000 $cend]
} else {
set YBEG_IN [pad10000 $ynow]
set YEND_IN [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_IN == $YEND_IN} {
set YSTR_IN "$YBEG_IN"
} else {
set YSTR_IN "$YBEG_IN-$YEND_IN"
}

if {$ipccout == "yes" } {
set cbeg [expr ${ybas_year}-${fil1_year}+$ynow]
set cend [expr $cbeg+$yinc-1]
set YBEG_OUT [pad10000 $cbeg]
set YEND_OUT [pad10000 $cend]
} else {
set YBEG_OUT [pad10000 $ynow]
set YEND_OUT [pad10000 [expr $ynow+$yinc-1]]
}

if {$YBEG_OUT == $YEND_OUT} {
set YSTR_OUT "$YBEG_OUT"
} else {
set YSTR_OUT "$YBEG_OUT-$YEND_OUT"
}

if {$ipccin == "yes" } {
set ifil ${ovar}_[lindex ${table} 0]_${YSTR_IN}_${expname}_${model_version}_${ensnum}.nc
} else {
set ifil ${ovar}_[lindex ${table} 0]_${YSTR_IN}_${runname}_${model_version}_${ensnum}.nc
}

if {$ipccout == "yes" } {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${expname}_${model_version}_${ensnum}.nc
} else {
set ofil ${ovar}_[lindex ${table} 0]_${YSTR_OUT}_${runname}_${model_version}_${ensnum}.nc
}

puts "Input MOC  =$idir/$ifil"
puts "Output MOC =$odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

if {[file exists $idir/$ifil] == 0} {
puts "$idir/$ifil doesn't exist."
continue
}

file delete -force $tdir/$ofil $odir/$ofil
```

```
set shape [nap_get netcdf -shape $idir/$ifil [string toupper $ovar]_[lindex $tregion 0]]

for {set tnow 1} {$tnow <= [lindex $shape 0]} {incr tnow} {

set tnowm1 [expr $tnow-1]

puts "ynow=$ynow tnow=$tnow"

nap "t0 = [nap_get netcdf $idir/$ifil TIME "{$tnowm1}"]"

if {$tnow == 1} {
nap "stfmmc = [nap_get netcdf $idir/$ifil [string toupper $ovar]_[lindex $tregion 0] "{$tnowm
nap "region=f64({0 1 2 3})"
$region set label global_ocean


set oshape "[lindex [$stfmmc shape] 0] [$region shape] [lrange [$stfmmc shape] end-1 end]"
nap "time = [nap_get netcdf $idir/$ifil [string toupper $ldim] "{$tnowm1}"]"
nap "depth = f64([nap_get netcdf $idir/$ifil [string toupper $zdim]])"
nap "lat = [nap_get netcdf $idir/$ifil [string toupper $ydim]]"
};#tnow=1

nap "stfmmc = reshape(f32(_),{$oshape})"

$stfmmc set dim $ldim region $zdim $ydim
$stfmmc set coo $ldim region $zdim $ydim
$stfmmc set missing f32(1e20)

if {$tnow == 1} {
$stfmmc netcdf -unlimited -index "$tnowm1,,," $tdir/$ofil "$ovar"
};#tnow

$t0 netcdf -unlimited -index "$tnowm1" $tdir/$ofil "time"

set rnt 0
foreach r $tregion {
nap "stfmmc = [nap_get netcdf $idir/$ifil [string toupper $ovar]_$r "{$tnowm1},,"]"

$stfmmc netcdf -unlimited -index "$tnowm1,$rnt,," $tdir/$ofil "$ovar"
incr rnt
};#r

if {$tnow == 1} {
catch {[nap_get netcdf $tdir/$ofil $ovar:_FillValue] netcdf \
$tdir/$ofil $ovar:missing_value}
source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
};#tnow=1

}

source ~IPCC/processing/tools/ipcc_bnds_ogcm.tcl

file rename -force $tdir/$ofil $odir/$ofil
```

```
};#ynow

};#iflg

return
exit
```

14.60 ipcc_zg

ipcc_zg.tcl is a Tcl-Nap script used to generate parameter *zg* defined in Table 9.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September 2004

source ~/.tclshrc

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/fill_holesA.tcl
source ~col414/tcl_fun/pad10000.tcl

set proc_name "ipcc_zg"
set time_freq "monthly"

puts "Entering procedure $proc_name"

set levs {1000 925 850 700 600 500 400 300 250 200 150 100 70 50 30 20 10}

set znt 0
foreach z $levs {
if {$znt == [expr [llength $levs]-1]} {
append zlevs "$z"
} else {
append zlevs "$z,"
}
incr znt
};#z

catch {unset land_sea_horiz_interp}
set pass 0

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
```

```
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
source ~IPCC/processing/tools/ipcc_derive.tcl

for {set area 0} {$area < [llength $dbeg]} {incr area} {

for {set dnow [lindex $dbeg $area] } {$dnow <= [lindex $dend $area]} {incr dnow} {

set idir [lindex $pdir $area]/netcdf.$dnow.dir
set odir [lindex $pdir $area]/netcdf.$dnow.dir

if {[file owned $odir] != 1} {
continue
} else {
incr pass
}

set ivar t01
set ovar ""

set ipre s${ivar}_${run}

set aaa [glob $idir/${ipre}*]
set mfil [string range $aaa [expr [string last "/" $aaa]+1] [string length $aaa]]

set ofil "$ovar[string range $mfil [string first "_" $mfil]
[expr [string length $mfil]-1]]"

puts "Output file=$odir/$ofil"

set miss 0
foreach z $levs {
set tfil "z[pad10000 $z]$ofil"
if {[file exists $odir/$tfil] != 1} {
incr miss
}
};#z

if {$miss == 0} {
continue
}

set ifils "s{t0?,t1?,q0?,q1?,psl}[string range $mfil [string first "_" $mfil]
[expr [string length $mfil]-1]]"

puts $idir/$ifils
```

```
dmget_tcl "$idir/$ifils [glob ~dix043/bin/heights_std]"

puts "Input files=$idir/$ifils"

cd $idir

if {$runname == "B35" || $runname == "C35" || $runname == "D35" || $runname == "E35"
|| $runname == "F35" || $runname == "G35" || $runname == "M35"
|| ($runname == "N35" && $dnow>12) || $runname == "s20" || $runname == "st1"
|| $runname == "a1b" || $runname == "cm3" || $runname == "ct3" || $runname == "qv3"
|| $runname == "qv4" || $runname == "n20"} {
exec /cs/datastore/csdar/dix043/bin/heights_std -z $zlevs -i $ofil -o $ofil
} else {
exec /cs/datastore/csdar/dix043/bin/heights_std --reduced -z $zlevs -i $ofil -o $ofil

};#runname

};#dnow

};#area

};#iflg

return
exit
```

14.60.1 Extreme_Indices

14.61 ipcc_ei_cdd

ipcc_cdd.tcl is a Tcl-Nap script used to generate parameter *cdd* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   3 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_cdd"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas 2;#MAM
set iseas 1;#DJF
```

