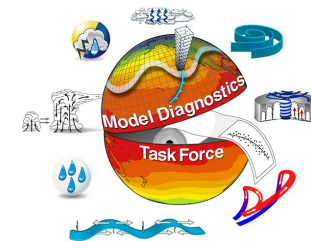


Model Diagnostics Task Force

A Walkthrough of the Technical Vision and the Diagnostics Packages

Aparna Radhakrishnan^{1,3,4}, Wenhao Dong^{2,3,4}
And MDTF Leads group

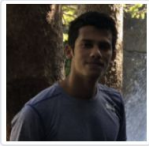
1. Princeton University CIMES, 2. UCAR, 3. NOAA GFDL, 4. MDTF



UFS Webinar Series, Dec 9th, 2021

Acknowledgments

Model Diagnostics Task Force Framework Team



Fiaz Ahmed
Co-I
UCLA



Dani Coleman
Software developer
NCAR



Wenhao Dong
Postdoctoral Researcher
GFDL



Jeffrey Durachta
MSD lead and resource
manager
GFDL



Andrew Gettelman
Co-PI
NCAR



John Krasting
Co-PI
GFDL



Jessica Liptak
Lead software developer
GFDL



Eric Maloney
Co-PI
Colorado State University



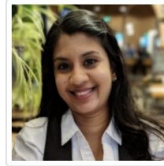
Elizabeth Maroon
University of Wisconsin



Yi Ming
Co-PI
GFDL



David Neelin
Lead PI
UCLA



**Aparna
Radhakrishnan**
Co-PI
GFDL



Paul Ullrich
Co-PI
UC Davis



Allison Wing
University PI member of
leads

MDTF Task Force.

Other contributors: Jack Chen, Thomas Jackson, Yi-Hung Kuo, Elizabeth Maroon, Ana Ordonez, Joyce Meyerson, Rayette Tolles-Abdullah, Unni Kirandumkara, Thomas Robinson, Tristan Dietz, Krishna Kumar, V. Balaji, Rusty Benson.

Funding acknowledgment: NOAA MAPP program, OAR CTI and OAR RDHPCS cloud awards

Other resources: Open-source GitHub actions templates and docs by the community.



Outline

1. Background and Motivation
2. Intro: Process Oriented Diagnostics and MDTF
3. Highlights: MDTF Framework Phase-3
4. Phase-3 and beyond: What are the 3Cs?
5. PODs Deep-dive: MJO PODs and Mesoscale Convective System over North America
6. Conclusion

Challenges: Need for details

The simulation of clouds in climate models remains challenging.

[IPCC AR5, Chapter 9]



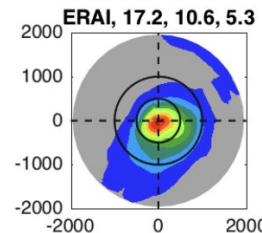
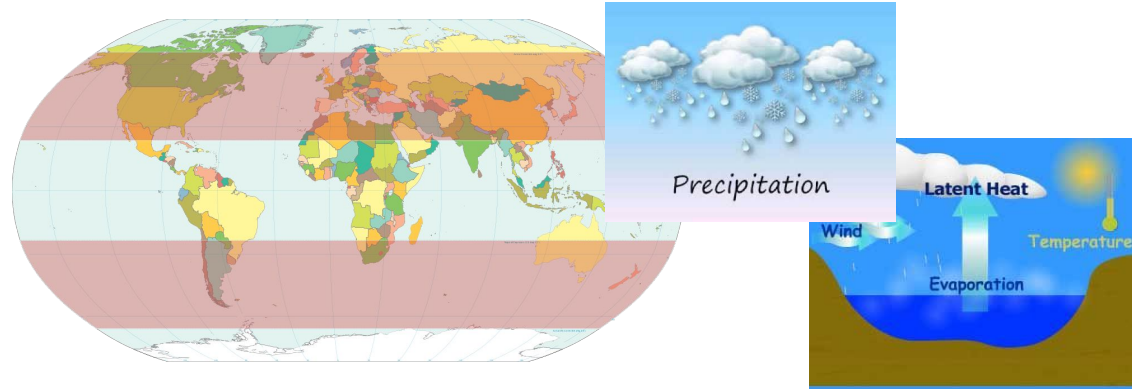
The majority of models underestimate the sensitivity of extreme precipitation to temperature variability or trends, especially in the tropics, which implies that models may underestimate the projected increase in extreme precipitation in the future.

[IPCC AR5, Chapter 9]

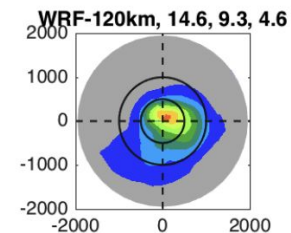
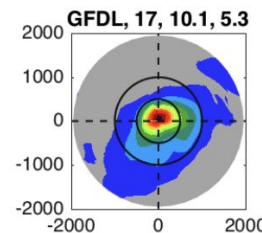


Storm track biases in the North Atlantic have improved slightly, but models still produce a storm track that is too zonal and underestimate cyclone intensity.

[IPCC AR5, Chapter 9]



Composite Mean Rain Rate



Extratropical cyclones (ETCs) are responsible for the majority of wintertime precipitation in the midlatitudes. e.g. , Hawcroft et al. 2012

Cyclone-centered composite mean precipitation J.Booth, 2017

Representation of several processes and their feedback are important to study extratropical cyclones and for mid-latitude precipitation simulation

Process Oriented Diagnostics

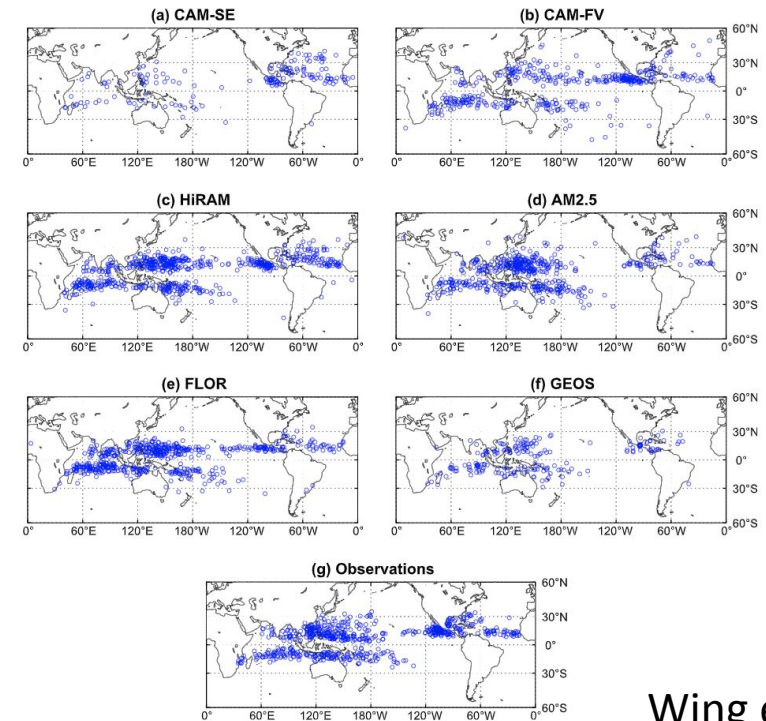
Processes tell us about the model state, or how the model state changes in space and time.
It's not just the state, it's the derivative of the state with respect to other things or with respect to time

Gettelman et al. AGU 2020

Increasingly, models are being analyzed in more detail against observations of **specific processes**, and the MDTF is approaching PODs in this spirit. The closer to a model process the observations and evaluation are, the better the ability to constrain the process and hence provide a guide to parameterization improvement.

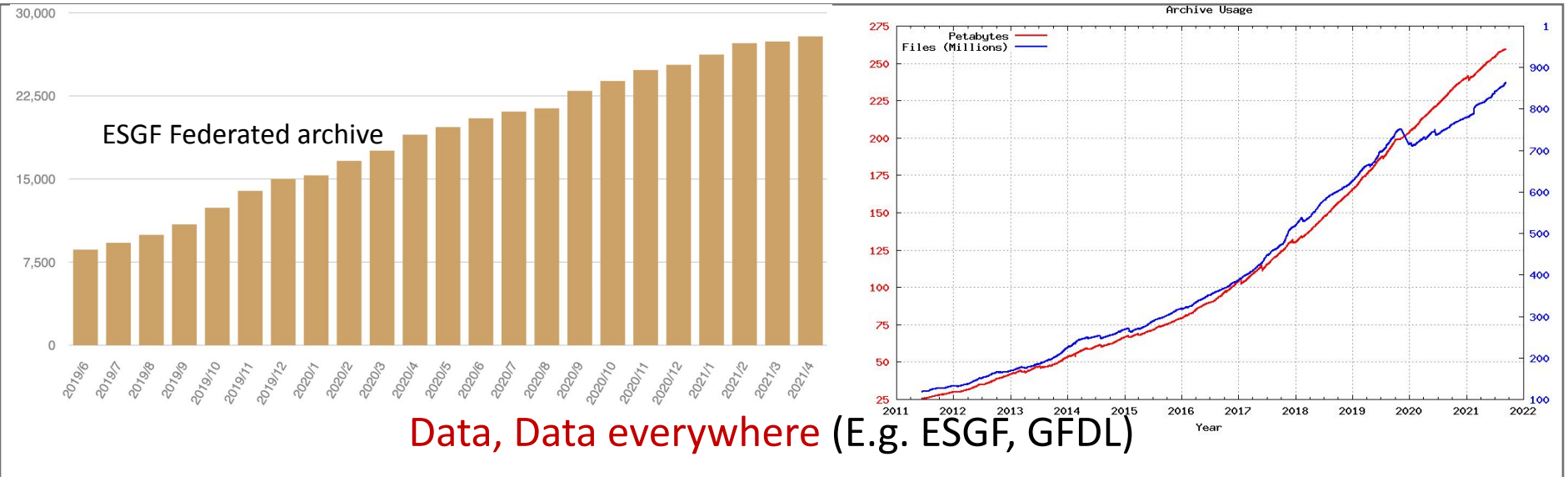
Maloney et al, 2019

Develop process-oriented diagnostics to identify model characteristics that are responsible for proper simulation of TCs and that will explain the inter-model spread in TC frequency and intensity distributions. These diagnostics go beyond simply quantifying the simulated TC activity and focus on how simulated TCs respond to their environments..



Wing et al, 2019

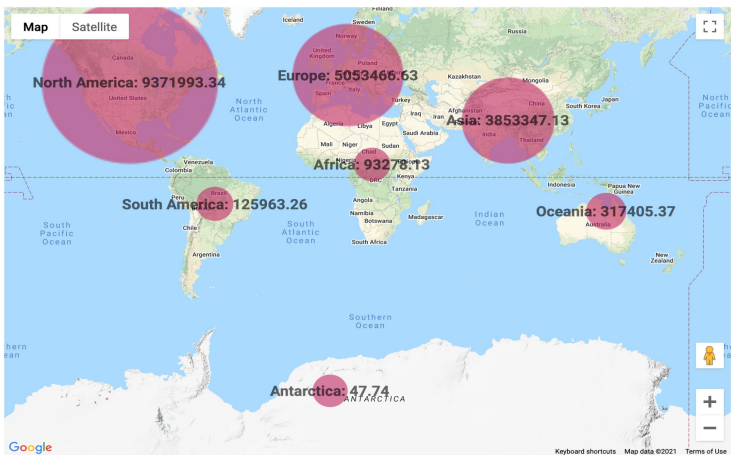
Need for collaboration



VARYING: experimental design, model output formats, variable names, units, resolution,....



Experts everywhere. (extends worldwide)



Diverse users (ESGF example only, from CMCC portal)



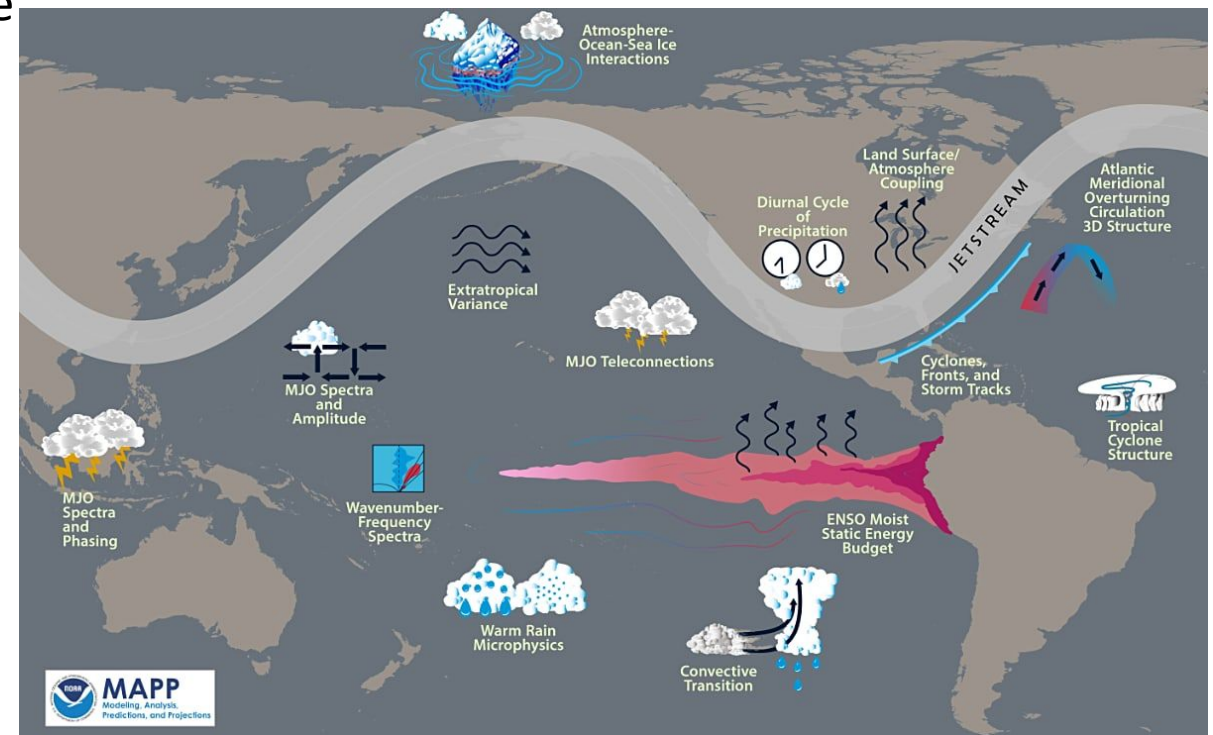
Need results now

Model Diagnostics Task Force: History and background

Engage research community developing **process-oriented diagnostics (PODs)** to improve the performance of Earth system models at **NOAA** , **NCAR**, **DOE**.

Multi-institutional effort funded through the Model Analysis, Prediction, and Projection (MAPP) program at NOAA's Climate Program Office

- History dates back to 2012 CLIVAR MJO Task Force
- Phase 1 framework development led by NCAR (2015-2018)
 - Public release of “version 2”
 - Initial set of 8 process-oriented diagnostics (PODs)
- **Phase 2 and 3 development led by GFDL (2018-2022, 2021-2024)**
 - More contributed diagnostics
 - Move toward open source software model
 - Lead PI - David Neelin (UCLA)
 - Collaboration, collaboration.



Expanding suite of **diagnostics**

Atmosphere

Convective transition diagnostics
MSE variance of tropical cyclones
Precip. buoyancy diagnostics
Precip. Distribution diagnostics
Rossby wave sources
Surface flux diagnostics
Top-heaviness metrics
TUTT index
Vertical processes

***Green denotes** diagnostics that were part of the 2018-22 MAPP funding opportunity.

Expanded set of new PODs are coming in for the new phase.

