MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Cloud Computing com Python: pacotes Zarr e Intake

Workshop Interno da DIMNT para início dos trabalhos com o MONAN-ATM/SFC de 2 a 3 de outubro de 2023.

https://github.com/monanadmin/monan_post/tree/main

Carlos Frederico Bastarz
CGCT/DIMNT/GAD
carlos.bastarz@inpe.br

MONAN
Model for Ocean-laNd-Atmosphere predictioN

MINISTÉRIO DA CIÊNCIA,TECNOLOGIA E INOVAÇÃO
GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

0 Introdução

- Além da **interpreação física** das previsões numéricas (e.g., da atmosfera) em ponto de grade e níveis de pressão, o pós-processamento envolve também o **acesso** aos dados. Sejam eles armazenados nos formatos **GRIB**, **netCDF**, **binários** etc, o seu acesso é frequentemente feito **localmente**, i.e., na máquina em que foram produzidos.

- Há vários **protocolos** que permitem o **acesso remoto** destes dados, como o **OPeNDAP**, mas estes requerem a configuração de um **servidor** para esta finalidade.

- O **Python** possui **bibliotecas** que permitem a construção de **catálogo de dados** os quais permitem o seu acesso **remoto** a partir de um ou mais **fontes de dados**. Além disso, dados muito **grandes** podem também ser convertidos para outros formatos com a finalidade de **comprimir** as informações e **acelerar** o seu acesso na **nuvem**.

- O **Zarr** e o **Intake** são duas bibliotecas que podem ser utilizadas para estas finalidades: **fragmentar** e **comprimir** e **acessar remotamente** os dados por meio de um **catálogo**!

- Nesta apresentação é fornecido um exemplo muito simples, utilizando dados provenientes do **SCANTEC**, mas as mesmas ideias podem ser aplicadas aos dados do **MONAN**.

1 Biblioteca Zarr

Using Cloud Computing to Analyze Model Output Archived in Zarr Format

TAYLOR A. GOWAN,[a] JOHN D. HOREL,[a] ALEXANDER A. JACQUES,[a] AND ADAIR KOVAC[a]

[a] *Department of Atmospheric Sciences, University of Utah, Salt Lake City, Utah*

(Manuscript received 28 July 2021, in final form 2 January 2022)

ABSTRACT: Numerical weather prediction centers rely on the Gridded Binary Second Edition (GRIB2) file format to efficiently compress and disseminate model output as two-dimensional grids. User processing time and storage requirements are high if many GRIB2 files with size O(100 MB, where B = bytes) need to be accessed routinely. We illustrate one approach to overcome such bottlenecks by reformatting GRIB2 model output from the High-Resolution Rapid Refresh (HRRR) model of the National Centers for Environmental Prediction to a cloud-optimized storage type, Zarr. Archives of the original HRRR GRIB2 files and the resulting Zarr stores on Amazon Web Services (AWS) Simple Storage Service (S3) are available publicly through the Amazon Sustainability Data Initiative. Every hour, the HRRR model produces 18- or 48-hourly GRIB2 surface forecast files of size O(100 MB). To simplify access to the grids in the surface files, we reorganize the HRRR model output for each variable and vertical level into Zarr stores of size O(1 MB), with chunks O(10 kB) containing all forecast lead times for 150 × 150 gridpoint subdomains. Open-source libraries provide efficient access to the compressed Zarr stores using cloud or local computing resources. The HRRR-Zarr approach is illustrated for common applications of sensible weather parameters, including real-time alerts for high-impact situations and retrospective access to output from hundreds to thousands of model runs. For example, time series of surface pressure forecast grids can be accessed using AWS cloud computing resources approximately 40 times as fast from the HRRR-Zarr store as from the HRRR-GRIB2 archive.