```
set iseas 3;#JJA

set iseas {0}
set iseas 4;#SON
set iseas 0;#annual
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set scale 1
} elseif {$iflg == 101 || $iflg == 103} {
set scale 1
} elseif {$iflg == 102} {
set scale 1
} else {
set scale 86400
}

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
```

```
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar prate
} elseif {$iflg == 101} {
set ivar lsp
} elseif {$iflg == 102} {
set ivar rain
} elseif {$iflg == 103} {
set ivar precip
} else {
set ivar pr
}
set ovar cdd

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
```

```
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];set yre [expr [lindex $years_tcl $ycnt]+
[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

puts $ifils

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
```

```
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 200*$normt]
set normy [expr 200*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}
```

```
set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

puts "total_days across all times=$total_days"

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
```

```
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons will be equal to
#number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,lat_ind,lon_ind"]*$sc

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*$s

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
```

```
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"

nap "below = i32(va_year<1.0)"

set ivs_year [[nap shape(va_year)]]

nap "zeros = reshape(i32(0),{1 [lrange $ivs_year 1 2]})"

nap "b0 = transpose(transpose(zeros,{0 2 1})//transpose(below,{0 2 1}),{0 2 1})"
nap "b1 = transpose(transpose(below,{0 2 1})//transpose(zeros,{0 2 1}),{0 2 1})"

nap "i = i32(0 .. [lindex $ivs_year 0])"
nap "i_pad = transpose(reshape(i,{[lrange $ivs_year 1 2] [expr [lindex $ivs_year 0]+1]}),
{2 0 1})"

nap "va_tmp = reshape(f32(0),{[lrange $ivs_year 1 2]})"
for {set x 0} {$x < [lindex $ivs_year 2]} {incr x} {
for {set y 0} {$y < [lindex $ivs_year 1]} {incr y} {
nap "i_ext = i_pad(,$y,$x)"
nap "b0_ext = b0(,$y,$x)"
nap "b1_ext = b1(,$y,$x)"
nap "list = ((b0_ext && ! b1_ext) # i_ext) - ((b1_ext && ! b0_ext) # i_ext)"
nap "cdd_ext = f32(max(list)) >>> 0"
$va_tmp set value cdd_ext "$y,$x"
};#y
};#x
nap "va_tmp = va_tmp >>> 0"

$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
```

```
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

14.62 ipcc_ei_etr

ipcc_etr.tcl is a Tcl-Nap script used to generate parameter *etr* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   2 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_etr"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}
```

```
set iseas 1;#DJF
set iseas 2;#MAM
set iseas 4;#SON
set iseas 3;#JJA
set iseas 0;#annual
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir
```

```
if {$iflg == 100} {
set ivar "tmin/tmax"
set ivar1 tmin
set ivar2 tmax
} elseif {$iflg == 101} {
set ivar "mnt2/mxt2"
set ivar1 mnt2
set ivar2 mxt2
} elseif {$iflg == 102} {
set ivar "tmin/tmax"
set ivar1 tmin
set ivar2 tmax
} elseif {$iflg == 103} {
set ivar "tmin/tmax"
set ivar1 mn2t
set ivar2 mx2t
} else {
set ivar "tasmin/tasmax"
set ivar1 tasmin
set ivar2 tasmax
}
set ovar etr

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
```

```
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils1}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils1 $idir/${ivar1}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils1 $idir/${ivar1}_${itab}_${yrb}-${yre}_
  ${expname}_${model_version}_${ensnum}.nc
  }
}
catch {unset ifils2}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils2 $idir/${ivar2}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils2 $idir/${ivar2}_${itab}_${yrb}-${yre}_
  ${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils1 $ifils2"
```

```
set ivs [nap_get netcdf -shape [lindex $ifils1 0] ${ivar1}]
nap "lat = [nap_get netcdf [lindex $ifils1 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils1 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 150*$normt]
set normy [expr 150*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
```

```
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] ..
[lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv1 = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "iv2 = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
```

```
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv1 set value [nap_get netcdf [lindex $ifils1 $dnt] ${ivar1} "day_ind_djf,lat_ind,lon_ind"]
"tim_ind,,"
$iv2 set value [nap_get netcdf [lindex $ifils2 $dnt] ${ivar2} "day_ind_djf,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils1 $dnt] time "day_ind_djf"]
"tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv1 set value [nap_get netcdf [lindex $ifils1 $dnt] ${ivar1} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"
$iv2 set value [nap_get netcdf [lindex $ifils2 $dnt] ${ivar2} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]
```

```
  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils1 $dnt] time "day_ind_long"]
"tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv1 set value [nap_get netcdf [lindex $ifils1 $dnt] ${ivar1} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"
$iv2 set value [nap_get netcdf [lindex $ifils2 $dnt] ${ivar2} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils1 $dnt] time "day_ind_long"]
"tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va1_year = iv1(day_ind_ynow,,)"
nap "va2_year = iv2(day_ind_ynow,,)"
nap "va_tmp = max(va2_year)-min(va1_year)"

$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask
```

```
};#ynow

};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea


return
exit
```

14.63 ipcc_ei_fd

ipcc_fd.tcl is a Tcl-Nap script used to generate parameter *fd* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   2 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_fd"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {1 2 3 4}
```

```
set iseas 4;#SON
set iseas 2;#MAM
set iseas 1;#DJF
set iseas 3;#JJA
set iseas 0;#annual
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set offset 273.15
} elseif {$iflg == 101 || $iflg == 103} {
set offset 273.15
} elseif {$iflg == 102} {
set offset 0.0
} else {
set offset 273.15
}

source ~IPCC/processing/tools/ipcc_intro.tcl
set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
```

```
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar tmin
} elseif {$iflg == 101} {
set ivar mnt2
} elseif {$iflg == 102} {
set ivar tmin
} elseif {$iflg == 103} {
set ivar mn2t
} else {
set ivar tasmin
}
set ovar fd

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
```

```
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_
  ${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}
```

```
set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 900*$normt]
set normy [expr 900*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
```

```
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow
```

```
if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask
```

```
};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"
nap "va_tmp = sum(va_year<0.0)"
$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg
```

```
};#isea

return
exit
```

14.64 ipcc_ei_gsl

ipcc_gsl.tcl is a Tcl-Nap script used to generate parameter *gsl* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November  15 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_gsl"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set isea 0;#annual
```

```
set sea_lab "_ANN"

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set offset 273.15
} elseif {$iflg == 101} {
set offset 273.15
} elseif {$iflg == 103} {
set offset 273.15
} else {
set offset 273.15
}

source ~IPCC/processing/tools/ipcc_intro.tcl
set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar air
} elseif {$iflg == 101 || $iflg == 103} {
set ivar t2
} else {
set ivar tas
}
set ovar gsl