Ocean

Arctic Ocean ML diagnostics
Arctic surface SW feedbacks

Marine Ecosystems CPT

CA Current biogeo. processes
Northeast Pacific SSTs
Time evolution of biogeo. variables
Time evolution of marine heat waves

Land Climate Process Team

Convective triggering
Heated condensation
ILAMB
LCL deficit
Mixing diagrams
RH tendency

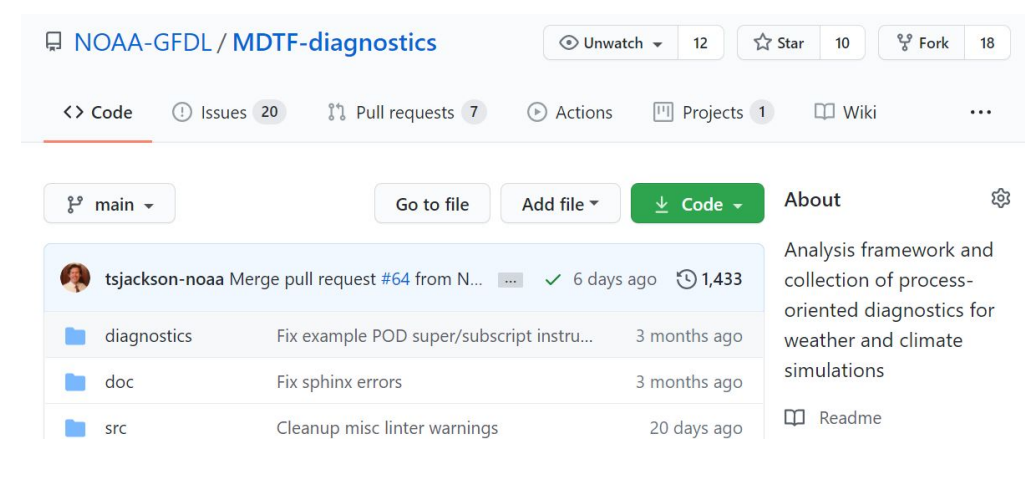
Climate Sensitivity Task Force

Boundary layer clouds
Cloud controlling factors
Emergent constraints
Humidity cloud circulation
Kernel-based climate feedbacks
Low clouds

Framework Software Enhancements (Phase-2)

T. Jackson, D. Coleman, Y. H. Kuo

- Migrated to open source development on GitHub
- Anaconda-based installation routines
- Migrated to Python 3.x
- Adoption of CMOR/CF variable naming conventions
- Detection of internal GFDL formats vs. CMIP output
- Improved unit and workflow testing
- Introduced settings.jsonc file
- Collaborated with DOE's Coordinated Metrics and Evaluation Capabilities (CMEC) on a joint CMEC-MDTF standard for setting and naming conventions

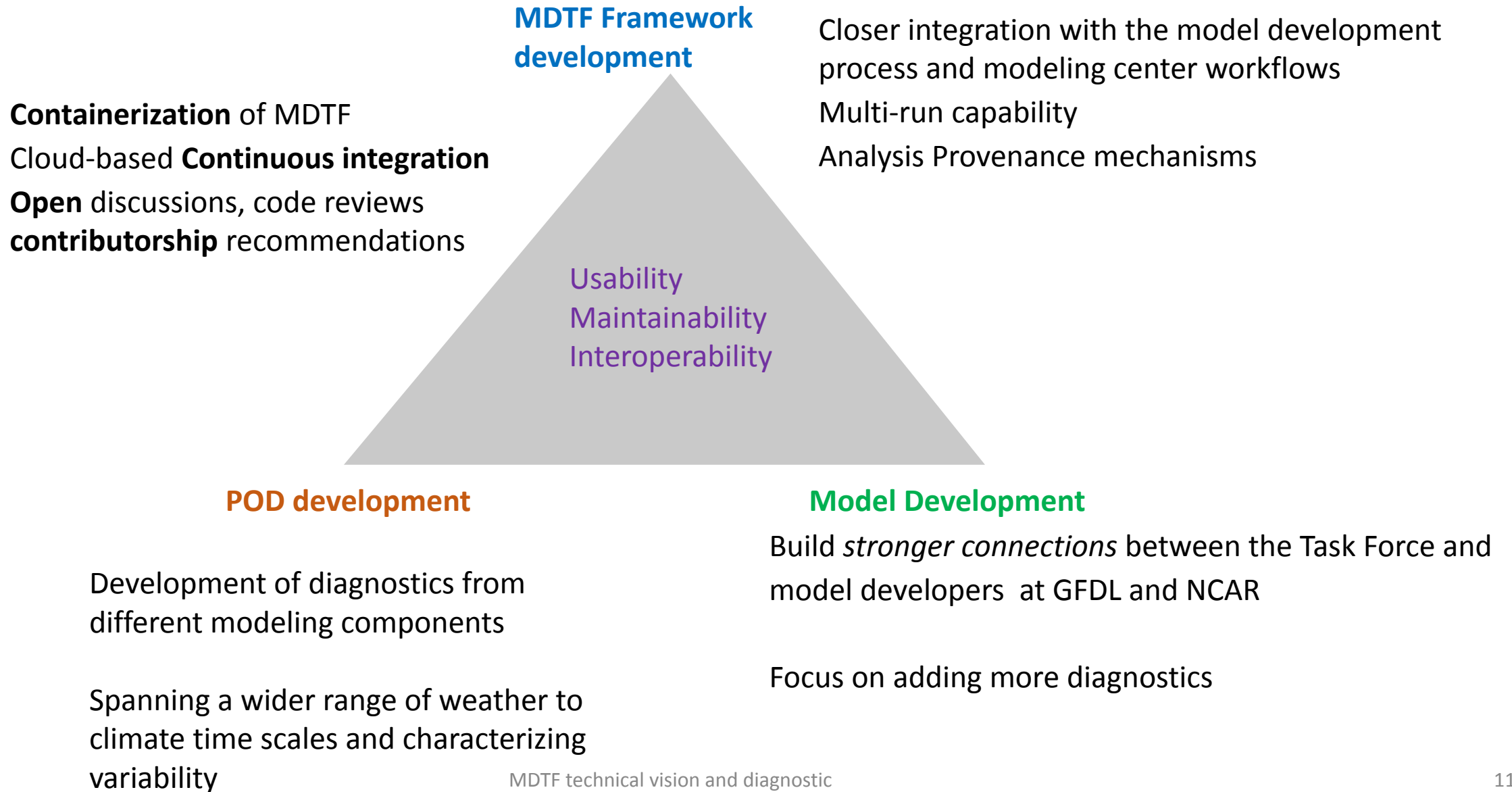


MDTF Diagnostic code now hosted at <https://github.com/NOAA-GFDL/MDTF-diagnostics>

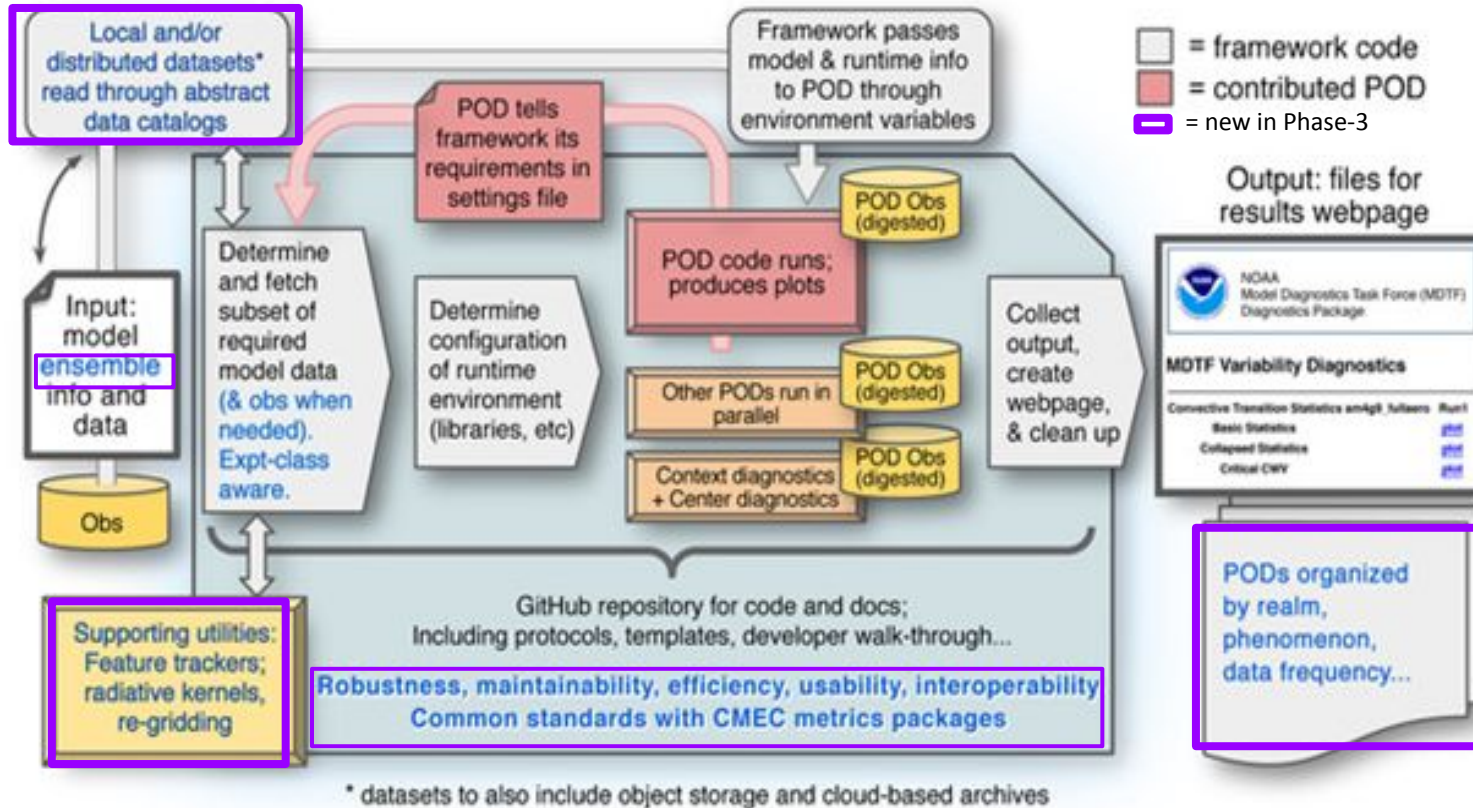
MDTF-diagnostics framework: software for running process-oriented diagnostics (PODs) on climate and weather data.
Write-once and Run anywhere!

Developments centered on increasing community *participation*, improving the software *portability*, and *providing clear documentation* to developers and end-users.

MDTF Phase-3 and beyond



Framework **overview** (Phase-3)



Scientist-developers **contribute** PODs to the MDTF-diagnostics repository

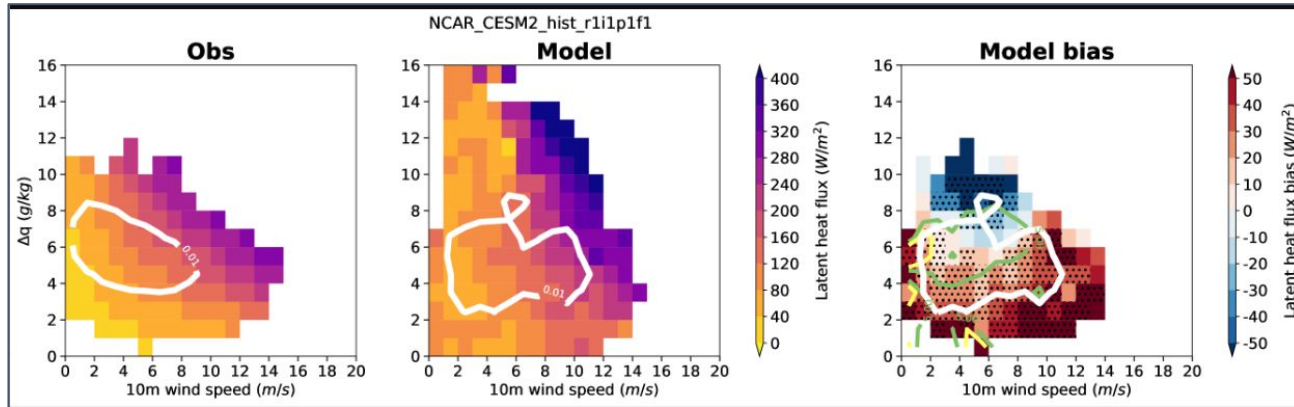
MDTF-diagnostics **framework**

- configures the **environments** to run desired PODs
- **processes the input data** to match the specified standard (CMIP, GFDL, CESM/NCAR)
- **runs the PODs**
- organizes the **output files into a webpage**

Python-based driver handles **data ingest** and manipulation, calls the **diagnostic routines**, and **generates an html page** with the results.

MDTF POD output webpage example

POD: Ocean Flux Matrix



The screenshot shows a web browser window with the title 'MDTF ocean surface flux diagnostic'. The address bar shows the file path: `/home/jessica.liptak/mdtf/wkdir/MDTF_NCARM_CESM2_hist_r1i1p1f1_2000_2009/ocn...`. The page header includes the NOAA logo and the text 'NOAA Model Diagnostics Task Force (MDTF) Diagnostics Package'. The main content area is titled 'Example diagnostic: ocean surface flux diagnostic'. Below the title is a paragraph of text explaining the diagnostic. At the bottom, there are two links: 'Time period: 2000-2009 diagnostic matrix for latent heat flux' and 'NCAR_CESM2_hist_r1i1p1f1 OBS plot'. Two blue arrows point from the 'plot' link to the 'Model bias' plot in the figure above.

Example diagnostic: ocean surface flux diagnostic

The example plot shows the latent heat flux diagnostic based on observational dataset (left) from the TAO/TRITON and RAMA arrays. The latent heat flux values are binned based on the corresponding surface 10m wind speed (x direction) and moisture disequilibrium of specific humidity at the surface (y direction). The white contour shows the 1% data points of all available points. The latent heat flux diagnostic based on model is shown in the middle. The right panel shows the difference between the model and observational data. The hatched area shows the difference between model and observation are statistically significant with 99% confidence. The yellow and green contours show the 5 mm/day and 10 mm/day values associated with the surface wind speed and moisture disequilibrium values.

Time period: 2000-2009
diagnostic matrix for latent heat flux

NCAR_CESM2_hist_r1i1p1f1 OBS

POD developers provide documentation for their POD

POD developer: Chia-Weh Hsu
<https://github.com/chiaweh2>

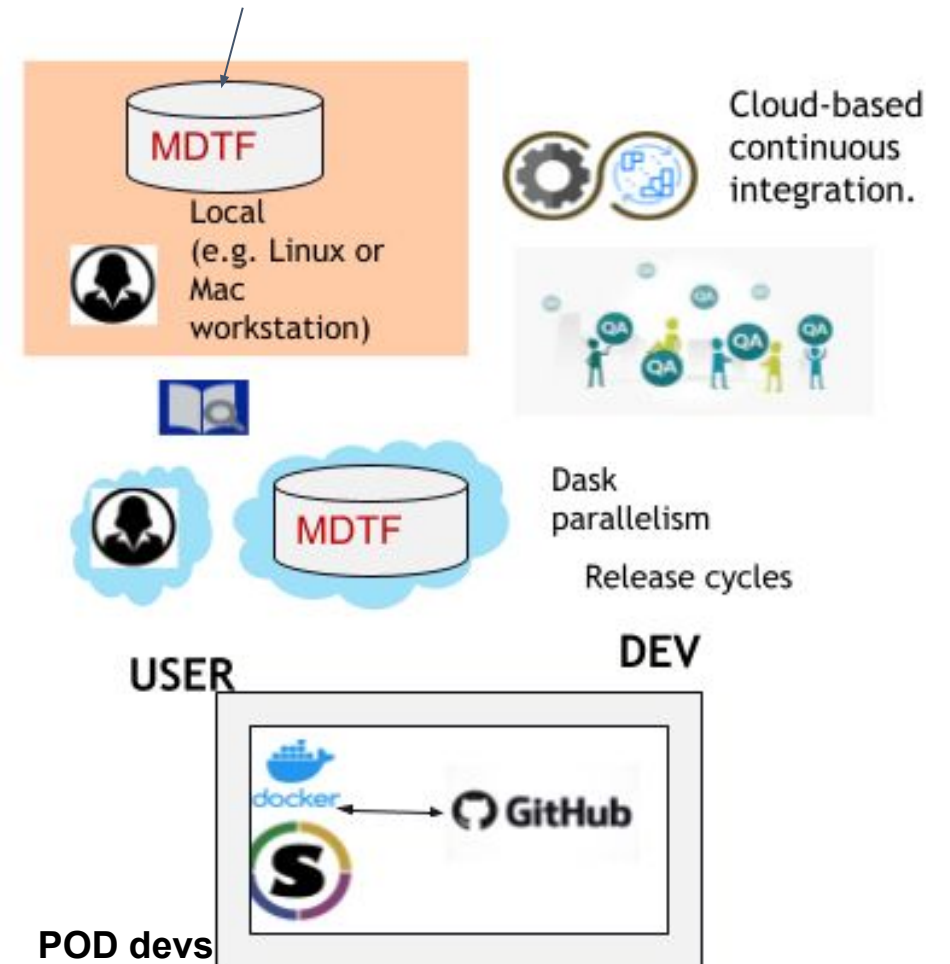
3Cs: Containers, CI and cloud

Containers for simplified, portable, repeatable software.