SIGNIFICANCE STATEMENT: The rapid evolution of computing power and data storage have enabled numerical weather prediction forecasts to be generated faster and with more detail than ever before. The increased temporal and spatial resolution of forecast model output can force end users with finite memory and storage capabilities to make pragmatic decisions about which data to retrieve, archive, and process for their applications. We illustrate an approach to alleviate this access bottleneck for common weather analysis and forecasting applications by using the Amazon Web Services (AWS) Simple Storage Service (S3) to store output from the High-Resolution Rapid Refresh (HRRR) model in Zarr format. Zarr is a relatively new data storage format that is flexible, compressible, and designed to be accessed with open-source software either using cloud or local computing resources. The HRRR-Zarr dataset is publicly available as part of the AWS Sustainability Data Initiative.

Gowan, T. A., J. D. Horel, A. A. Jacques, and A. Kovac, 2022: Using Cloud Computing to Analyze Model Output Archived in Zarr Format. J. Atmos. Oceanic Technol., 39, 449–462, https://doi.org/10.1175/JTECH-D-21-0106.1.

Zarr-Python

Version: 2.16.1

Download documentation: Zipped HTML

Useful links: Installation | Source Repository | Issue Tracker | Gitter

Zarr

Zarr is a file storage format for chunked, compressed, N-dimensional arrays based on an open-source specification.

https://zarr.readthedocs.io/en/stable/

MONAN Model for Ocean-laNd-Atmosphere predictioN

INPE

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

2 Biblioteca Intake

O que é o Intake?

- Biblioteca do Python para distribuição e acesso de dados.
- Fornece uma coleção de drivers para leitura de direfentes tipos de dados;
- Os dados são distribuídos por meio de catálogos;
- Catálogos são coleções de dados!

https://intake.readthedocs.io/en/latest/

3 Um exemplo
Preparação dos Dados: Zarr

Como preparar dados binários para o formato Zarr?

Open in Colab

1. Importar as bibliotecas necessárias;
2. Obter e organizar os dados;
3. Definir um compressor para ser aplicado às variáveis de interesse.

1

```
import zarr
import xarray as xr

Exps = ['BAMH']
Stats = ['VIES', 'RMSE']

data = '20230216002023030300'

burl = 'https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data'
```

2

```
!wget https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/VIESBAMH_20230216002023030300F.ctl
!wget https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/VIESBAMH_20230216002023030300F.scan
```

MONAN Model for Ocean-laNd-Atmosphere predictioN

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

3 Um exemplo

Preparação dos Dados: Zarr

```
compressor = zarr.Blosc(cname="zstd", clevel=3, shuffle=2)
encoding = {
    'zgeo250': {"compressor": compressor},
    'pslc000': {"compressor": compressor},
    'uvel500': {"compressor": compressor},
    'umes500': {"compressor": compressor},
    'zgeo500': {"compressor": compressor},
    'vvel500': {"compressor": compressor},
    'vtmp850': {"compressor": compressor},
    'uvel850': {"compressor": compressor},
    'vvel850': {"compressor": compressor},
    'temp850': {"compressor": compressor},
    'temp250': {"compressor": compressor},
    'umes925': {"compressor": compressor},
    'vtmp925': {"compressor": compressor},
    'temp500': {"compressor": compressor},
    'vtmp500': {"compressor": compressor},
    'umes850': {"compressor": compressor},
    'uvel250': {"compressor": compressor},
    'vvel250': {"compressor": compressor},
    'zgeo850': {"compressor": compressor},
}

for exp in Exps:
    for stat in Stats:
        print(exp, stat)
        filein = str(stat) + str(exp) + '_' + str(data) + 'F.ctl'
        fnamein = os.path.join(filein)
        fnameout = str(stat) + str(exp) + '_' + str(data) + 'F.zarr'
        dset = open_CtlDataset(fnamein).to_zarr(fnameout, mode='w', consolidated=True, encoding=encoding)
```

3

Open in Colab

Como preparar dados binários para o formato Zarr?

1. Importar as bibliotecas necessárias;
2. Obter e organizar os dados;
3. Definir um compressor para ser aplicado às variáveis de interesse.