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
```

```
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];set
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/ocean/ipcc/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/ocean/ipcc/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_
${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt
```

```
catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

set ntims [expr $ntims/4]

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 200*$normt]
set normy [expr 200*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]
```

```
if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] ..
[lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
```

```
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"]
"tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]
```

```
  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"
nap "va_tmp = sum(va_year<0.0)"
$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow


};#ytask
};#xtask
```

```
$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

return
exit
```

14.65 ipcc_ei_hwdi

ipcc_hwdi.tcl is a Tcl-Nap script used to generate parameter *hwdi* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   2 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_hwdi"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}
```

```
set iseas 2;#MAM
set iseas 3;#JJA
set iseas 1;#DJF
set iseas {1 2 3 4}
set iseas 4;#SON
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set offset 273.15
} elseif {$iflg == 101} {
set offset 273.15
} elseif {$iflg == 102} {
set offset 0.0
} elseif {$iflg == 103} {
set offset 273.15
} else {
set offset 273.15
}

source ~IPCC/processing/tools/ipcc_intro.tcl

nap "tabove = 5.0"
nap "tconsec = 6"

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
```

```
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar tmax
} elseif {$iflg == 101} {
set ivar mxt2
} elseif {$iflg == 102} {
set ivar tmax
} elseif {$iflg == 103} {
set ivar mx2t
} else {
set ivar tasmax
}
set ovar hwdi

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]
```

```
if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

if {$iflg == 100} {
set pdir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set pdir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set pdir ${IPCC}/data/other/silo
} elseif {$iflg == 16 || ( $iflg >= 20 && $iflg <= 24 ) || $iflg == 26} {
set pdir ${IPCC}/data/${model_version}/20C3M/A4/run1
} else {
```

```
set pdir1 ${IPCC}/data/${model_version}/20C3M/A4/run2
set pdir2 ${IPCC}/data/${model_version}/20C3M/A4/run3
}

if {$iflg == 16 || ( $iflg >= 20 && $iflg <= 24 ) || $iflg == 26} {
set pfil tmaxnorm_A4_1961-1990${sea_lab}_20C3M_${model_version}_run1.nc
} elseif {$iflg == 100 || $iflg == 101 || $iflg == 102 || $iflg == 103} {
set pfil tmaxnorm_A4_1961-1990${sea_lab}_${expname}_${model_version}_${ensnum}.nc
} elseif {$iflg < 100} {
set pfil1 tmaxnorm_A4_1961-1990${sea_lab}_20C3M_${model_version}_run2.nc
set pfil2 tmaxnorm_A4_1961-1990${sea_lab}_20C3M_${model_version}_run3.nc
} else {
puts "Don't know iflg=$iflg yet."
return
}

if {[info vars pfil] != ""} {
puts "$ifils $pdir/$pfil"
dmget_tcl "$ifils $pdir/$pfil"
} elseif {[info vars pfil1] != "" && [info vars pfil2] != ""} {
puts "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
dmget_tcl "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
} else {
puts "Problem, pfil or pfil1&pfil2 should be defined."
}

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 400*$normt]
set normy [expr 400*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
```

```
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
```

```
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,
lat_ind,lon_ind"]-$offset "tim_ind,,"
```

```
set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  nap "day_ind_ynow_tmax = day_ind_concat_leap"
  incr total_days $nday_ind_concat_leap
```

```
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  nap "day_ind_ynow_tmax = day_ind_concat_noleap"
  incr total_days $nday_ind_concat_noleap
  }

if {[info vars pfil] != ""} {
nap "tmaxnorm = [nap_get netcdf $pdir/$pfil tmaxnorm "day_ind_ynow_tmax,
lat_ind,lon_ind"]-$offset"
} elseif {[info vars pfil1] != "" && [info vars pfil2] != ""} {
nap "tmaxnorm1 = [nap_get netcdf $pdir1/$pfil1 tmaxnorm "day_ind_ynow_tmax,
lat_ind,lon_ind"]-$offset"
nap "tmaxnorm2 = [nap_get netcdf $pdir2/$pfil2 tmaxnorm "day_ind_ynow_tmax,
lat_ind,lon_ind"]-$offset"
nap "tmaxnorm = (tmaxnorm1+tmaxnorm2)/2.0"
}

nap "va_year = iv(day_ind_ynow,,)"

nap "above = va_year>(tmaxnorm+tabove)"

set ivs_year [[nap shape(va_year)]]

nap "zeros = reshape(i32(0),{1 [lrange $ivs_year 1 2]})"

nap "b0 = transpose(transpose(zeros,{0 2 1})//transpose(above,{0 2 1}),{0 2 1})"
nap "b1 = transpose(transpose(above,{0 2 1})//transpose(zeros,{0 2 1}),{0 2 1})"

nap "i = i32(0 .. [lindex $ivs_year 0])"
nap "i_pad = transpose(reshape(i,{[lrange $ivs_year 1 2] [expr
[lindex $ivs_year 0]+1]}),{2 0 1})"

nap "va_tmp = reshape(f32(0),{[lrange $ivs_year 1 2]})"

for {set x 0} {$x < [lindex $ivs_year 2]} {incr x} {
for {set y 0} {$y < [lindex $ivs_year 1]} {incr y} {
nap "i_ext = i_pad(,$y,$x)"
nap "b0_ext = b0(,$y,$x)"
nap "b1_ext = b1(,$y,$x)"
nap "list = ((b0_ext && ! b1_ext) # i_ext) - ((b1_ext && ! b0_ext) # i_ext)"

nap "hwdi_ext = f32(max(list*(list>=tconsec)))>>>0"

$va_tmp set value hwdi_ext "$y,$x"
};#y
};#x

$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow
```

```
};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

14.66 ipcc_ei_pq95

ipcc_pq95.tcl is a Tcl-Nap script used to generate parameter *pq95*.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November  26 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set percentage "90"
set percentage "80"
set percentage "60"
set percentage "40"
set percentage "50"
set percentage "20"
set percentage "95"

set type "";#for all days percentiles.
set type "w";#for wet days only percentiles.

set wd_cutoff 1.0

set proc_name "ipcc_ei_pq${percentage}"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
```

```
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}

set iseas 0;#annual
set iseas 3;#JJA
set iseas 4;#SON
set iseas 2;#MAM
set iseas 1;#DJF
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set scale 1
} elseif {$iflg == 101} {
set scale 1
} elseif {$iflg == 102} {
set scale 1
} elseif {$iflg == 103} {
set scale 1
} else {
set scale 86400
}
```

```
source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar prate
} elseif {$iflg == 101} {
set ivar lsp
} elseif {$iflg == 102} {
set ivar rain
} elseif {$iflg == 103} {
set ivar precip
} else {
set ivar pr
}
set ovar pq${type}${percentage}

set time_freq "invariant"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

##########
set ybeg {1961};set yend {1990};set yinc 10;set yoff 0
##########

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}
```

```
for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];
set nend [lindex [$years v -format "%d"] end];set ndiff [expr $nend-$nbeg+1];
set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_
${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
```

```
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 100*$normt]
set normy [expr 100*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$nlats $nlons})"
```

```
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
```

```
set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons will be equal to
#number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,
lat_ind,lon_ind"]*$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]*$scale "tim_ind,,"
```

```
set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]*$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$type == "w"} {
nap "ivb=((iv>=$wd_cutoff)/(iv>=$wd_cutoff))*iv"
nap "va_tmp = percentile(ivb,${percentage})"
} else {
nap "va_tmp = percentile(iv,${percentage})"
}

$va set value va_tmp "lat_ind,lon_ind"

};#ytask
};#xtask

$va set coo lat lon
$va set dim lat lon
$va set missing 1e20
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg
```

```
};#isea

return
exit
```