Continuous Integration Review, Test, Merge often, Automate builds and testing with scalable resources (**CLOUD**)

- Incorporate lessons learned from the above exercise and Task Force feedback to build our **production container** image.
- Leverage community-developed data cataloguing APIs and **cloud-optimized analysis workflows** (e.g.xarray, dask)
 - MDTF PODs in the cloud using existing publicly hosted cloud-data (e.g. CMIP6/netCDF/Zarr under ASDI initiative)

Container Recipe with: MDTF framework
code+environment+dependency



Now: CI pipeline

build (ubuntu-latest, /usr/share/miniconda3, tests/github_actions



GitHub hosted runners

On push, PR

- > Download miniconda 3
- > ✓ Verify miniconda
- > ✓ Install XQuartz if macOS
- > ✓ Set environment variables
- > ✓ Install Conda Environments
- > ✓ Generate Model Data
- > ✓ Get Observational Data for Set 1
- > ✓ Run diagnostic tests set 1
- > ✓ Get observational data for set 2
- > ✓ Run diagnostic tests set 2
- > ✓ Get observational data for set 3
- > ✓ Run diagnostic tests set 3
- > ✓ Run unit tests
- > ✓ Post Download Miniconda 3
- > ✓ Post Run actions/checkout@v2
- > ✓ Complete job

Code review

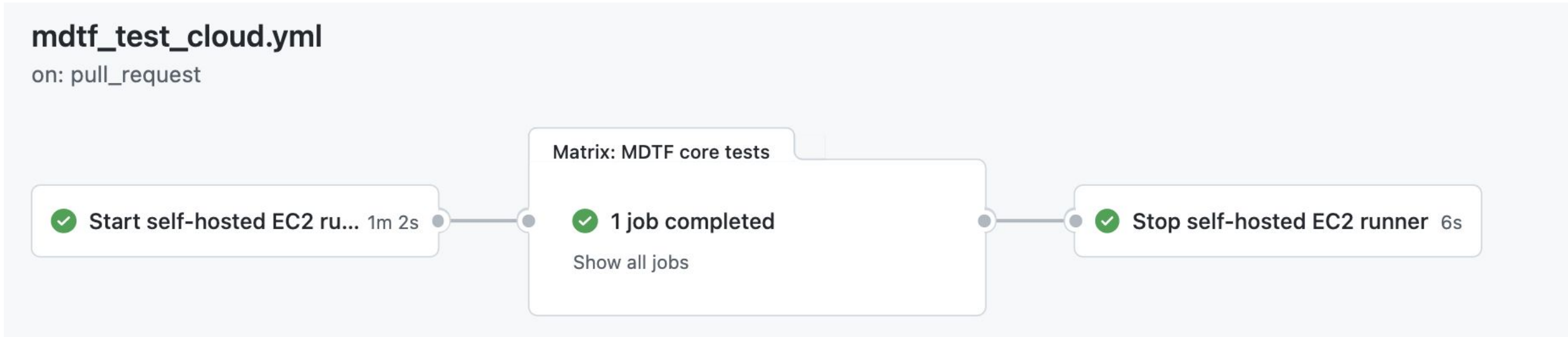
Unit testing
PODs run with synthetic model data generated by the mdtf-test-data package

The screenshot shows a GitHub Actions CI pipeline status page. At the top, there is a red banner indicating 'Changes requested' with 2 reviews. Below this, there is a section for '2 changes requested'. The main status is 'All checks have passed' with 3 successful checks. The checks listed are: 'MDTF_test / build (ubuntu-latest, /usr/share/miniconda3, tests/github_actions_test_ubuntu_s...', 'MDTF_test / build (macos-latest, /Users/runner/miniconda3, tests/github_actions_test_maco...', and 'LGTM analysis: Python' which is successful in 9m. At the bottom, there is a red banner indicating 'Merging is blocked' because merging can only be performed automatically once the requested changes are addressed.

LGTM software runs linter and security analysis on Python code

Use of GitHub actions and GitHub hosted runners for Continuous integration

In progress: Test#1 CI pipeline using Amazon self-hosted elastic compute cloud runners (EC2)



GitHub Actions

Use of elastic compute instances in AWS as self-hosted runners to explore scalable and cost-optimized solutions. Basic testing routines remain the same.

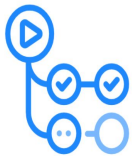


Amazon
EC2

In progress: Test#2 CI pipeline using AWS ECS and CodeBuild

Deploy
succeeded 20 days ago in 22m 52s

- > ✓ Set up job
- > ✓ Checkout
- > ✓ Configure AWS credentials
- > ✓ Login to Amazon ECR
- > ✓ Build, tag, and push image to Amazon ECR
- > ✓ Fill in the new image ID in the Amazon ECS task definition
- > ✓ Deploy Amazon ECS task definition
- > ✓ Post Login to Amazon ECR
- > ✓ Post Configure AWS credentials
- > ✓ Post Checkout
- > ✓ Complete job



GitHub Actions



For application tests
and status-reporting

Deploy to Amazon ECS
Deploy a container to an Amazon ECS service powered by AWS Fargate or Amazon EC2.



Set up this workflow

```
# Build a docker container and  
# push it to ECR so that it can  
# be deployed to ECS.
```

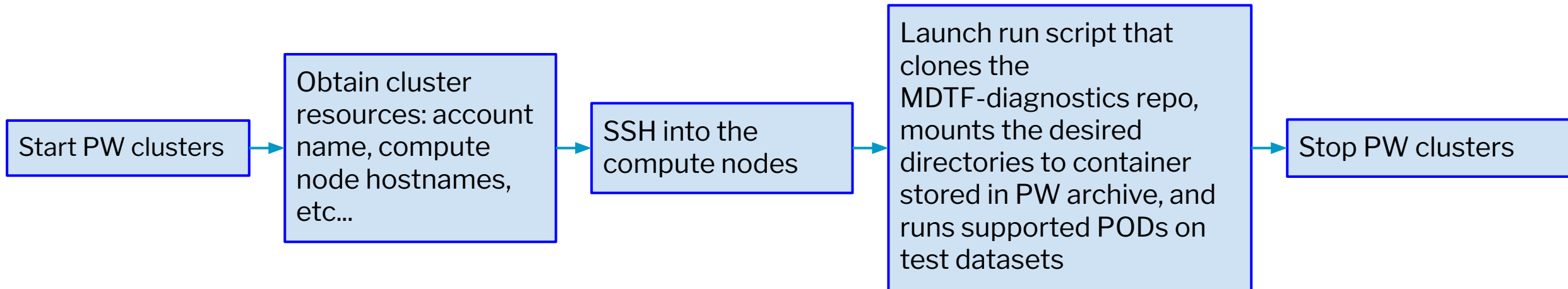
actions/starter-workflows

Automation
Scalability
Convenience
Cost-optimized
Usability

Use of AWS Elastic Container Registry (ECR) to build/push MDTF docker image. Use of Elastic Container Service (ECS) to deploy MDTF application. Basic testing routines to be run using CodeBuild.

In progress: Test#3 CI pipeline using Google Cloud

Use of GitHub actions, Singularity containers and Google Cloud cluster in NOAA Parallelworks infrastructure to explore CI pipelines



GitHub Actions



Google Cloud

Slide credit: Jess Liptak

Part-1: Take home message

MDTF:

- Provides and Promotes **unified,open** frameworks for **process oriented diagnostics**
- Leverages use of CMOR/CF **conventions**
- Collaborates on metrics and diagnostic standards with **CMEC**
- Enables researchers focus more on actual scientific research
 - **Write once, run everywhere**
- Strives to simplify user experience (**3Cs + D**-ocumentation)
- Fosters engagement with **modeling centres** (NCAR, GFDL)
- Facilitates model **evaluation** and model **development**

1. MDTF framework

Take MJO PODs for example

Apply them to three GFDL models (AM4.0, CM4.0, ESM4.0)

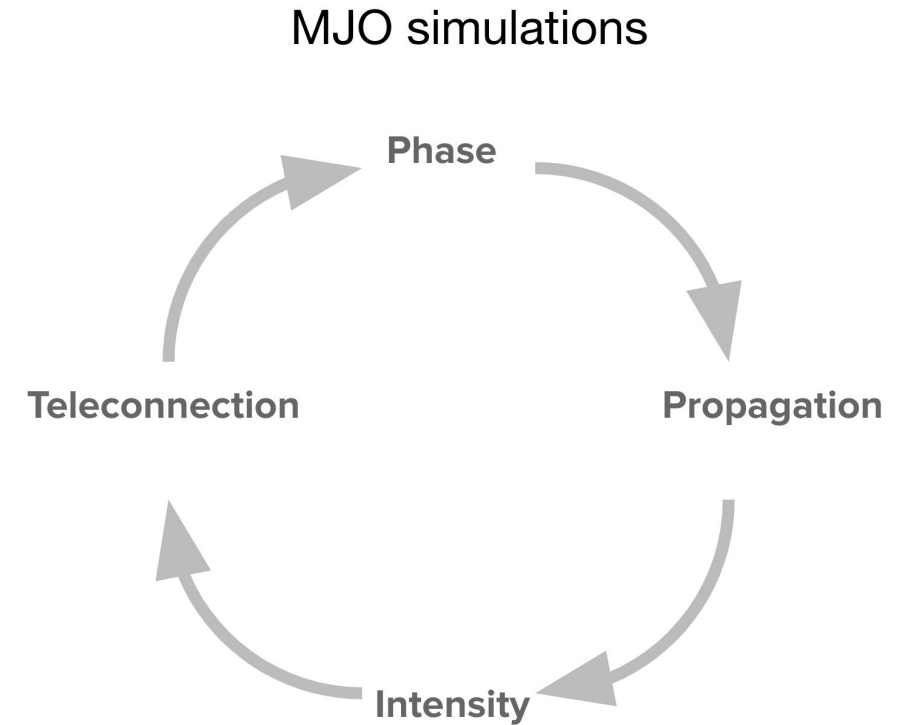
1. In-house development

Mesoscale Convective System over North America

Acknowledgement: Y. Ming, J. Krasting, T. Jackson

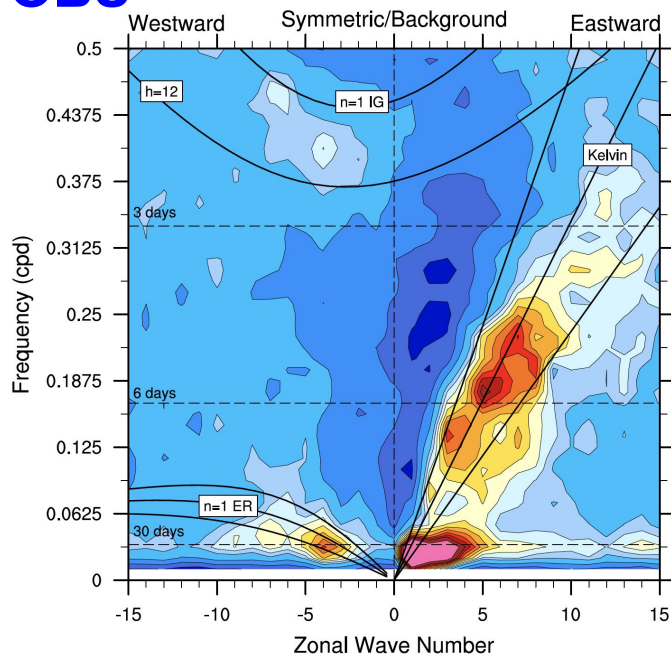
Process-oriented diagnostics (PODs)

- Convective transition
- Diurnal cycle of precipitation
- Extratropical variance
- Wavenumber-frequency spectra
- MJO prop and amp
- MJO spectra and phasing
- MJO teleconnection
-

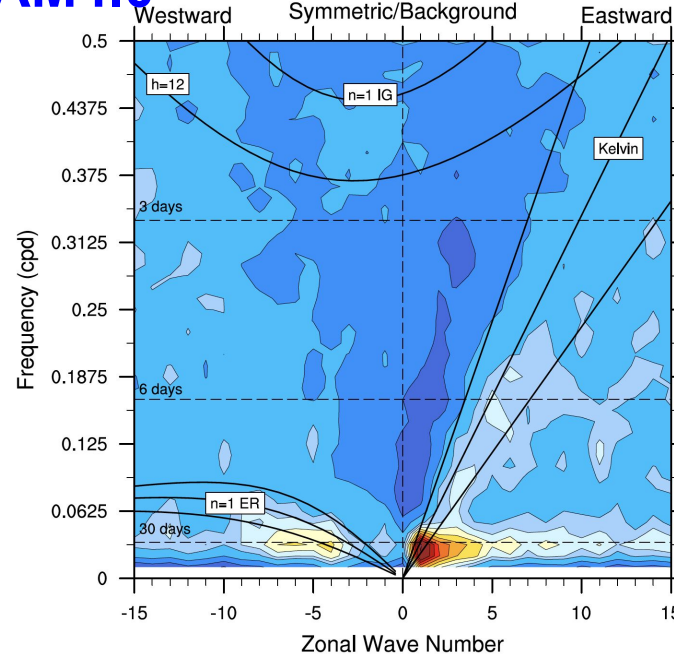


OBS

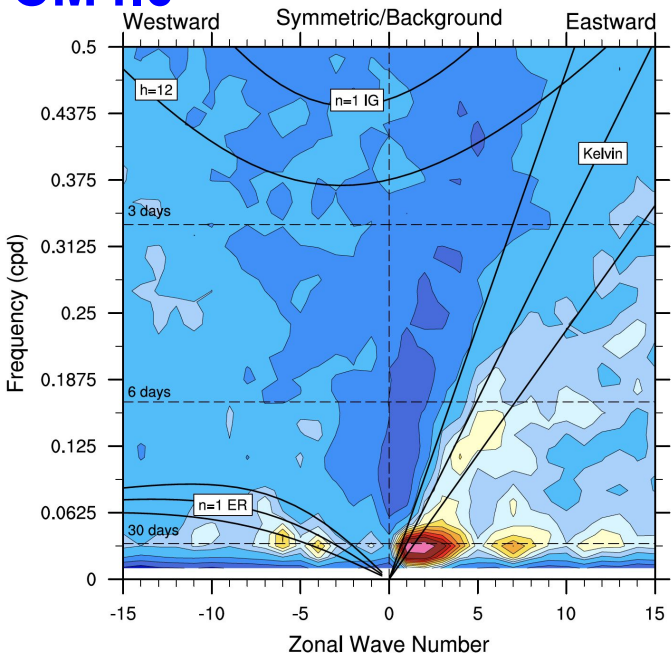
TRMM_PRECT LOG[Power: 15S-15N]

**AM4.0**

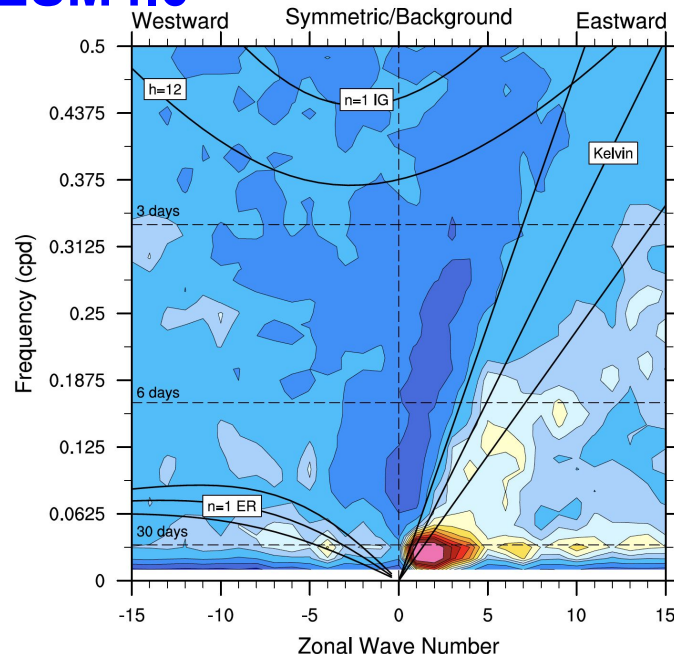
AM4.0_PRECT LOG[Power: 15S-15N]

**CM4.0**

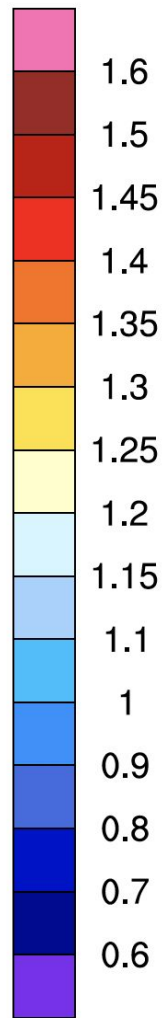
CM4.0_PRECT LOG[Power: 15S-15N]

**ESM4.0**

ESM4.0_PRECT LOG[Power: 15S-15N]