MONAN Model for Ocean-laNd-Atmosphere predictioN

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

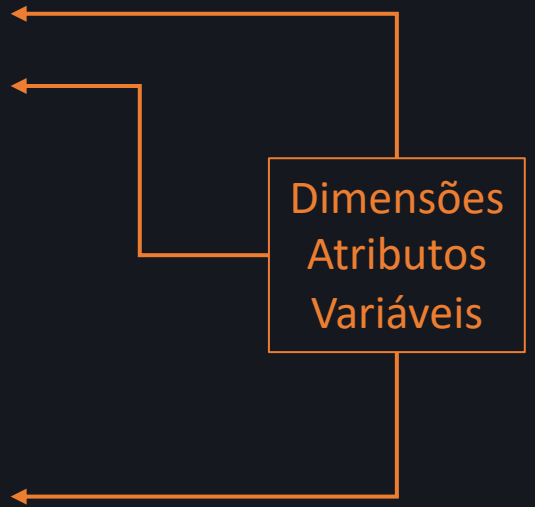
3 Um exemplo
Preparação dos Dados: Zarr

Como os dados no formato Zarr são organizados?

Open in Colab

- As informações referentes às **dimensões** e **variáveis** são fragmentadas em **pequenos arquivos indexados**!

```
(base) 👽 [/pesq/share/das/dist/carlos.bastarz/SCANTEC-2.1.0/dataout/periodo/T1/gl/RMSEDTC_2023021600202303030OF.zarr]
carlos.bastarz@headnode 14:07 $ ls -lrta
total 120
-rw-r--r--  1 carlos.bastarz ensemble    24 May 30 08:55 .zgroup
-rw-r--r--  1 carlos.bastarz ensemble   133 May 30 08:55 .zattrs
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 time
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 lon
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 lat
-rw-r--r--  1 carlos.bastarz ensemble 17900 May 30 08:55 .zmetadata
drwxr-sr-x 24 carlos.bastarz ensemble  4096 May 30 08:55 .
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 vvel250
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 vtmp500
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 vtmp850
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 vvel850
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 umes925
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 uvel250
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 zgeo850
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 vvel500
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 temp500
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 vtmp925
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 uvel500
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 zgeo500
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 temp850
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 umes850
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 umes500
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 temp250
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 zgeo250
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 pslc000
drwxr-sr-x  2 carlos.bastarz ensemble  4096 May 30 08:55 uvel850
drwxrwsr-x 23 carlos.bastarz ensemble  4096 Jun  1 15:09 ..
```



Arquivo de índice:

- **zmetadata**: contém todas as informações sobre o armazenamento das dimensões e variáveis;
- Toda a estrutura e arquivos de índice são criados pelo Zarr quando os arquivos são salvos neste formato!

MONAN Model for Ocean-laNd-Atmosphere predictioN

INPE

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

3 Um exemplo
Preparação dos Dados: Zarr

Como os dados no formato Zarr são organizados?

Open in Colab

- As informações referentes às **dimensões** e **variáveis** são fragmentadas em **pequenos arquivos indexados**!

```
(base) 👽 [/pesq/share/das/dist/carlos.bastarz/SCANTEC-2.1.0/dataout/periodo/T1/gl/RMSEDTC_2023021600202303030OF.zarr/temp850]
carlos.bastarz@headnode 14:16 $ ls -ltra
total 13216
-rw-r--r--  1 carlos.bastarz ensemble     155 May 30 08:55 .zattrs
-rw-r--r--  1 carlos.bastarz ensemble     366 May 30 08:55 .zarray
drwxr-sr-x 24 carlos.bastarz ensemble    4096 May 30 08:55 ..
-rw-r--r--  1 carlos.bastarz ensemble 1108611 May 30 08:55 9.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1109827 May 30 08:55 8.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1107574 May 30 08:55 7.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1110423 May 30 08:55 6.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1111353 May 30 08:55 5.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1115643 May 30 08:55 4.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1124129 May 30 08:55 3.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1134198 May 30 08:55 2.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1108913 May 30 08:55 11.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1107272 May 30 08:55 10.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1132442 May 30 08:55 1.0.0
-rw-r--r--  1 carlos.bastarz ensemble 1125826 May 30 08:55 0.0.0
drwxr-sr-x  2 carlos.bastarz ensemble    4096 May 30 08:55 .
```