14.67 ipcc_ei_r10

ipcc_r10.tcl is a Tcl-Nap script used to generate parameter *r10* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   3 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_r10"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}
```

```
set iseas 3;#JJA
set iseas 2;#MAM
set iseas 4;#SON
set iseas 1;#DJF
set iseas 0;#annual
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set scale 1
} elseif {$iflg == 101} {
set scale 1
} elseif {$iflg == 102} {
set scale 1
} elseif {$iflg == 103} {
set scale 1
} else {
set scale 86400
}

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
```

```
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar prate
} elseif {$iflg == 101} {
set ivar lsp
} elseif {$iflg == 102} {
set ivar rain
} elseif {$iflg == 103} {
set ivar precip
} else {
set ivar pr
}
set ovar r10

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];
set nend [lindex [$years v -format "%d"] end];set ndiff [expr $nend-$nbeg+1];
set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
```

```
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
```

```
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 200*$normt]
set normy [expr 200*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
```

```
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
```

```
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,
lat_ind,lon_ind"]*$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]*$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]*$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
```

```
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"
nap "va_tmp = sum(va_year>=10.0)"
$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow
```

```
};#iflg

};#isea

return
exit
```

14.68 ipcc_ei_r5d

ipcc_r5d.tcl is a Tcl-Nap script used to generate parameter *r5d* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   3 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set window "3"
set window "10"
set window "5"

set proc_name "ipcc_ei_r${window}d"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
```

```
set iflgs {103};set calendar "leap";#era40hr

set iseas 0;#annual

set iseas {0}

set iseas 1;#DJF
set iseas 3;#JJA
set iseas 4;#SON
set iseas 2;#MAM
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

set win ${window};#win should be odd number..
set winm1 [expr $win-1];set wind2 [expr $win/2];set wind2p1 [expr $wind2+1]

foreach iflg $iflgs {

if {$iflg == 100} {
set scale 1
} elseif {$iflg == 101} {
set scale 1
} elseif {$iflg == 102} {
set scale 1
} elseif {$iflg == 103} {
set scale 1
} else {
set scale 86400
}

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]
```

```
if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar prate
} elseif {$iflg == 101} {
set ivar lsp
} elseif {$iflg == 102} {
set ivar rain
} elseif {$iflg == 103} {
set ivar precip
} else {
set ivar pr
}
set ovar r${window}d

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
```

```
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_
${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
```

```
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 1500*$normt]
set normy [expr 1500*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
```

```
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
```

```
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
```

```
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"

set va_xy [lrange [[nap shape(va_year)]] 1 2]
set tlast  [expr [[nap nels(day_ind_ynow)]]-$wind2]
nap "va_tmp = reshape(0,{$va_xy})"

for {set tnow $wind2p1} {$tnow <= $tlast} {incr tnow} {
set t_beg [expr $tnow-$wind2p1];set t_end [expr $t_beg+$winm1]
nap "t_ind = $t_beg .. $t_end"
nap "va_tmp = va_tmp >>> sum(va_year(t_ind,,))"
};#tnow

$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
```

```
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

14.69 ipcc_ei_r95t

ipcc_r95t.tcl is a Tcl-Nap script used to generate parameter *r95t* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November  22 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set percentage "20"
set percentage "40"
set percentage "50"
set percentage "60"
set percentage "80"
set percentage "90"
set percentage "95"

set type "";#for all days percentiles.
set type "w";#for wet days only percentiles.

set wd_cutoff 1.0

set proc_name "ipcc_ei_r${percentage}t"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
```

```
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}

set iseas 0;#annual

set iseas 2;#MAM
set iseas 4;#SON

set iseas 1;#DJF
set iseas 3;#JJA
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set scale 1
} elseif {$iflg == 101} {
set scale 1
} elseif {$iflg == 102} {
set scale 1
} elseif {$iflg == 103} {
set scale 1
} else {
set scale 86400
```

```
}

source ~IPCC/processing/tools/ipcc_intro.tcl
set itab A2a;set otab A4

set IPCC [glob ~IPCC]

f {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar prate
} elseif {$iflg == 101} {
set ivar lsp
} elseif {$iflg == 102} {
set ivar rain
} elseif {$iflg == 103} {
set ivar precip
} else {
set ivar pr
}
set ovar r${type}${percentage}t

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {
```

```
if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
```

```
}

if {$iflg == 100} {
set pdir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set pdir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set pdir ${IPCC}/data/other/silo
} elseif {$iflg == 16 || ( $iflg >= 20 && $iflg <= 24 ) || $iflg == 26} {
set pdir ${IPCC}/data/${model_version}/20C3M/A4/run1
} else {
set pdir1 ${IPCC}/data/${model_version}/20C3M/A4/run2
set pdir2 ${IPCC}/data/${model_version}/20C3M/A4/run3
}

if {$iflg < 100} {
set pfil1 pq${type}${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run2.nc
set pfil2 pq${type}${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run3.nc
} elseif {$iflg == 100 || $iflg == 101 || $iflg == 102 || $iflg == 103} {
set pfil pq${type}${percentage}_A4_1961-1990${sea_lab}_${expname}_
${model_version}_${ensnum}.nc
}

if {$iflg < 100} {
puts "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
dmget_tcl "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
} else {
puts "$ifils $pdir/$pfil"
dmget_tcl "$ifils $pdir/$pfil"
}

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 200*$normt]
set normy [expr 200*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
```

```
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}
for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}

if {$iflg < 100} {

nap "pqp1 = [nap_get netcdf $pdir1/$pfil1 pq${type}${percentage} "lat_ind,lon_ind"]"
nap "pqp2 = [nap_get netcdf $pdir2/$pfil2 pq${type}${percentage} "lat_ind,lon_ind"]"

nap "pqp = (pqp1+pqp2)/2.0"
} else {
nap "pqp = [nap_get netcdf $pdir/$pfil pq${type}${percentage} "lat_ind,lon_ind"]*$scale"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
```

```
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
```

```
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt
```

```
if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"

if {$type == "w"} {
  nap "ivb = sum(va_year>=$wd_cutoff)"
  nap "ivc = ((va_year>=$wd_cutoff)/(va_year>=$wd_cutoff))*va_year"
  nap "ivd = sum(ivc>pqp)"
  nap "va_tmp = (ivd/ivb)*100.0"
} else {
  nap "ivd = sum(va_year>pqp)"
  nap "va_tmp = (ivd/nels(day_ind_ynow))*100.0"
}

$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
```

```
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

14.70 ipcc_ei_sdii

ipcc_sdii.tcl is a Tcl-Nap script used to generate parameter *sdii* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   3 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_sdii"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}
```

```
set iseas 0;#annual

set iseas 1;#DJF
set iseas 4;#SON
set iseas 3;#JJA
set iseas 2;#MAM
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set scale 1
} elseif {$iflg == 101} {
set scale 1
} elseif {$iflg == 102} {
set scale 1
} elseif {$iflg == 103} {
set scale 1
} else {
set scale 86400
}

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
```

```
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar prate
} elseif {$iflg == 101} {
set ivar lsp
} elseif {$iflg == 102} {
set ivar rain
} elseif {$iflg == 103} {
set ivar precip
} else {
set ivar pr
}
set ovar sdii

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
```

```
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
```

```
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 900*$normt]
set normy [expr 900*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
```

```
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
```

```
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"]
"tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]*
$scale "tim_ind,,"

set tim_beg [expr $tim_end+1]
```

```
  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"
nap "va_tmp = sum((va_year>=1.0)*va_year)/sum(va_year>=1.0)"
$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil
```

```
};#pnow

};#iflg

};#isea

return
exit
```