Wavenumber-Frequency Spectra



- Large difference in Kelvin waves
- Robust MJO signal

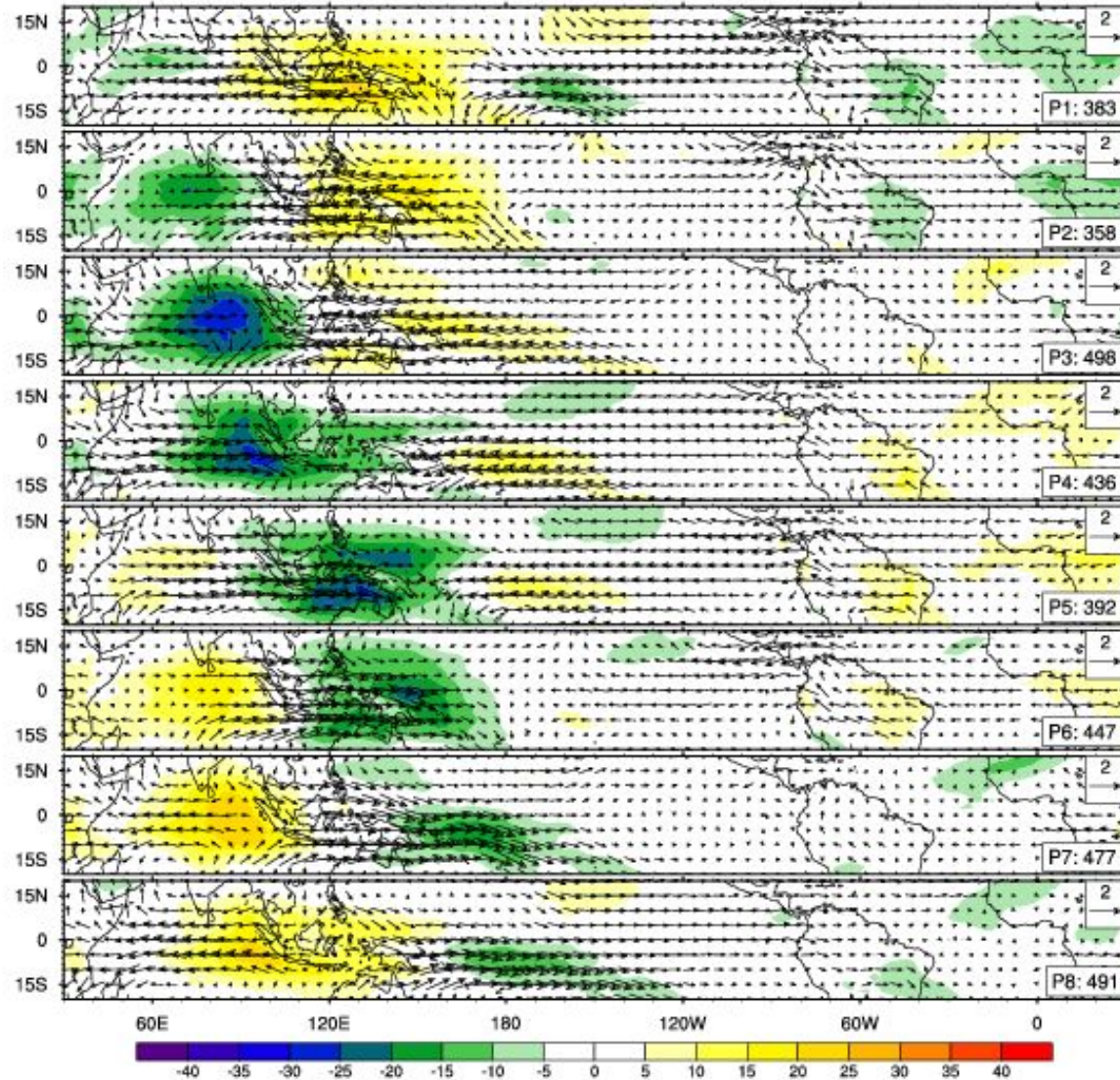
Variable(s): daily precip

Black solid lines in each plot denote theoretical equatorially trapped wave dispersion curves

MJO Spectral and Phasing

OBS

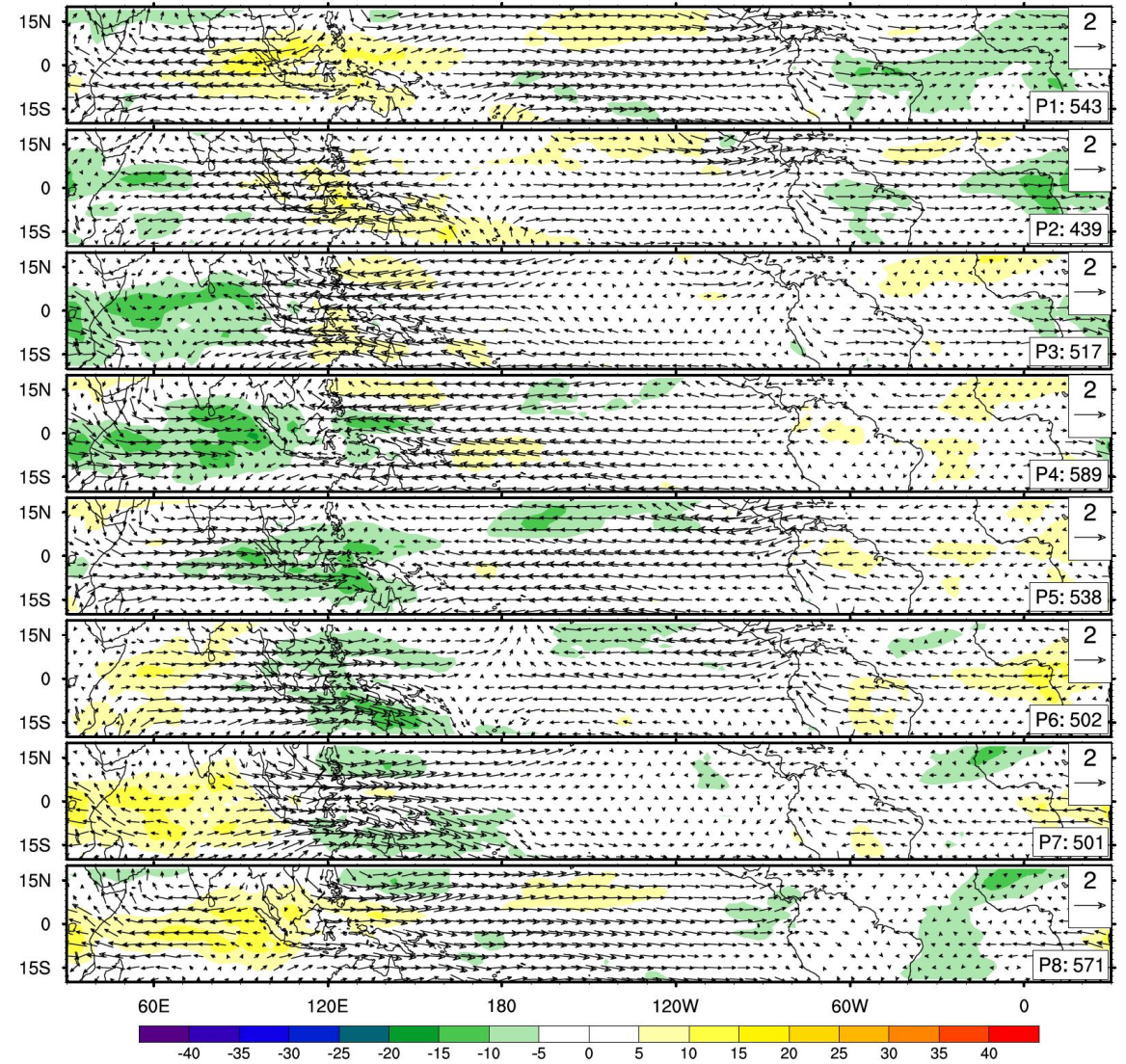
ERA during 1980-2010: Nov to Apr



Composite life cycle of the boreal winter MJO events

AM4.0

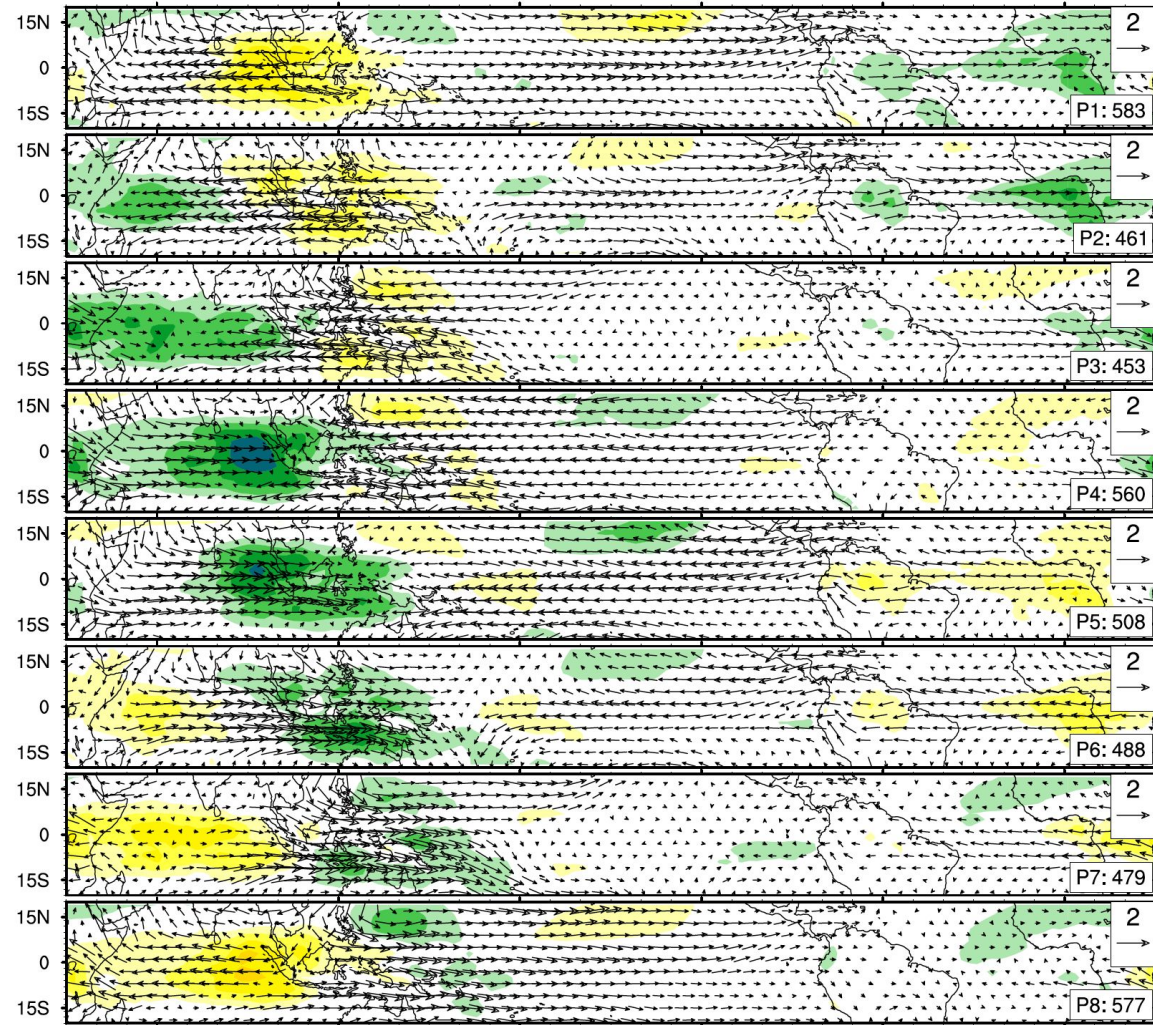
AM4.0 during 1980-2014: Nov to Apr



Variable(s): daily olr & uv(850 hPa)

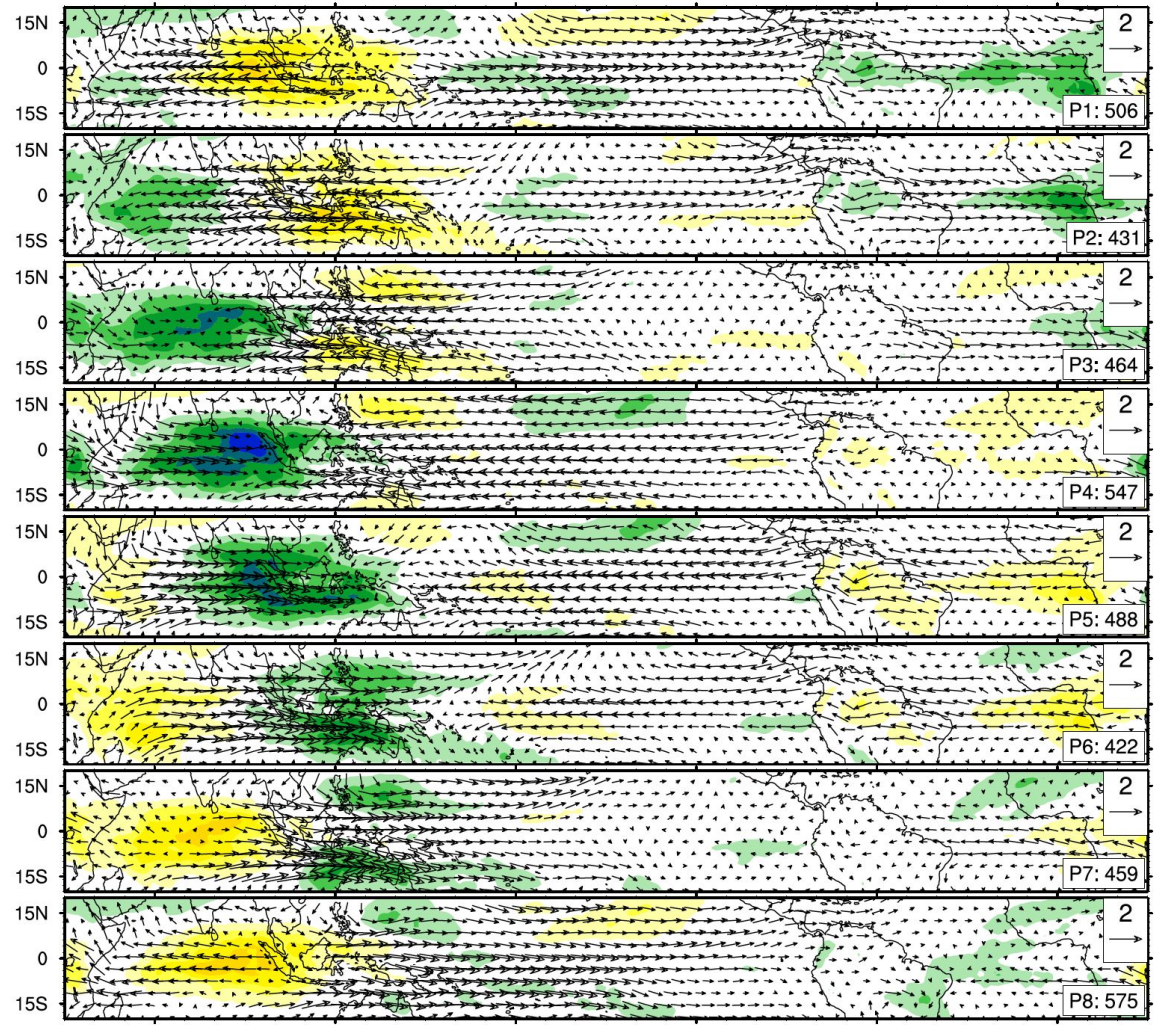
CM4.0

CM4.0 during 1980-2014: Nov to Apr



ESM4.0

ESM4.0 during 1980-2014: Nov to Apr

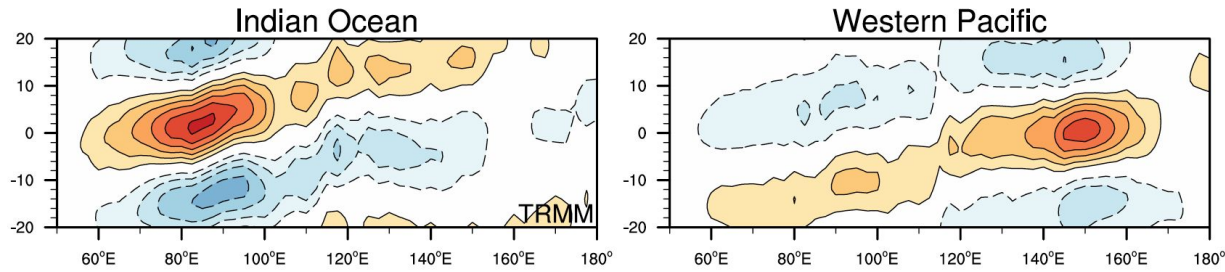


Variable(s): daily olr & uv(850 hPa)