Pedaços (chuncks) de informação!

```
carlos.bastarz@headnode 14:15 $ more .zattrs
{
    "_ARRAY_DIMENSIONS": [
        "time",
        "lat",
        "lon"
    ],
    "comment": "absolute temperature @ 850 hpa [k]",
    "storage": "99"
}
```

```
carlos.bastarz@headnode 14:15 $ more .zarray
{
    "chunks": [
        1,
        401,
        901
    ],
    "compressor": {
        "blocksize": 0,
        "clevel": 5,
        "cname": "lz4",
        "id": "blosc",
        "shuffle": 1
    },
    "dtype": "<f4",
    "fill_value": "NaN",
    "filters": null,
    "order": "C",
    "shape": [
        12,
        401,
        901
    ],
    "zarr_format": 2
}
```

Dimensão de cada chunck

Tipo De compressão

Endianess e outras informações

MONAN Model for Ocean-laHd-Atmosphere predictioN

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

3 Um exemplo
Preparação dos Dados: Zarr

Como acessar os dados no formato Zarr?

Open in Colab

- Arquivos Zarr podem ser lidos diretamente em datasets do Xarray!

```
import zarr
import xarray as xr

ds = xr.open_zarr('VIESBAMH_20230216002023030300F.zarr')

#ou

#ds = xr.open_dataset('VIESBAMH_20230216002023030300F.zarr', engine='zarr', chunks='auto')

ds['temp850'].isel(time=-1).plot()
```

MONAN Model for Ocean-land-Atmosphere prediction

INPE

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

Um exemplo

Preparação dos Dados: Zarr

Como acessar os dados no formato Zarr?

- Estrutura de dados e chuncks.

Open in Colab

▼ Data variables:

pslc000	(time, lat, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
temp250	(time, lat, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
temp500	(time, lat, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
temp850	(time, lat, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
umes500	(time, lat, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	
	at, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...	

xarray.Dataset

▶ Dimensions: (**lat**: 401, **lon**: 901, **time**: 12)

▼ Coordinates:

lat	(lat)	float32	-80.0 -79.6 -79.2 ... 79.6 80.0
lon	(lon)	float32	0.0 0.4 0.8 ... 359.2 359.6 360.0
time	(time)	datetime64[ns]	2023-02-16 ... 2023-02-27

▶ Data variables: (19)

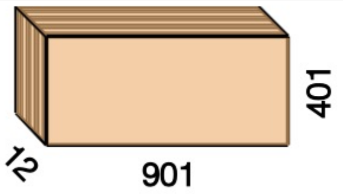
▼ Attributes:

comment :	virtual temperatura @ 925 hpa [k]
pdef :	None
storage :	99
title :	
undef :	-999.9

▼ Data variables:

pslc000	(time, lat, lon)	float32	dask.array<chunksize=(1, 401, 901), meta=n...

	Array	Chunk
Bytes	16.54 MiB	1.38 MiB
Shape	(12, 401, 901)	(1, 401, 901)
Count	13 Tasks	12 Chunks
Type	float32	numpy.ndarray



3 Um exemplo
Acesso aos Dados: Intake

Como construir um catálogo com o Intake?

1. Um catálogo do Intake pode conter múltiplas fontes de dados em diferentes formatos (eg., CSV, Zarr, Grib, netCDF, HDF etc);
2. O arquivo deve respeitar o formato YML.

```yaml
sources:

  scantec_df_bias_rmse_acor_gl:
    args:
      urlpath: https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/scantec_df_T1_gl.csv
    description: Bias, Root Mean Square Error and Anomaly Correlation for the DTC experiment (average gl area)
    driver: csv
    metadata:
      catalog_dir: https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/
      tags:
        - atmosphere
        - scantec
        - bam
        - data_assimilation
      url: https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/


  scantec_gl_rmse_dtc:
    args:
      consolidated: true
      urlpath: https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/RMSEDTC_202302160020230303O0F.zarr
    description: Root Mean Square Error for the DTC experiment (gl area)
    driver: intake_xarray.xzarr.ZarrSource
    metadata:
      catalog_dir: https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/
      tags:
        - atmosphere
        - scantec
        - bam
        - data_assimilation
      url: https://s0.cptec.inpe.br/pesquisa/das/dist/carlos.bastarz/MONAN/monan_post/data/
```