14.71 ipcc_ei_tmaxnorm

ipcc_tmaxnorm.tcl is a Tcl-Nap script used to generate parameter *tmaxnorm*.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November  15 2005

#this script assumes that for leap calendar input,
#that there is at least one day with a leap year in it...

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set proc_name "ipcc_ei_tmaxnorm"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr
```

```
set iseas {0}

set iseas 0;#annual
set iseas 2;#MAM
set iseas 3;#JJA
set iseas 4;#SON
set iseas 1;#DJF
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl
set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC
```

```
file mkdir $odir

if {$iflg == 100} {
set ivar tmax
} elseif {$iflg == 101} {
set ivar mxt2
} elseif {$iflg == 102} {
set ivar tmax
} elseif {$iflg == 103} {
set ivar mn2t
} else {
set ivar tasmax
}
set ovar tmaxnorm

set time_freq "daily"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

##########
set ybeg {1961};set yend {1990};set yinc 10;set yoff 0
##########

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];
set nend [lindex [$years v -format "%d"] end];set ndiff [expr $nend-$nbeg+1];
set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
```

```
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]
```

```
set normt 1000
set normx [expr 100*$normt]
set normy [expr 100*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$iflg == 100 || $iflg == 101 || $iflg == 102 || $iflg == 103} {
nap "vec1=nels(day_ind_leap)//$nlats//$nlons"
nap "vec2=nels(day_ind_leap)"
} else {
nap "vec1=nels(day_ind_noleap)//$nlats//$nlons"
nap "vec2=nels(day_ind_noleap)"
}

nap "va=reshape(f32(0),vec1)"
nap "va_cnt=reshape(f32(0),vec1)"
nap "time=reshape(f64(0),vec2)"

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}
```

```
set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
```

```
};#ynow

if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons will be equal to
#number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask
```

```
};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"

if {$isea == 0 && [expr $ynow%4] != 0 && $calendar == "leap"} {
#for ANN will need to seperate out Feb 29.

  nap "vec3=nels(day_ind_leap)//$nlat_ind//$nlon_ind"
  nap "vb_year=reshape(f64(0),vec3)"

  $vb_year set value va_year(0..58,,) "0..58,,"
  $vb_year set value va_year(59..364,,) "60..365,,"

  nap "va_tmp = va(,lat_ind,lon_ind)+vb_year"
  nap "vb_cnt_year = reshape(i32(0),vec3)"

  $vb_cnt_year set value 1 "0..58,,"
  $vb_cnt_year set value 1 "60..365,,"

  nap "va_cnt_tmp = va_cnt(,lat_ind,lon_ind)+vb_cnt_year"

} elseif {$isea == 1 && [expr $ynow%4] != 0 && $calendar == "leap"} {
#for DJF easy, just leave out last time,

  nap "vec3=nels(day_ind_leap)//$nlat_ind//$nlon_ind"
  nap "vb_year=reshape(f64(0),vec3)"
  $vb_year set value va_year(0..89,,) "0..89,,"
  nap "va_tmp = va(,lat_ind,lon_ind)+vb_year"
  nap "vb_cnt_year = reshape(i32(0),vec3)"
  $vb_cnt_year set value 1 "0..89,,"
  nap "va_cnt_tmp = va_cnt(,lat_ind,lon_ind)+vb_cnt_year"

} else {

  nap "va_tmp = va(,lat_ind,lon_ind)+va_year"
```

```
  nap "va_cnt_year = reshape(i32(1),shape(va_year))"
  nap "va_cnt_tmp = va_cnt(,lat_ind,lon_ind)+va_cnt_year"
}

#reassign back to the main array/counting array.
  $va set value va_tmp ",lat_ind,lon_ind"
  $va_cnt set value va_cnt_tmp ",lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask

nap "va = va/va_cnt"

#could fix this up so that dates actually represent something meaningful.
nap "nt0 = shape(va)"
nap "nt1 = nt0(0)-1"
nap "time = f64(0 .. nt1)"

$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

14.72 ipcc_ei_tn90

ipcc_tn90.tcl is a Tcl-Nap script used to generate parameter *tn90* defined in Table 18.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier September  8 2005
#Modified     November   3 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set percentage "10"
set percentage "90"

if {[format "%d" $percentage] != 10 && [format "%d" $percentage] != 90} {
puts "Percentage must be 10 or 90."
return
}

set proc_name "ipcc_ei_tn$percentage"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "yes"
set clobber "no"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
```

```
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}

set iseas 1;#DJF
set iseas 4;#SON
set iseas 2;#MAM
set iseas 0;#annual
set iseas 3;#JJA
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

if {$iflg == 100} {
set offset 273.15
} elseif {$iflg == 101} {
set offset 273.15
} elseif {$iflg == 102} {
set offset 0.0
} elseif {$iflg == 103} {
set offset 273.15
} else {
set offset 273.15
}