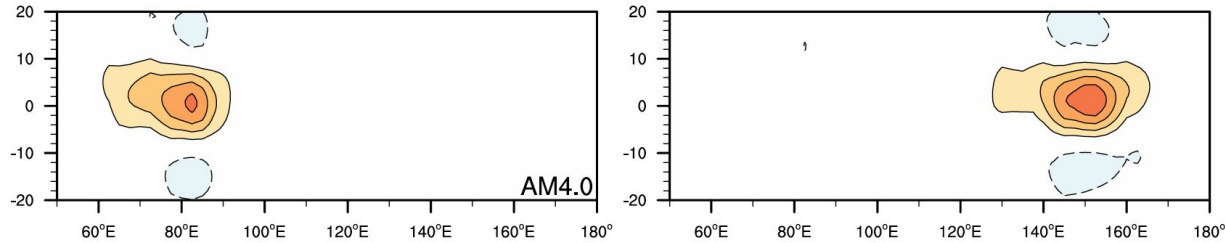
MJO Propagation and Amplitude

Longitude-time evolution of rainfall anomalies along the equator

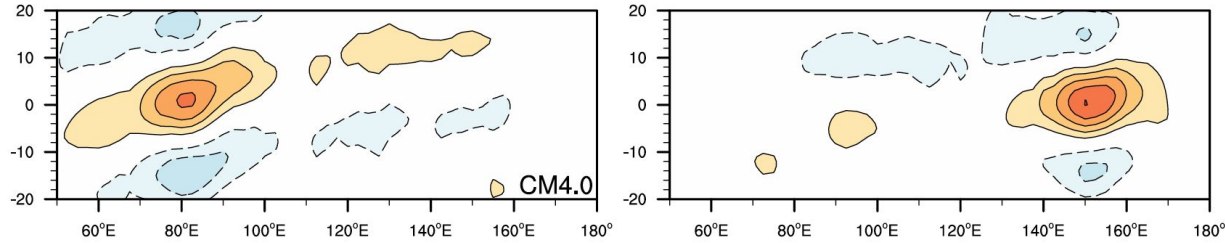
OBS



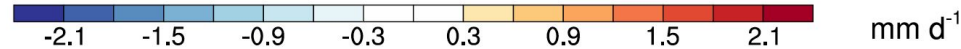
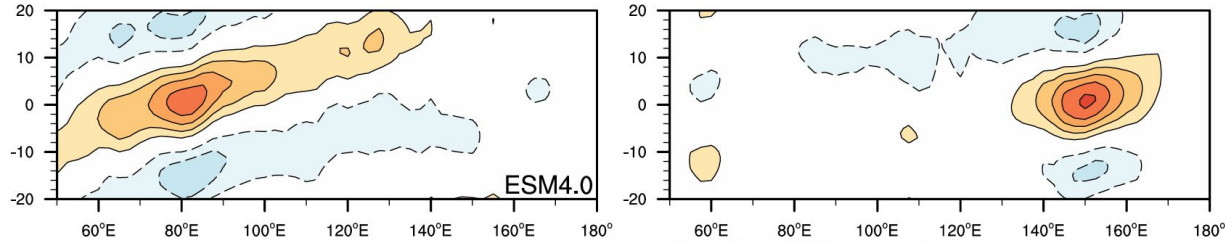
AM4.0



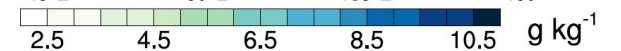
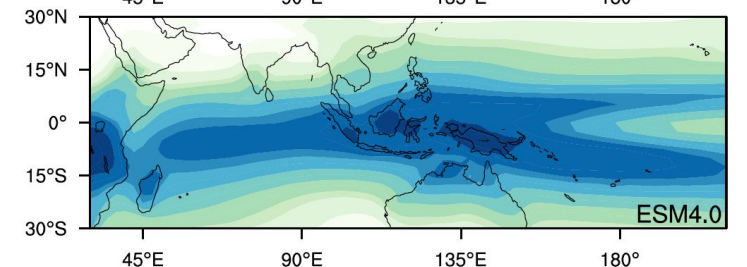
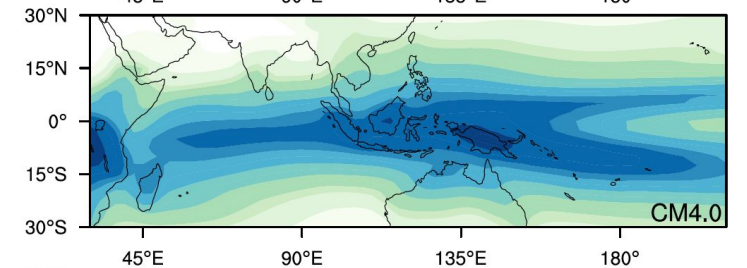
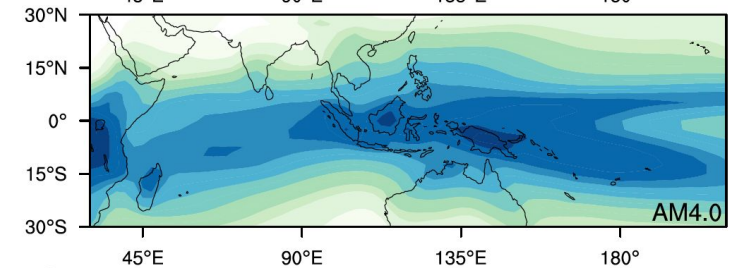
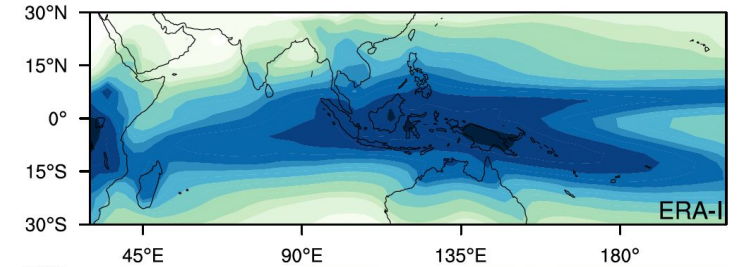
CM4.0



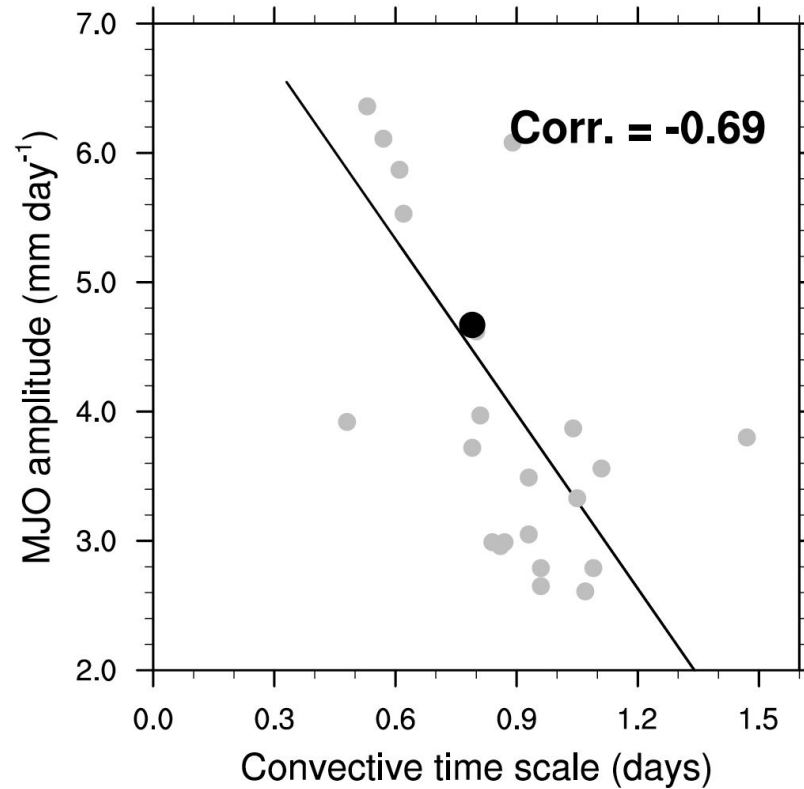
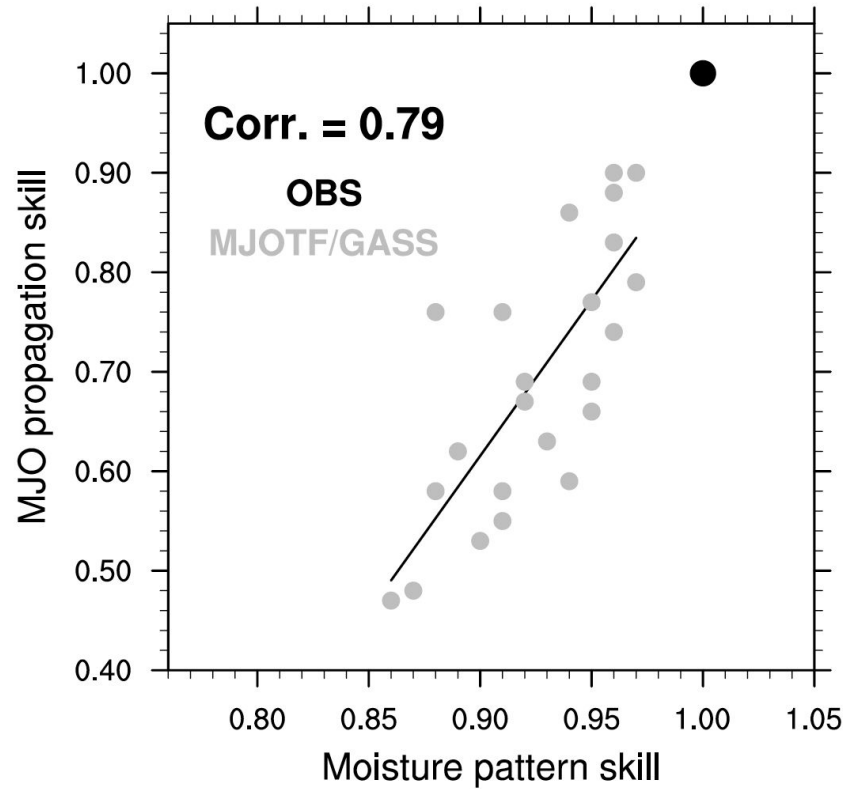
ESM4.0



900-650 hPa Winter Mean Specific Humidity



Variable(s): daily precip & specific humidity



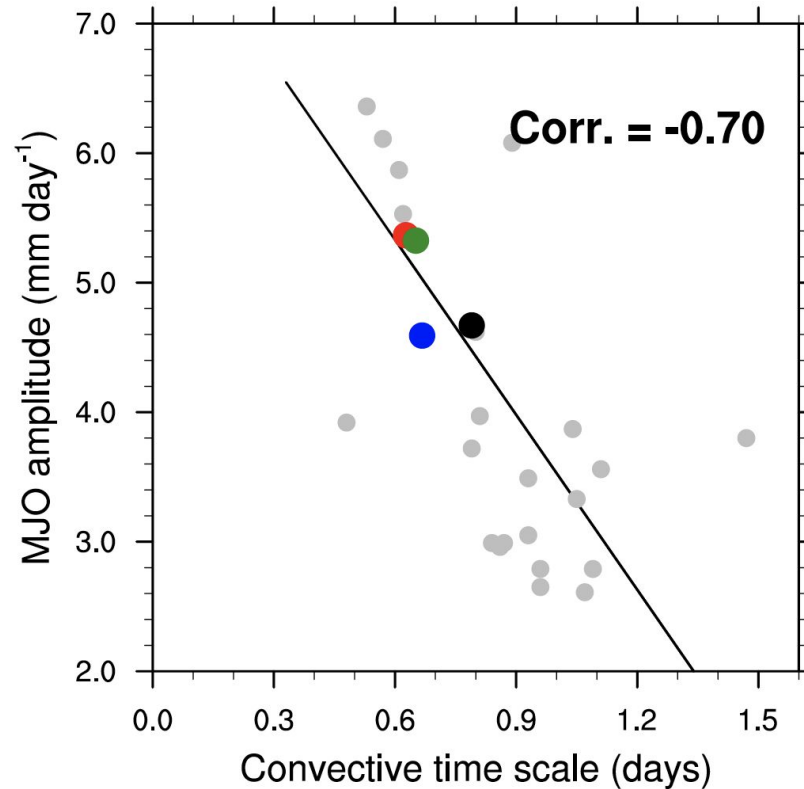
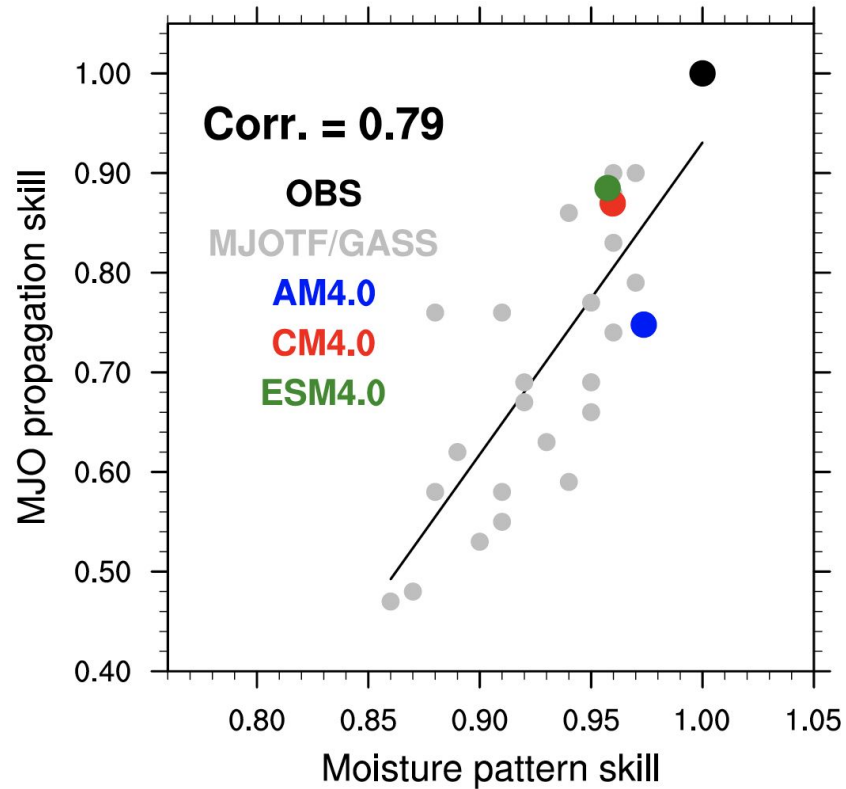
- **Better moisture
+Circulation pattern
>>Eastward propagation**
 - **Stronger MJO amplitude
>>Shorter convective time
scales**
- Variable(s): daily precip & specific humidity*

MJO propagation skill: *pattern correlation of anomalous rainfall Hovmöller diagram*

Moisture pattern skill: *pattern correlation of wintertime moisture over Maritime Continent (20°S–20°N, 90°E–135°E)*

MJO amplitude: *the standard deviations of winter intraseasonal rainfall*

Convective time scale: *the ratio of mean precipitable water anomaly to mean precipitation anomaly over Indian Ocean (5°S–5°N, 75°E–85°E) using a regression approach*



- **Better moisture
+Circulation pattern
>>Eastward propagation**
- **Stronger MJO amplitude
>>Shorter convective time
scales**

Variable(s): daily precip & specific humidity

MJO propagation skill: *pattern correlation of anomalous rainfall Hovmöller diagram*

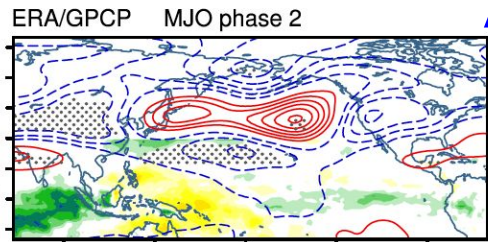
Moisture pattern skill: *pattern correlation of wintertime moisture over Maritime Continent (20°S – 20°N , 90°E – 135°E)*

MJO amplitude: *the standard deviations of winter intraseasonal rainfall*

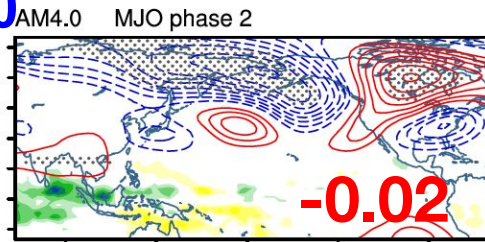
Convective time scale: *the ratio of mean precipitable water anomaly to mean precipitation anomaly over Indian Ocean (5°S – 5°N , 75°E – 85°E) using a regression approach*