Exemplo:

- Arquivo **catalog.yml**:
  - Contém duas fontes de dados;
  - Cada formato possui um driver próprio;
  - Cada fonte de dados possui um endereço próprio;
  - Metadados podem ser utilizados para descrever as fontes de dados;
- Fonte de dados **scantec_df_bias_rmse_acor_gl**:
  - scantec: indica que é um dado proveniente do SCANTEC;
  - df: indica que é um dataframe (uma tabela);
  - bias_rmse_acor: indica que a tabela contém estas informações;
  - gl: indica o domínio (global).

3 Um exemplo
Acesso aos Dados: Intake

Como acessar um catálogo com o Intake?

Open in Colab

1. Importe a biblioteca Intake;
2. Obtenha o arquivo de catálogo de dados (caso não esteja armazenado no disco local);
3. Liste as fontes de dados;
4. Crie um objeto Dask;
5. Manipule as informações.

```
import intake

!wget https://raw.githubusercontent.com/monanadmin/monan_post/main/notebooks/catalog.yml

catalog = intake.open_catalog('catalog.yml')

list(catalog)

catalog['scantec_as_rmse_bamh’]

ds = catalog['scantec_as_rmse_bamh'].to_dask()

ds['temp850'].isel(time=-1).plot()
```

```
['scantec_df_bias_rmse_acor_gl',
 'scantec_gl_rmse_dtc',
 'scantec_gl_mean_dtc',
 'scantec_gl_vies_dtc',
 'scantec_hn_rmse_dtc',
 'scantec_hn_mean_dtc',
 'scantec_hn_vies_dtc',
 'scantec_tr_rmse_dtc',
 'scantec_tr_mean_dtc',
 'scantec_tr_vies_dtc',
 'scantec_hs_rmse_dtc',
 'scantec_hs_mean_dtc',
 'scantec_hs_vies_dtc',
 'scantec_as_rmse_dtc',
 'scantec_as_mean_dtc',
 'scantec_as_vies_dtc',
 'scantec_gl_rmse_bamh',
 'scantec_gl_mean_bamh',
 'scantec_gl_vies_bamh',
 'scantec_hn_rmse_bamh',
 'scantec_hn_mean_bamh',
 'scantec_hn_vies_bamh',
 'scantec_tr_rmse_bamh',
 'scantec_tr_mean_bamh',
 'scantec_tr_vies_bamh',
 'scantec_hs_rmse_bamh',
 'scantec_hs_mean_bamh',
 'scantec_hs_vies_bamh',
 'scantec_as_rmse_bamh',
 'scantec_as_mean_bamh',
 'scantec_as_vies_bamh',
 'scantec_gl_rmse_bamh0',
 'scantec_gl_mean_bamh0',
 'scantec_gl_vies_bamh0',
 'scantec_hn_rmse_bamh0',
 'scantec_hn_mean_bamh0',
 'scantec_hn_vies_bamh0',
 'scantec_tr_rmse_bamh0',
 'scantec_tr_mean_bamh0',
 'scantec_tr_vies_bamh0',
 'scantec_hs_rmse_bamh0',
 'scantec_hs_mean_bamh0',
 'scantec_hs_vies_bamh0',
 'scantec_as_rmse_bamh0',
 'scantec_as_mean_bamh0',
 'scantec_as_vies_bamh0']
```

MONAN Model for Ocean-laHd-Atmosphere predictioN

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

Obrigado!

MONAN
Model for Ocean-laNd-Atmosphere predictioN