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]
```

```
if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar tmin
} elseif {$iflg == 101} {
set ivar mnt2
} elseif {$iflg == 102} {
set ivar tmin
} elseif {$iflg == 103} {
set ivar mn2t
} else {
set ivar tasmin
}
set ovar tn$percentage

set time_freq "yearly"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
```

```
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];
set nend [lindex [$years v -format "%d"] end];set ndiff [expr $nend-$nbeg+1];
set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]


if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_
${model_version}_${ensnum}.nc
  }
}

if {$iflg == 100} {
```

```
set pdir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set pdir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set pdir ${IPCC}/data/other/silo
} elseif {$iflg == 16 || ( $iflg >= 20 && $iflg <= 24 ) || ( $iflg >= 26 && $iflg <= 28 ) } {
set pdir1 ${IPCC}/data/${model_version}/20C3M/A4/run1
set pdir2 ${IPCC}/data/${model_version}/20C3M/A4/run2
set pdir3 ${IPCC}/data/${model_version}/20C3M/A4/run3
} else {
set pdir1 ${IPCC}/data/${model_version}/20C3M/A4/run2
set pdir2 ${IPCC}/data/${model_version}/20C3M/A4/run3
}

if {$iflg == 16 || ( $iflg >= 20 && $iflg <= 24 ) || ( $iflg >= 26 && $iflg <= 28 )} {
set pfil1 tnq${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run1.nc
set pfil2 tnq${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run2.nc
set pfil3 tnq${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run3.nc
} elseif {$iflg == 100 || $iflg == 101 || $iflg == 102 || $iflg == 103} {
set pfil tnq${percentage}_A4_1961-1990${sea_lab}_${expname}_${model_version}_${ensnum}.nc
} elseif {$iflg < 100} {
set pfil1 tnq${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run2.nc
set pfil2 tnq${percentage}_A4_1961-1990${sea_lab}_20C3M_${model_version}_run3.nc
} else {
puts "Don't know iflg=$iflg yet."
return
}

if {[info vars pfil] != ""} {
puts "$ifils $pdir/$pfil"
dmget_tcl "$ifils $pdir/$pfil"
} elseif {[info vars pfil1] != "" && [info vars pfil2] != "" && [info vars pfil3] != ""} {
puts "$ifils $pdir1/$pfil1 $pdir2/$pfil2 $pdir3/$pfil3"
dmget_tcl "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
} elseif {[info vars pfil1] != "" && [info vars pfil2] != ""} {
puts "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
dmget_tcl "$ifils $pdir1/$pfil1 $pdir2/$pfil2"
} else {
puts "Problem, pfil or pfil1&pfil2 should be defined."
}

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
set nlons [lindex $ivs 2]
```

```
set normt 1000
set normx [expr 300*$normt]
set normy [expr 300*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$ndiffm1 $nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$ndiff $nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}


if {[info vars pfil] != ""} {
nap "tnqp = [nap_get netcdf $pdir/$pfil tnq${percentage} "lat_ind,lon_ind"]-$offset"
} elseif {[info vars pfil1] != "" && [info vars pfil2] != "" && [info vars pfil3] != ""} {
```

```
nap "tnqp1 = [nap_get netcdf $pdir1/$pfil1 tnq${percentage} "lat_ind,lon_ind"]-$offset"
nap "tnqp2 = [nap_get netcdf $pdir2/$pfil2 tnq${percentage} "lat_ind,lon_ind"]-$offset"
nap "tnqp3 = [nap_get netcdf $pdir3/$pfil3 tnq${percentage} "lat_ind,lon_ind"]-$offset"
nap "tnqp = (tnqp1+tnqp2+tnqp3)/3.0"

} elseif {[info vars pfil1] != "" && [info vars pfil2] != ""} {
nap "tnqp1 = [nap_get netcdf $pdir1/$pfil1 tnq${percentage} "lat_ind,lon_ind"]-$offset"
nap "tnqp2 = [nap_get netcdf $pdir2/$pfil2 tnq${percentage} "lat_ind,lon_ind"]-$offset"
nap "tnqp = (tnqp1+tnqp2)/2.0"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]

#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
```

```
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow


if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask
```

```
} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,
lat_ind,lon_ind"]-$offset "tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0

};#dnt

if {$isea == 1} {
set yfirst [expr $nbeg+1]
} else {
set yfirst $nbeg
}

set total_days 0
for {set ynow $yfirst} {$ynow <= $nend} {incr ynow} {
set ynowmnbeg [expr $ynow-$yfirst]

  if {[expr $ynow%4] == 0} {
  nap "day_ind_ynow = day_ind_concat_leap + $total_days"
  incr total_days $nday_ind_concat_leap
  } else {
  nap "day_ind_ynow = day_ind_concat_noleap + $total_days"
  incr total_days $nday_ind_concat_noleap
  }

nap "va_year = iv(day_ind_ynow,,)"

if {$percentage == 10} {
nap "va_tmp = 100.0*sum(va_year<tnqp)/nels(day_ind_ynow)"
} elseif {$percentage == 90} {
nap "va_tmp = 100.0*sum(va_year>tnqp)/nels(day_ind_ynow)"
}

$va set value va_tmp "$ynowmnbeg,lat_ind,lon_ind"

  if {$xtask == 0 && $ytask == 0} {
  nap "time_year = time_join(day_ind_ynow)"
  nap "time_tmp = am(time_year)"
  $time set value time_tmp "$ynowmnbeg"
  };#x/ytask

};#ynow

};#ytask
};#xtask
```

```
$va set coo time lat lon
$va set dim time lat lon
$va set missing 1e20
$va netcdf -unlimited $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