Phase Composite: Anomalous geopotential height and precipitation

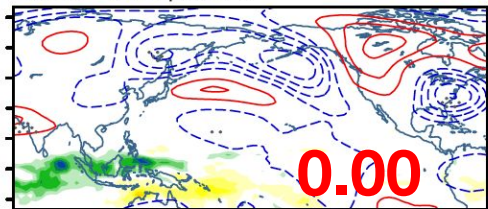
OBS



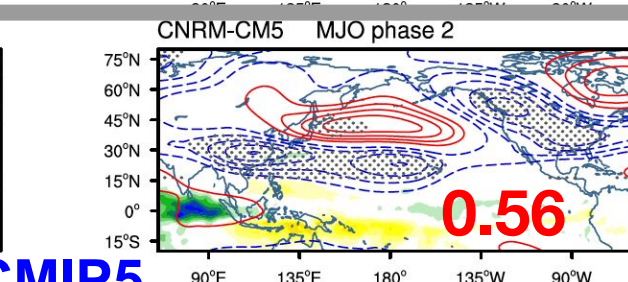
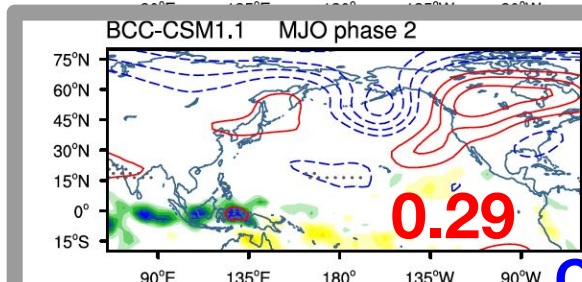
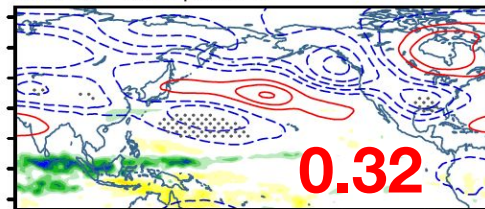
AM4.0



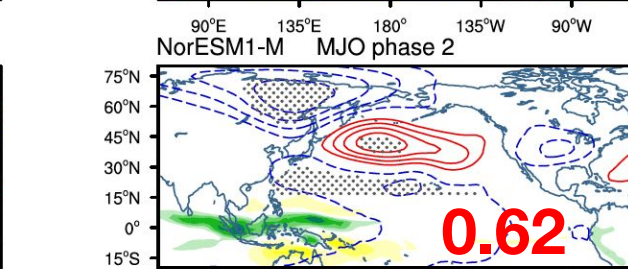
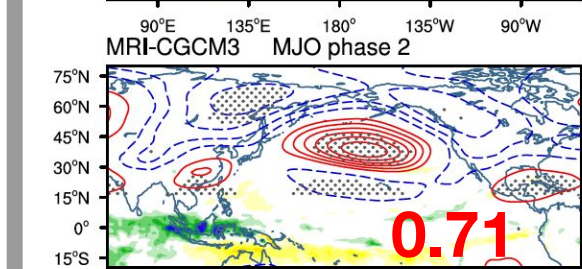
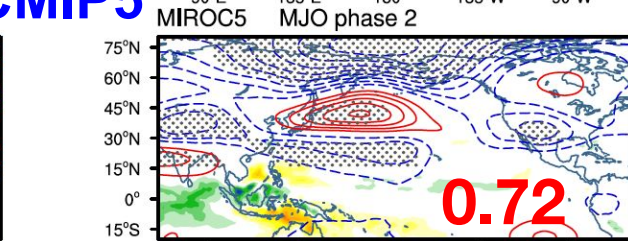
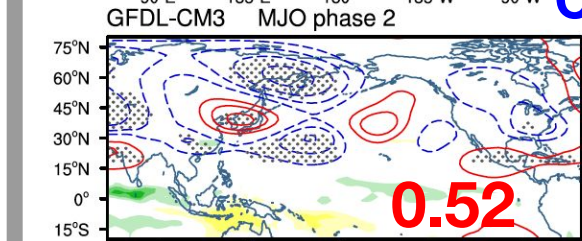
CM4.0



ESM4.0

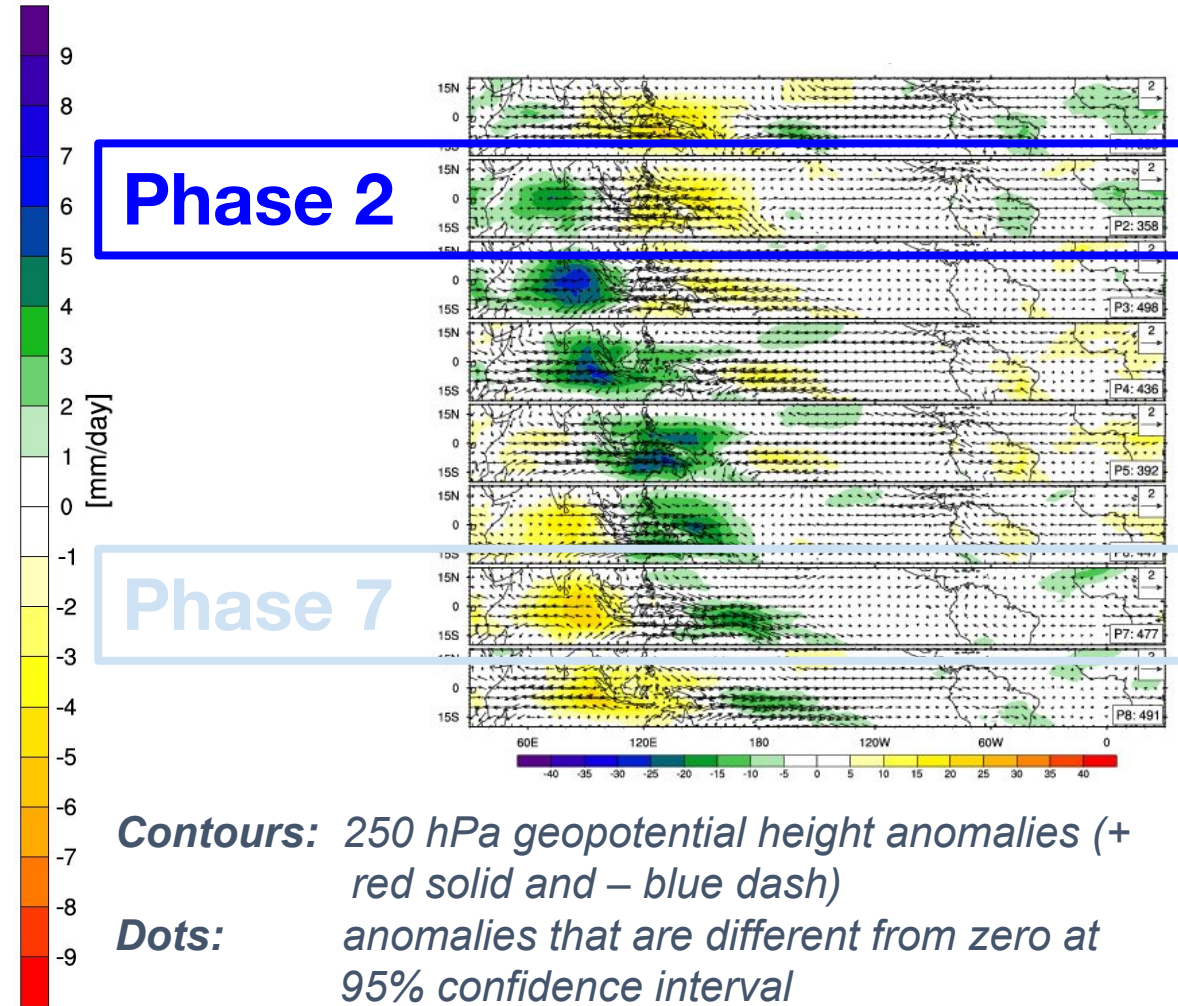


CMIP5



MJO Teleconnection

Variable(s): daily precip & geopotential height (250 hPa)



Contours: 250 hPa geopotential height anomalies (+ red solid and - blue dash)

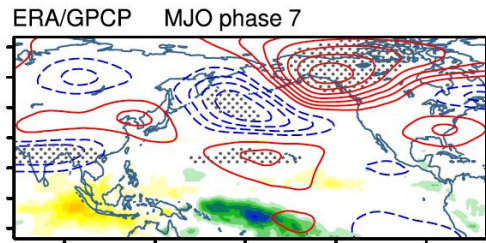
Dots: anomalies that are different from zero at 95% confidence interval

Shading: tropical precipitation anomalies

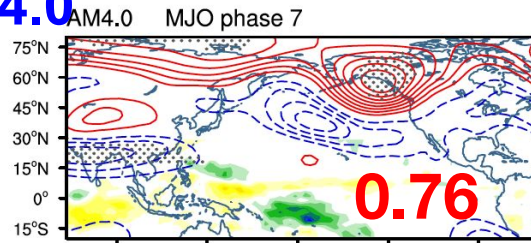
Number: pattern correlation of geopotential height anomalies

Phase Composite: Anomalous geopotential height and precipitation

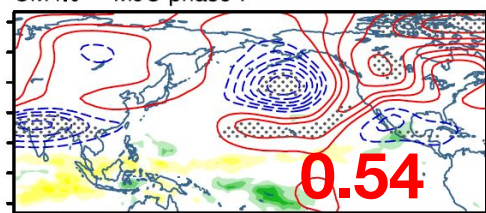
OBS



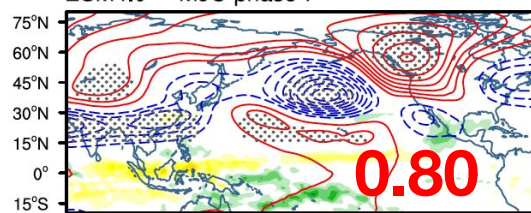
AM4.0



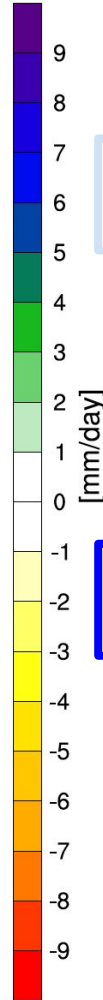
CM4.0



ESM4.0

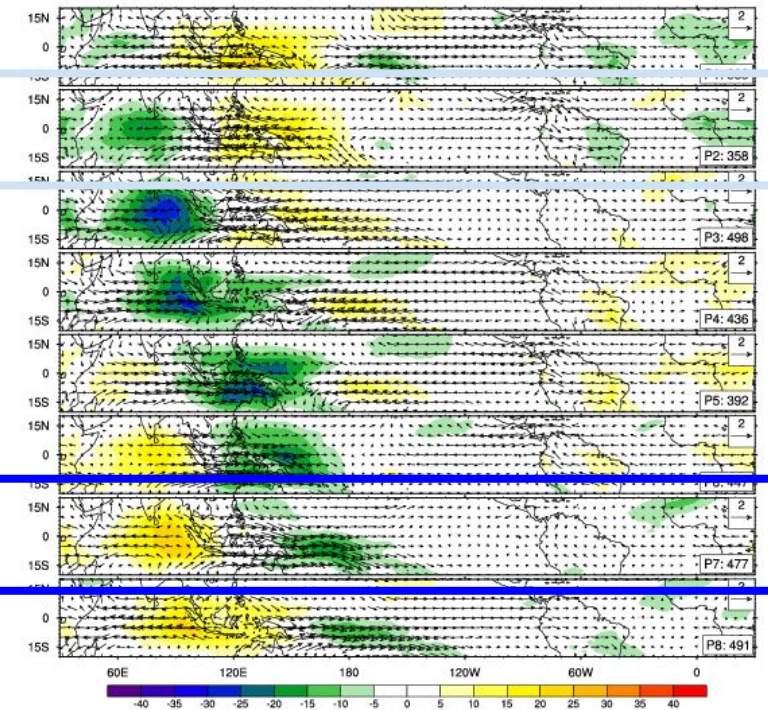


Variable(s): daily precip & geopotential height (250 hPa)



Phase 2

Phase 7



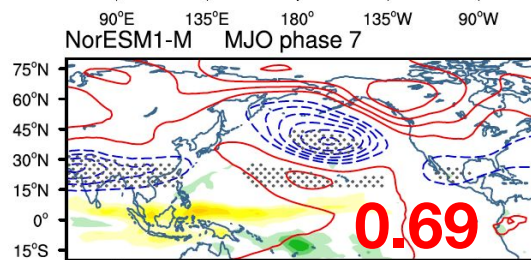
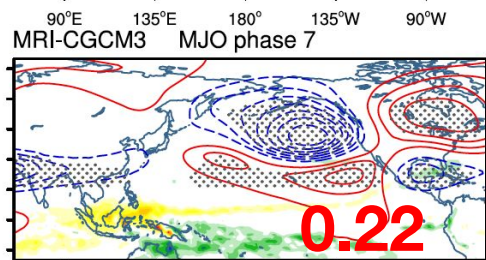
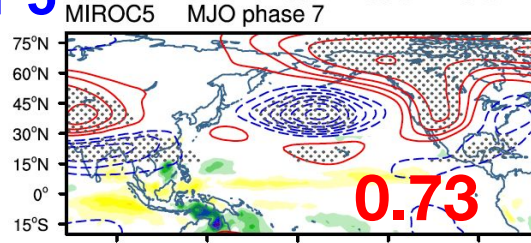
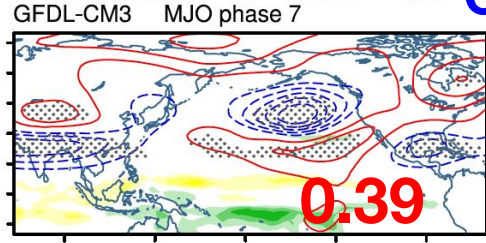
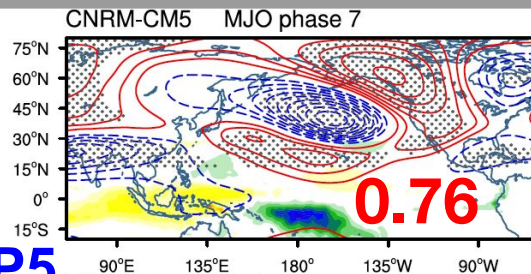
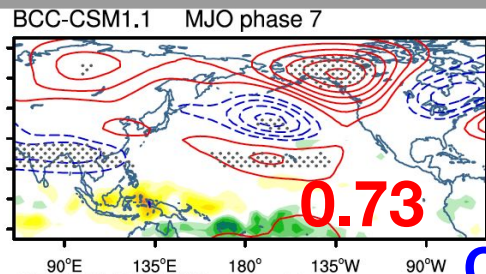
Contours: 250 hPa geopotential height anomalies (+ red solid and - blue dash)

Dots: anomalies that are different from zero at 95% confidence interval

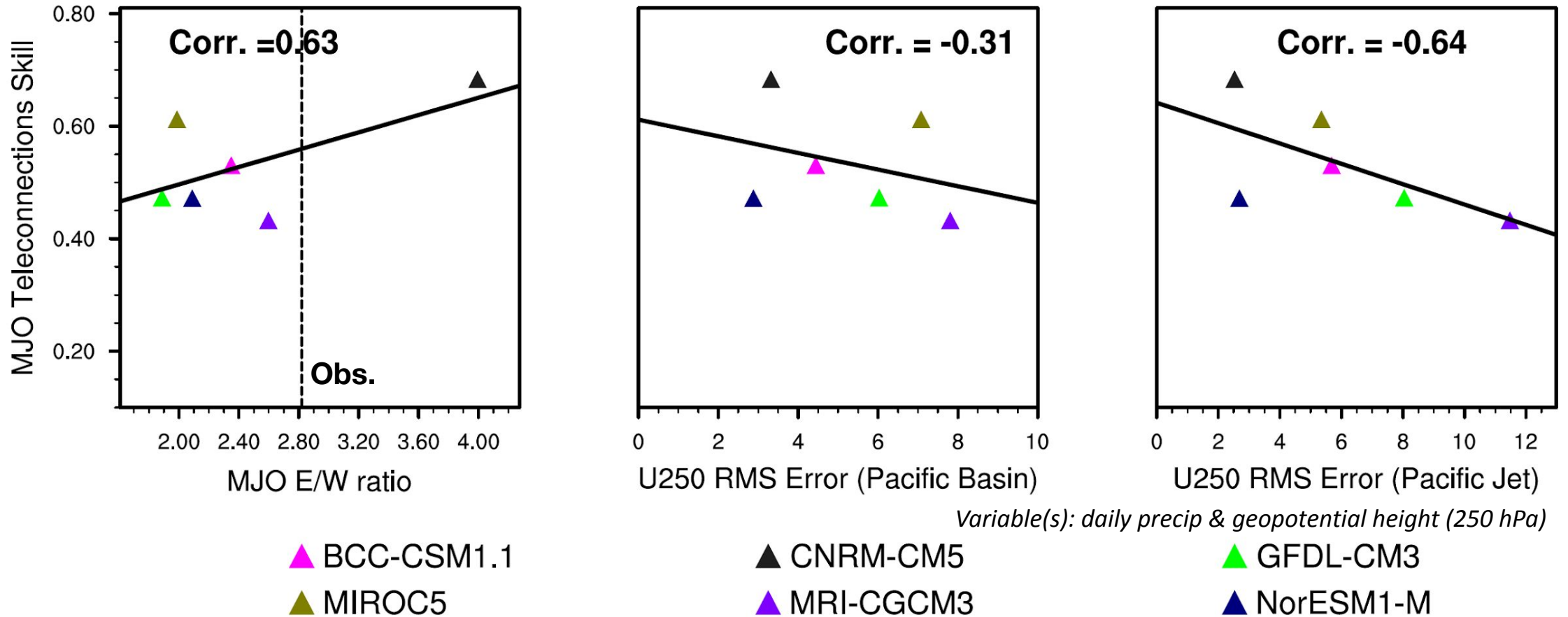
Shading: tropical precipitation anomalies