14.73 ipcc_ei_tnq90

ipcc_ei_tn90.tcl
ipcc_tnq90.tcl is a Tcl-Nap script used to generate parameter *tnq90*.

```
#!/tools/tcl-nap/6.2.3/bin/tclsh
#CSIRO Australia
#Mark Collier November  22 2005
#Modified     November  22 2005

source ~IPCC/processing/tools/hostname.tcl

set hostname [hostname]

set percentage "10"
set percentage "90"

set proc_name "ipcc_ei_tnq$percentage"

source ~/.tclshrc

source ~col414/tcl_fun/dmget_tcl.tcl
source ~IPCC/processing/tools/ipcc_cmv.tcl
source ~IPCC/processing/tools/xyzt_edges.tcl

set clobber "no"
set clobber "yes"

set calendar "noleap";#365 day years
set calendar "leap";#365/366 day years

set iflgs {1 2 3 5 6 7};set calendar "noleap";#Bob's group
set iflgs {8 14 15 19};set calendar "noleap";#Tony's group
set iflgs {1};set calendar "noleap";set calendar "noleap";#Bob's/Mark's st1
set iflgs {2};set calendar "noleap";#Hal's/Bob's uk1
set iflgs {3};set calendar "noleap";#Bob's m20
set iflgs {5};set calendar "noleap";#Bob's s20
set iflgs {6};set calendar "noleap";#Bob's cm3
set iflgs {7};set calendar "noleap";#Bob's ct3
set iflgs {8};set calendar "noleap";#Tony's a1b
set iflgs {13};set calendar "noleap";#Martin's qv3
set iflgs {14};set calendar "noleap";#Tony's qv4
set iflgs {15};set calendar "noleap";#Tony's n20
set iflgs {19};set calendar "noleap";#Tony's o20
set iflgs {16};set calendar "noleap";#Hal's N35 PIcntrl
set iflgs {20};set calendar "noleap";#Tony's B35 20C3M
set iflgs {21};set calendar "noleap";#Tony's B35 SRESA2
set iflgs {22};set calendar "noleap";#Martin's M35 1pctto2x
set iflgs {23};set calendar "noleap";#Tony's D35 SRESA1B
set iflgs {24};set calendar "noleap";#Martin's E35 SRESB1
set iflgs {26};set calendar "noleap";#Martin's C35 Commit
set iflgs {27};set calendar "noleap";#Tony's F35 20C3M
set iflgs {28};set calendar "noleap";#Mark's G35 20C3M
set iflgs {36};#Leon's
set iflgs {100};set calendar "leap";#ncep
set iflgs {101};set calendar "leap";#era40
set iflgs {102};set calendar "leap";#silo
```

```
set iflgs {103};set calendar "leap";#era40hr

set iseas {0}

set iseas 0;#annual

set iseas 2;#MAM
set iseas 4;#SON
set iseas 1;#DJF
set iseas 3;#JJA
set iseas {1 2 3 4 0}

foreach isea $iseas {

if {$isea == 0} {
set sea_lab "_ANN"
} elseif {$isea == 1} {
set sea_lab "_DJF"
} elseif {$isea == 2} {
set sea_lab "_MAM"
} elseif {$isea == 3} {
set sea_lab "_JJA"
} elseif {$isea == 4} {
set sea_lab "_SON"
} else {
puts "Don't know that season/period."
return
}

source ~IPCC/processing/extreme_indices/ipcc_ei_days.tcl

set ndy_leap [[nap nmyr_leap(-1)]]
set ndy_noleap [[nap nmyr_noleap(-1)]]

foreach iflg $iflgs {

source ~IPCC/processing/tools/ipcc_intro.tcl

set itab A2a;set otab A4

set IPCC [glob ~IPCC]

if {$iflg < 100} {
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$iflg == 100} {
set idir ${IPCC}/data/other/ncep
set odir ${IPCC}/data/other/ncep
} elseif {$iflg == 101 || $iflg == 103} {
set idir ${IPCC}/data/other/era40
set odir ${IPCC}/data/other/era40
} elseif {$iflg == 102} {
set idir ${IPCC}/data/other/silo
set odir ${IPCC}/data/other/silo
} else {
puts "Those iflgs not setup yet."
```

```
return
}
set tdir /work/IPCC

file mkdir $odir

if {$iflg == 100} {
set ivar tmin
} elseif {$iflg == 101} {
set ivar mnt2
} elseif {$iflg == 102} {
set ivar tmin
} elseif {$iflg == 103} {
set ivar mn2t
} else {
set ivar tasmin
}
set ovar tnq$percentage

set time_freq "invariant"

ipcc_cmv $ovar xdim ydim zdim ldim lvars pnum levs ostn olgn ouni ovar table exter_scale
exter_offset land_sea_interp $time_freq var_derivation

source ~IPCC/processing/extreme_indices/ipcc_ei_runs.tcl

##########
set ybeg {1961};set yend {1990};set yinc 10;set yoff 0
##########

set yincm1 [expr $yinc-1]

if {[llength $ybeg] != [llength $yend]} {
puts "ybeg and yend have different lengths ([llength $ybeg]/[llength $yend])."
return
}

for {set pnow 0} {$pnow < [llength $ybeg]} {incr pnow} {

if {$yoff == 0} {
nap "years = [expr $ybeg+$yoff] .. $yend ... $yinc"
} else {
nap "years = $ybeg // [expr $ybeg+$yoff] .. $yend ... $yinc"
}
if {[[nap years(-2)+$yinc]] == $yend} {
nap "years = years // [[nap years(-1)]]"
}
set nbeg [lindex [$years v -format "%d"] 0];set nend [lindex [$years v -format "%d"] end];
set ndiff [expr $nend-$nbeg+1];set ndiffm1 [expr $ndiff-1]
set years_tcl [$years v -format "%d"]

if {$nbeg >= 2001 && $runname == "m20"} {
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
} elseif {$nbeg >= 2001 && $runname == "B35"} {
```

```
set expname "SRESA2"
set idir ${IPCC}/data/${model_version}/${expname}/${itab}/${ensnum}
set odir ${IPCC}/data/${model_version}/${expname}/${otab}/${ensnum}
}

set ofil ${ovar}_${otab}_${nbeg}-${nend}${sea_lab}_${expname}_${model_version}_${ensnum}.nc

puts "Output file $odir/$ofil"

if {[file isfile $odir/$ofil] == 1 && $clobber == "no"} {
continue
}

file delete -force $tdir/$ofil $odir/$ofil

set yincs_sum 0
catch {unset yincs}
set ylst [expr [llength $years_tcl]-1]
set ylstm1 [expr $ylst-1]

for {set ycnt 0} {$ycnt < $ylst} {incr ycnt} {
set ycntp1 [expr $ycnt+1]
if {$ycnt == $ylstm1} {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]+1]
} else {
lappend yincs [expr [lindex $years_tcl $ycntp1]-[lindex $years_tcl $ycnt]]
}
set yincs_sum [expr $yincs_sum+[lindex $yincs end]]
};#ycnt

catch {unset ifils}
for {set ycnt 0} {$ycnt < [expr [llength $years_tcl]-1]} {incr ycnt} {
set yrb [lindex $years_tcl $ycnt];
set yre [expr [lindex $years_tcl $ycnt]+[lindex $yincs $ycnt]-1]
  if {$yrb == $yre} {
  lappend ifils $idir/${ivar}_${itab}_${yrb}_${expname}_${model_version}_${ensnum}.nc
  } else {
  lappend ifils $idir/${ivar}_${itab}_${yrb}-${yre}_${expname}_${model_version}_${ensnum}.nc
  }
}

dmget_tcl "$ifils"

set ivs [nap_get netcdf -shape [lindex $ifils 0] ${ivar}]
nap "lat = [nap_get netcdf [lindex $ifils 0] lat]"
nap "lon = [nap_get netcdf [lindex $ifils 0] lon]"

set ntims [expr [lindex $ivs 0]/[lindex $yincs 0]*$yincs_sum]

if {$isea == 1} {
set ntims [expr $ntims/4-1]
} elseif {$isea > 1 && $isea < 5} {
set ntims [expr $ntims/4]
}

set nlats [lindex $ivs 1]
```

```
set nlons [lindex $ivs 2]

set normt 1000
set normx [expr 100*$normt]
set normy [expr 100*$normt]
set norm [expr $normx*$normy*$normt]
set ntasks [expr ($nlons*$nlats*$ntims)/($norm)]
set ntasksx [expr ($nlons*$ntims)/$normx]
set ntasksy [expr ($nlats*$ntims)/$normy]
set x1 [expr $nlons/$ntasksx]
set y1 [expr $nlats/$ntasksy]
if {$x1 <= 0} {
puts "Must increase normx until x1 is greater than 0."
return
}
if {$y1 <= 0} {
puts "Must increase normy until y1 is greater than 0."
return
}
nap "xvec_n = 0 .. [expr $nlons-1] ... $x1"
nap "yvec_n = 0 .. [expr $nlats-1]... $y1"
set xvec_t [$xvec_n v]
set yvec_t [$yvec_n v]
set xvec_sh [[nap shape(xvec_n)]]
set yvec_sh [[nap shape(yvec_n)]]
set xvec_shm1 [expr $xvec_sh-1]
set yvec_shm1 [expr $yvec_sh-1]
puts "Number of tasks = [expr $xvec_shm1*$yvec_shm1]"

if {$isea == 1} {
nap "va=reshape(f32(0),{$nlats $nlons})"
nap "time=reshape(f64(0),$ndiffm1)"
} else {
nap "va=reshape(f32(0),{$nlats $nlons})"
nap "time=reshape(f64(0),$ndiff)"
}

for {set xtask 0} {$xtask < $xvec_shm1} {incr xtask} {
set xtaskp1 [expr $xtask+1]

if {$xtask == 0} {
nap "lon_ind = [lindex $xvec_t $xtask] .. [lindex $xvec_t $xtaskp1]"
} else {
nap "lon_ind = [expr [lindex $xvec_t $xtask]+$xtask*1/$xtask] .. [lindex $xvec_t $xtaskp1]"
}

for {set ytask 0} {$ytask < $yvec_shm1} {incr ytask} {
set ytaskp1 [expr $ytask+1]
if {$ytask == 0} {
nap "lat_ind = [lindex $yvec_t $ytask] .. [lindex $yvec_t $ytaskp1]"
} else {
nap "lat_ind = [expr [lindex $yvec_t $ytask]+$ytask*1/$ytask] .. [lindex $yvec_t $ytaskp1]"
}

set nlat_ind [[nap nels(lat_ind)]]
set nlon_ind [[nap nels(lon_ind)]]
```

```
#these loops for computing length of daily fields...
set total_days 0
set total_days_noskip 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {
set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$dnt == 0 && $ynow == $dnow && $isea == 1} {
  if {[expr $dnow%4] == 0} {
  set skip0 $skip0_leap
  } else {
  set skip0 $skip0_noleap
  }
} else {
set skip0 0
}
  if {[expr $ynow%4] == 0} {
  incr total_days [expr $nday_ind_concat_leap-$skip0]
  } else {
  incr total_days [expr $nday_ind_concat_noleap-$skip0]
  }
};#ynow
};#dnt

nap "iv = reshape(f64(0),{$total_days $nlat_ind $nlon_ind})"
nap "time_join = reshape(f64(0),{$total_days})"

set tim_beg 0
for {set dnt 0} {$dnt <= [expr [llength $yincs]-1]} {incr dnt} {

set dnow [lindex $years_tcl $dnt]
set dnowpinc [expr $dnow+[lindex $yincs $dnt]-1]

set total_days 0
for {set ynow $dnow} {$ynow <= $dnowpinc} {incr ynow} {
if {$ynow == $dnow} {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_long =  day_ind_leap"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_long =  day_ind_noleap"
  incr total_days $ndy_noleap
  }
} else {
  if {[expr $ynow%4] == 0} {
  nap "day_ind_add =  day_ind_leap + $total_days"
  incr total_days $ndy_leap
  } else {
  nap "day_ind_add =  day_ind_noleap + $total_days"
  incr total_days $ndy_noleap
  }
nap "day_ind_long = day_ind_long // day_ind_add"
}
};#ynow
```

```
if {$dnt == 0 && $isea == 1} {
#First December is missing, and so number of seasons
#will be equal to number of years minus 1.
#later compress time by averaging in these loops.

  if {[expr $dnow%4] == 0} {
  nap "day_ind_djf = day_ind_long(nmyr_leap(1)..nels(day_ind_long)-1)"
  } else {
  nap "day_ind_djf = day_ind_long(nmyr_noleap(1)..nels(day_ind_long)-1)"
  }

set tim_end [expr $tim_beg+[[nap nels(day_ind_djf)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_djf,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {

$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_djf"] "tim_ind"

  };#x/ytask
continue
};#dnt/isea

if {$dnt == 0 && $isea != 1} {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

} else {

set tim_end [expr $tim_beg+[[nap nels(day_ind_long)]]-1]
nap "tim_ind = $tim_beg .. $tim_end"

$iv set value [nap_get netcdf [lindex $ifils $dnt] ${ivar} "day_ind_long,lat_ind,lon_ind"]
"tim_ind,,"

set tim_beg [expr $tim_end+1]

  if {$xtask == 0 && $ytask == 0} {
$time_join set value [nap_get netcdf [lindex $ifils $dnt] time "day_ind_long"] "tim_ind"
  };#x/ytask

};#dnt==0
```

```
};#dnt

nap "va_tmp = percentile(iv,$percentage)"
$va set value va_tmp "lat_ind,lon_ind"

};#ytask
};#xtask

$va set coo lat lon
$va set dim lat lon
$va set missing 1e20
$va netcdf $tdir/$ofil $ovar

source ~IPCC/processing/tools/ipcc_dims.tcl
source ~IPCC/processing/tools/ipcc_att.tcl
source ~IPCC/processing/tools/ipcc_bnds_agcm.tcl

nap "missing_value = f32(1e20)"
$missing_value netcdf $tdir/$ofil $ovar:missing_value

puts "Moving $tdir/$ofil to $odir/$ofil ..."
file rename $tdir/$ofil $odir/$ofil

};#pnow

};#iflg

};#isea

return
exit
```