Number: pattern correlation of geopotential height anomalies

CMIP5



Teleconnection skill V.S. MJO skills and Mean State

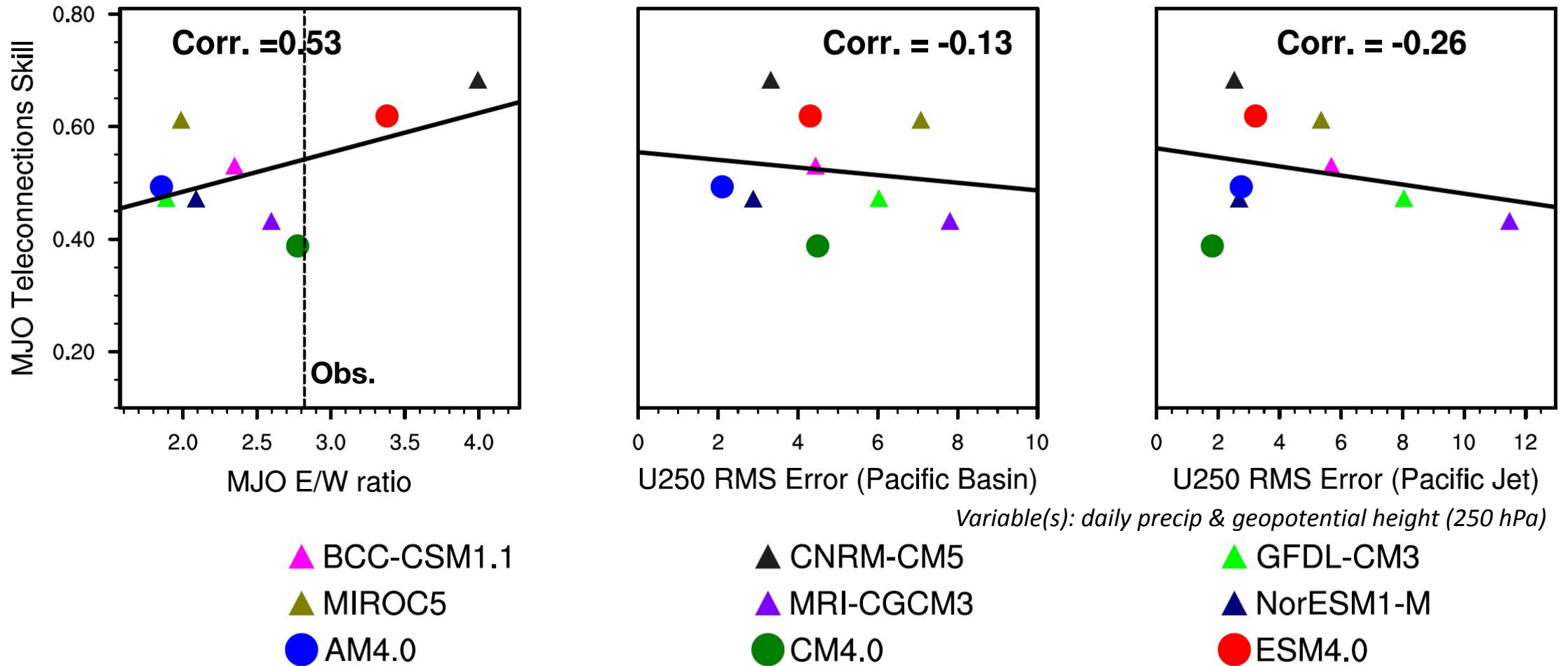


MJO Teleconnection Skill: Pattern correlation of Z250 anomalies over 15°N–80°N, 130°E–60°W

MJO E/W ratio: MJO E/W wavenumber-frequency propagation power (Zonal wave number 1-3 & period 30-60 days)

U250 RMS Error: RMS error in longitude span of full Pacific Basin (*middle*) over 15°N–60°N, or the Pacific Jet region (*right*) for the good MJO models

Teleconnection skill V.S. MJO skills and Mean State



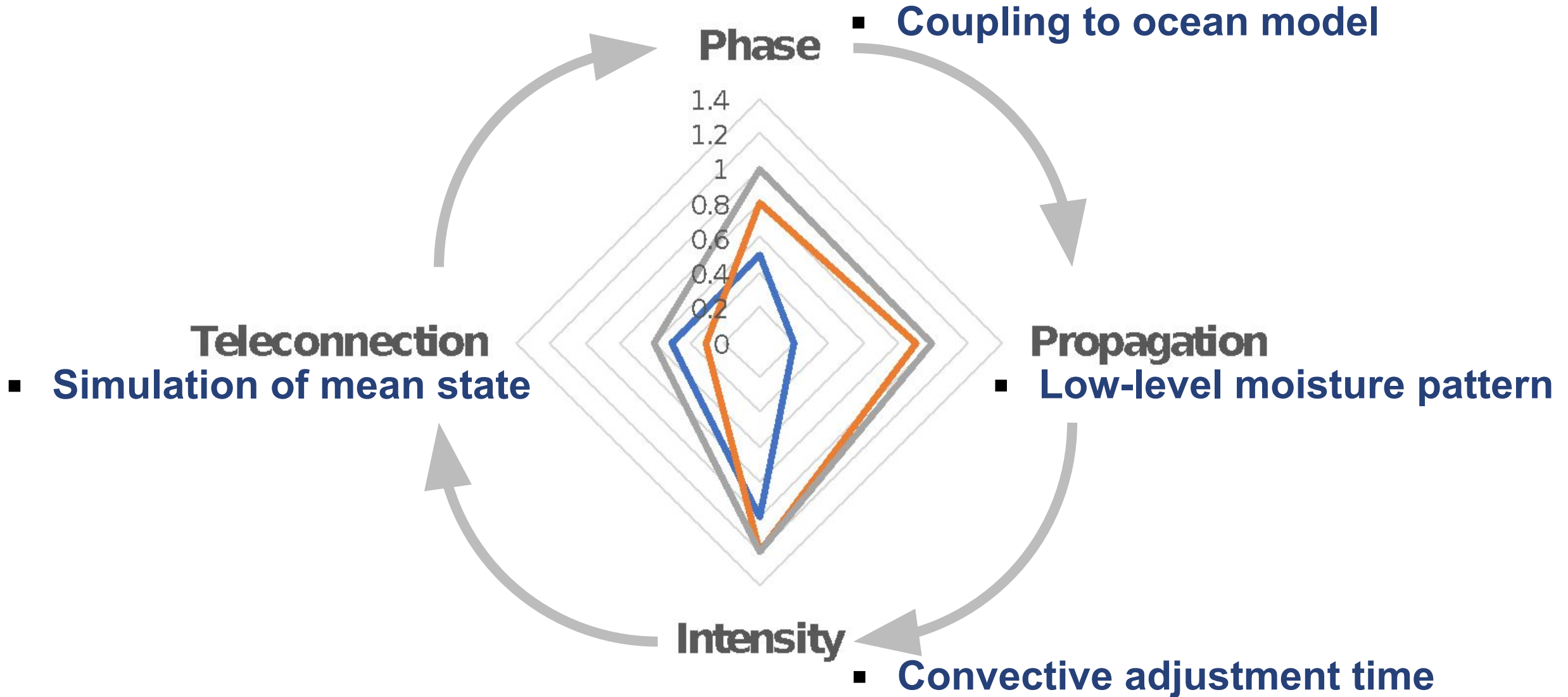
MJO Teleconnection Skill: Pattern correlation of Z250 anomalies over 15°N–80°N, 130°E–60°W

MJO E/W ratio: MJO E/W wavenumber-frequency propagation power (Zonal wave number 1-3 & period 30-60 days)

U250 RMS Error: RMS error in longitude span of full Pacific Basin (*middle*) over 15°N–60°N, or the Pacific Jet region (*right*) for the good MJO models

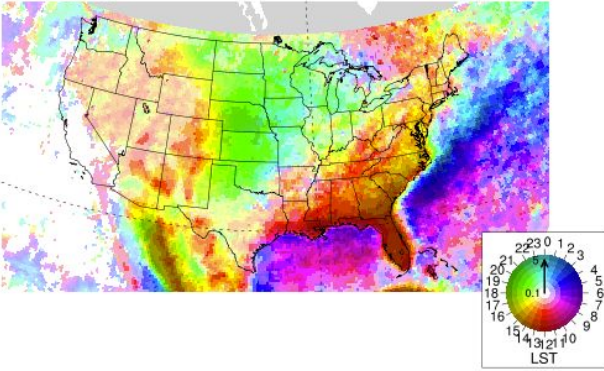
Scores of MJO simulations

—AM4.0 —CM4.0 —ESM4.0

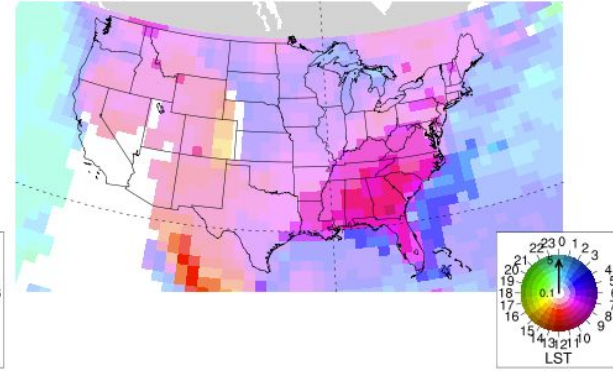


Diurnal cycle of precipitation

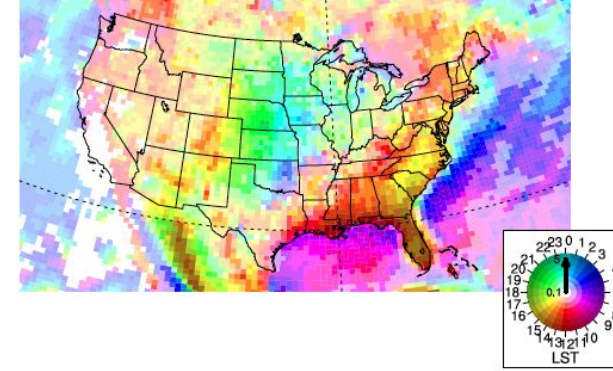
a. TRMM



b. AM4.0

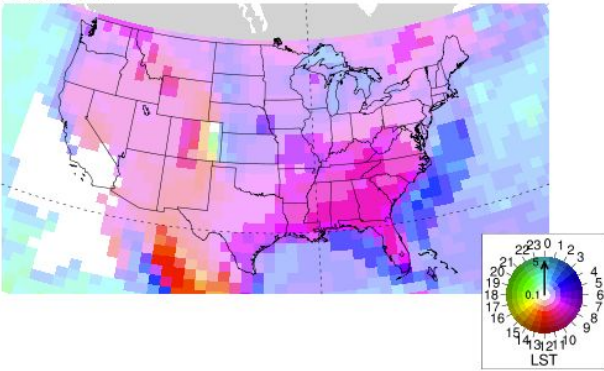


a. MSWEP

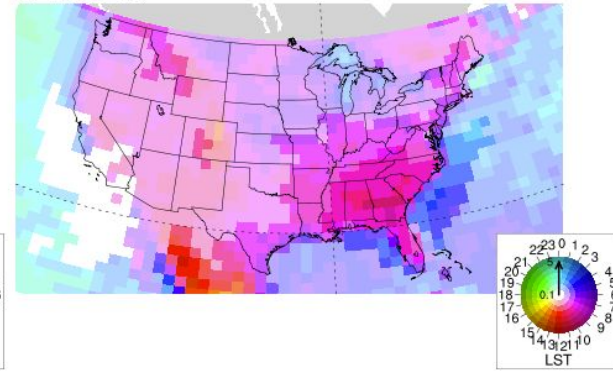


Variable(s): 3-hourly precip

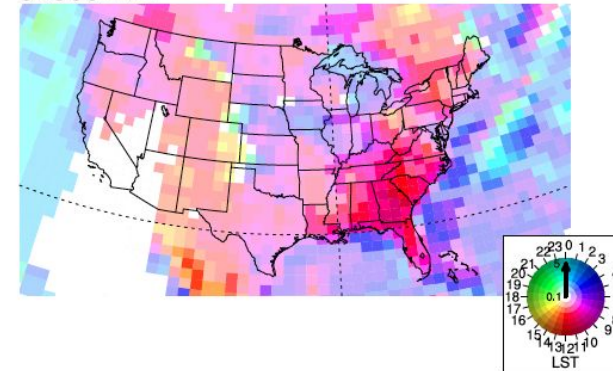
c. CM4.0



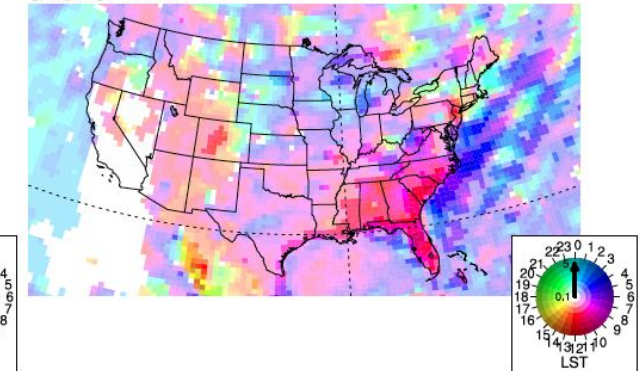
d. ESM4.0



b. C96AM4



c. C192AM4



Hue: phase of the diurnal cycle in local time
Saturation: amplitude of the diurnal cycle

Data and Method

- CL AUS (1985-2008)
3-hour $\frac{1}{3}$ ° Brightness temperature (T_b)
- C192AM4-PD (1985-2008)
3-hour ~ 50 km OLR
- Conversion from OLR to T_b

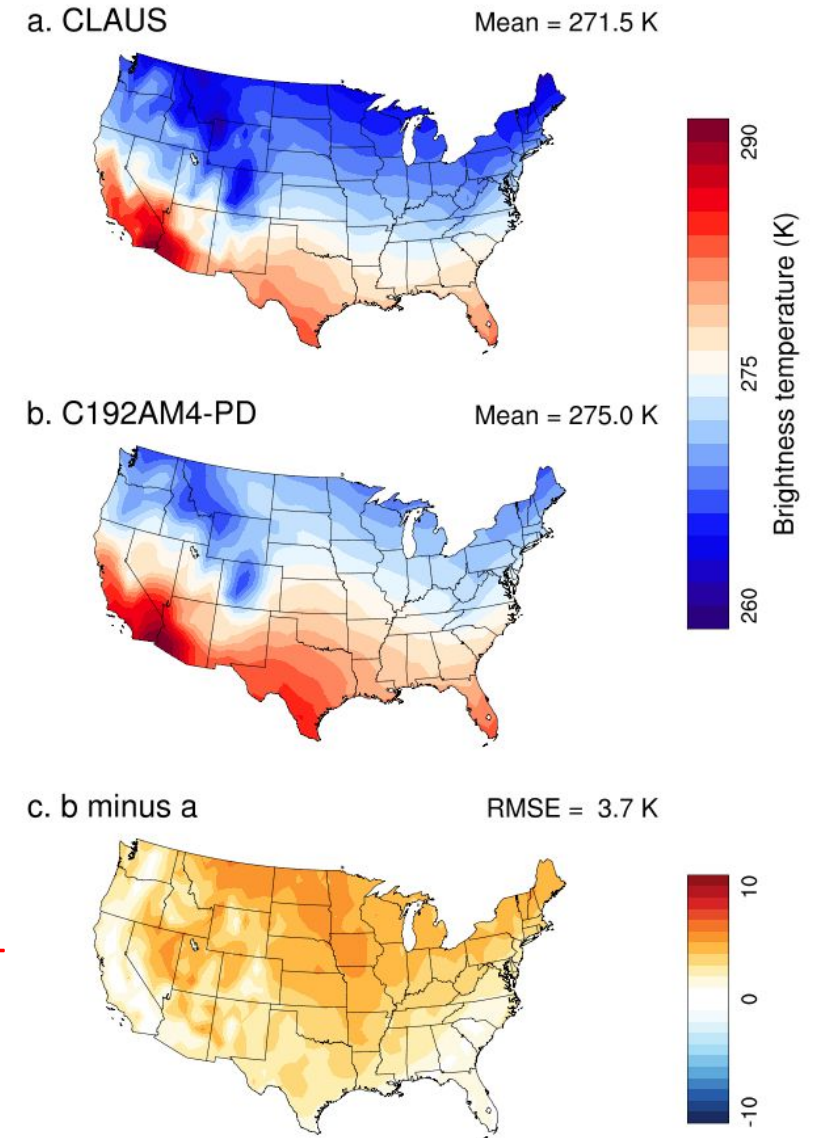
$$T_F = T_b(a+bT_b) \dots\dots (1)$$

$$OLR = \sigma T_F^4 \dots\dots (2)$$

where T_F is the flux equivalent brightness temperature, σ is the Stefan-Boltzmann constant, and a and b are empirical coefficients.