15 Disclaimer

The CSIRO IPCC Mk3.0 experimental data may be modified or updated at any time without any prior warning. Users of this model output are urged to check and reassure themselves of its numerical integrity and applicability to their individual research applications. If you have any questions regarding this document or data described within, please contact the author, preferably by e-mail.

CSIRO Marine and Atmospheric Research Papers

No. 001 **Whetton, P.H., McInnes, K.L., Jones, R.N., Hennessy, K.J., Suppiah, R., Page, C.M., Bathols, J. and Durack, P.J.** Australian Climate Change Projections for Impact Assessment and Policy Application. 2005. Electronic edition only.

No. 002 **Frederiksen, J.S. and Frederiksen C.** Decadal Changes in Southern Hemisphere Winter Cyclogenesis. 2006. Electronic edition only.

No. 003 **Collier, M.A.** The CSIRO ERA-40 2.5°×2.5° archive. 2005. Electronic edition only.

No. 004 **Davies, H.L.** Nap 6.3 User's Guide. 2007 Electronic edition only.

No. 005 **Fandry, C.B, Slawinski, D. and Penter, L.F.** Two-rocks moorings data report. 2006. Electronic edition only.

No. 006 **Pendrey, R.C.** Central heating for marine aquarium tanks. 2006. Electronic edition only.

No. 007 **Young, S.A.** The Cape Grim MiniLidar data set 1998-2000 data coverage, file format and reading software. 2006. Electronic edition only.