$a=1.228$ and $b= -1.106 \times 10^{-3} \text{ K}^{-1}$.

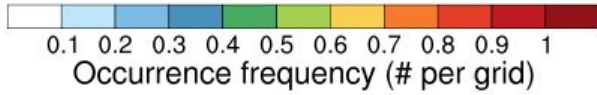
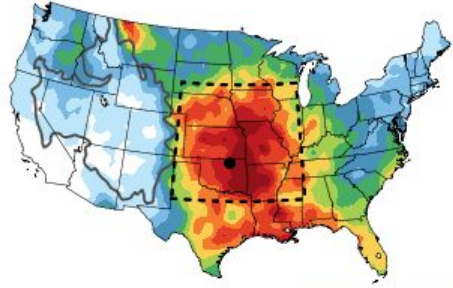
- A two-step algorithm is designed to identify MCSs based on a T_b threshold and a minimum area coverage threshold, which are set at **221** K and **30,000** km², respectively.



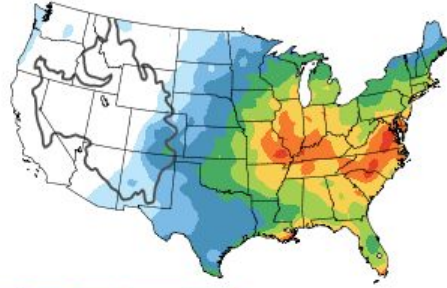
Variable(s): 3-hourly olr

Dong et al., in preparation

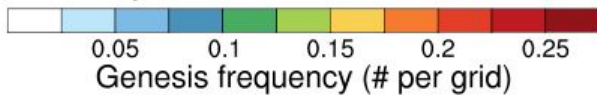
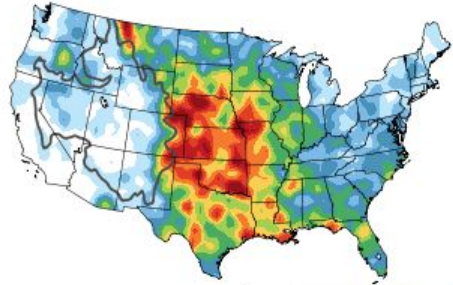
a. CLAU5



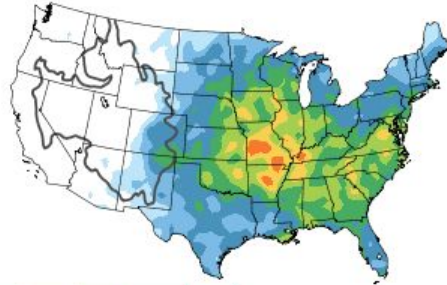
b. C192AM4-PD



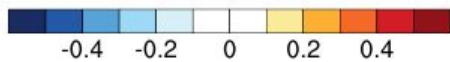
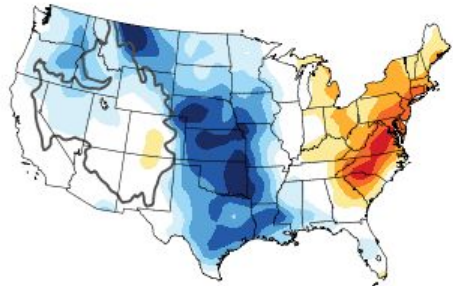
c. CLAU5



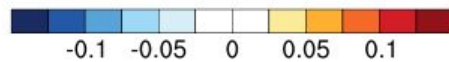
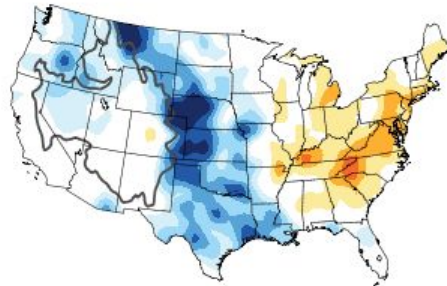
d. C192AM4-PD



e. b minus a



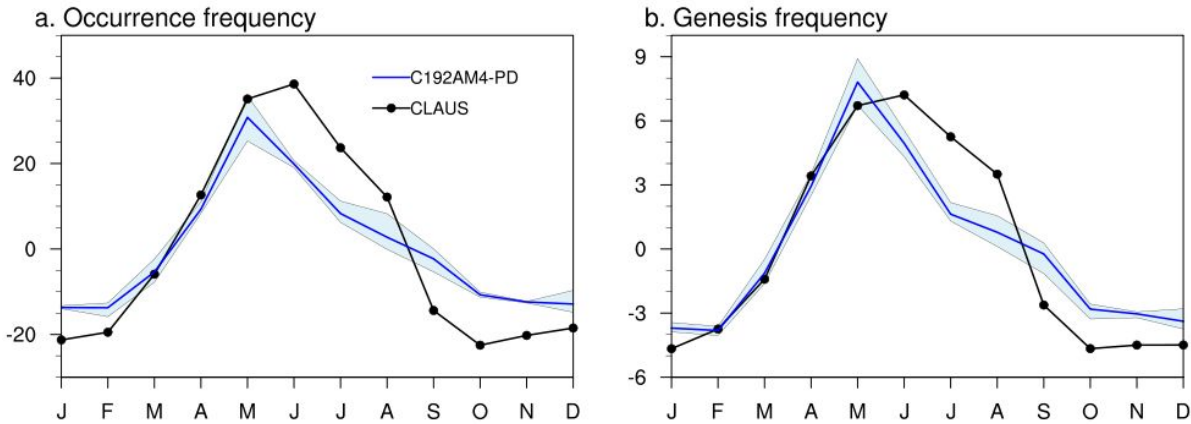
f. d minus c



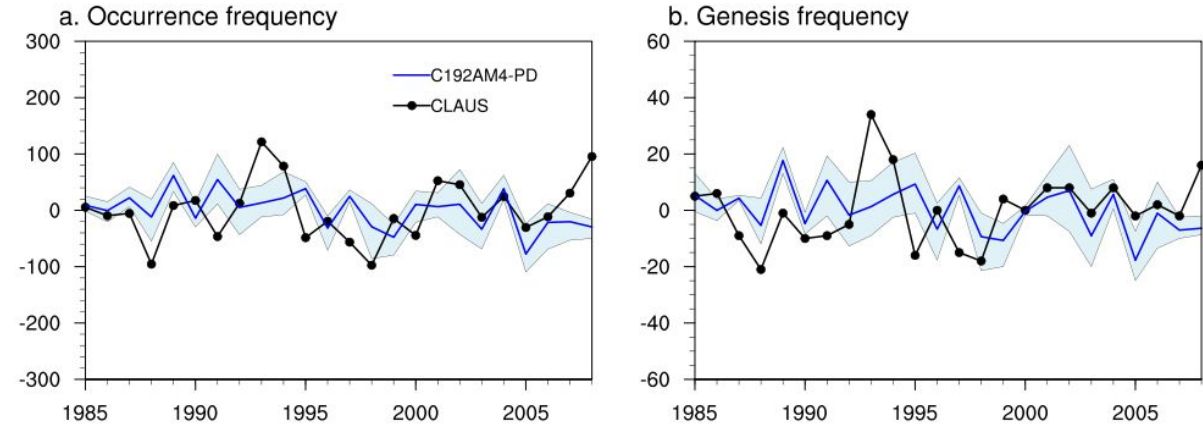
Occurrence and genesis

- Generated near the Rocky Mountain in observation
- Occurrence frequency generally follows genesis frequency
- Large bias over central US in the model

Seasonality and interannual variability



- Larger values in warm-season (Apr. - Aug.)
- Reasonable simulation of seasonal cycle

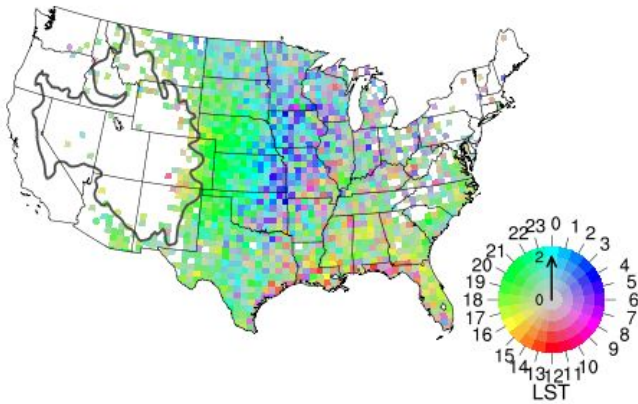


- No significant trend during 1985-2008
- The model could not capture the interannual variability

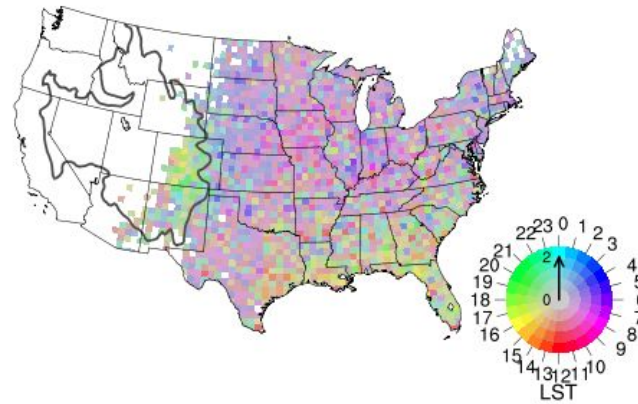
Dong et al., in preparation

Diurnal cycle

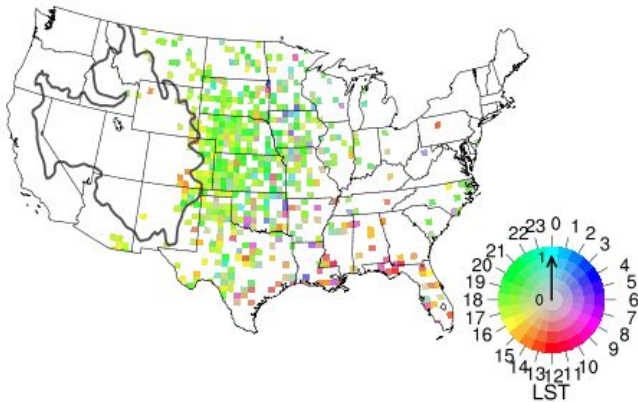
a. CLAU5 (Occurrence)



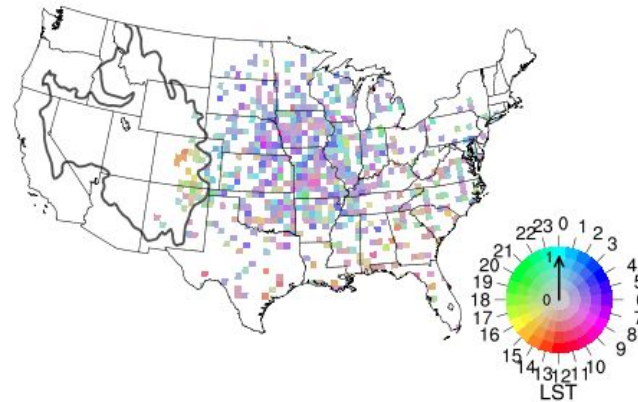
b. C192AM4-PD (Occurrence)



c. CLAU5 (Genesis)

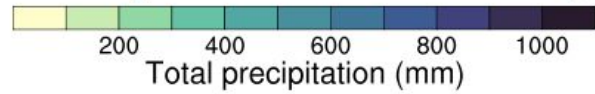
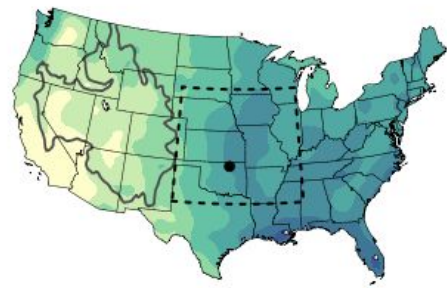


d. C192AM4-PD (Genesis)

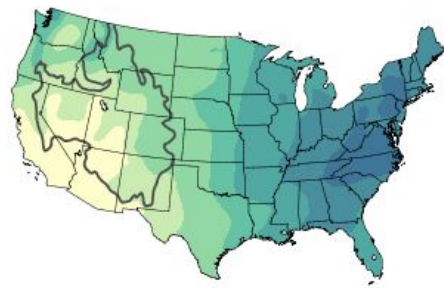


- Nocturnal peak in observation for both occurrence and genesis
- Diurnal cycle is weak in the model and the simulated peak is too early

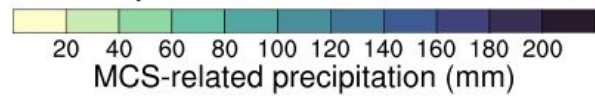
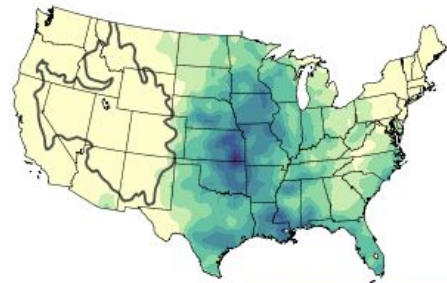
a. CLAU



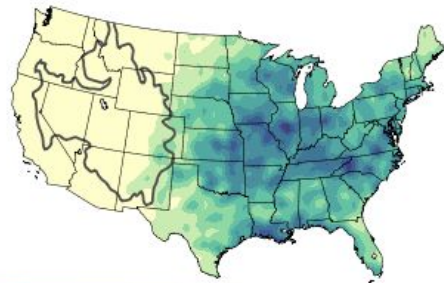
b. C192AM4-PD



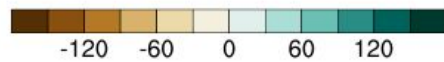
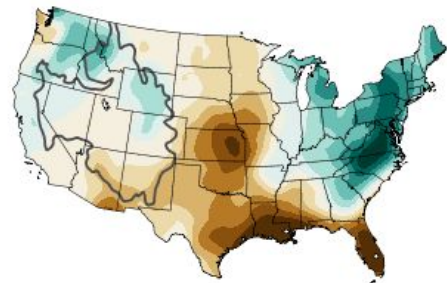
c. CLAU



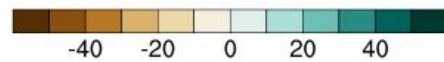
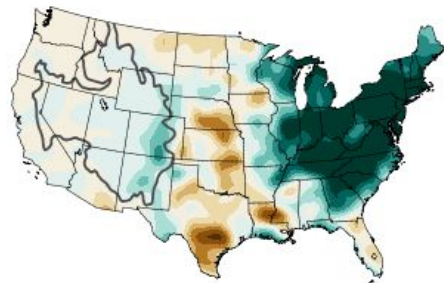
d. C192AM4-PD



e. b minus a



f. d minus c



Associated precipitation

- MCS-related precipitation contributes to ~40% of the total precipitation (**lower limit**)
- The mean dry bias could partly attributed to the MCS-related precipitation

Take home message

- MDTF-diagnostics package is a powerful tool/resource for model evaluation and model development
 - Provides and Promotes unified, open frameworks for process oriented diagnostics
 - Leverages use of CMOR/CF conventions
 - Collaborates on metrics and diagnostic standards with **CMEC**
 - Enables researchers focus more on actual scientific research
 - Write once, run everywhere
 - Strives to simplify user experience (3Cs + D-ocumentation)
 - Fosters engagement with modeling centres (NCAR, GFDL)
 - Facilitates model evaluation and model development
- Application in GFDL Seasonal prediction model (SEPAR), CM4_MG2 model, High-resolution model development, etc.
- GFDL Atmosphere Model Task Force: to advance “seamless” atmospheric modeling at GFDL building from the two major atmospheric modeling efforts in the laboratory, namely SHiELD (weather scale) and AM4 (climate focused).
 - Adopt the AM4 framework and introduce SHiELD functionality to create a seamless system
 - Test and examine short -term forecast skill

References

MDTF-DIAGNOSTICS GITHUB REPOSITORY: <https://github.com/NOAA-GFDL/MDTF-diagnostics>

mdtf-test-data package: <https://pypi.org/project/mdtf-test-data>

[Process-Oriented Evaluation of Climate and Weather Forecasting Models](#)

[From “Inspiration-driven” Research to “Industrial-strength” Research: Applying User-developed Climate Analytics at Large scale](#)

[Deploying user-developed scientific analyses on federated data archives](#)