

MERRA2_400.tavg1_2d_slv_Nx

Service Recipe For:

Filetype	Storage Location
NETCDF	S3



Note

This workflow used ArcGIS Server 10.8 and a CF v1.6 compliant NetCDF. For creating a Mosaic Dataset using MDCS, Linux ArcGIS Server 10.7.1 was used.

Step 1:

- Convert a sample NetCDF to a MRF to get the MRF XML structure

```
gdal_translate.exe -of MRF -co NOCOPY=TRUE -co COMPRESS=LERC -co CACHEDSOURCE=NETCDF:"/vsicurl/http://sdt-data.s3.amazonaws.com/gmao/merra2/MERRA2_400.tavg1_2d_slv_Nx.20190101.nc4":T2M:1 NETCDF:"/vsicurl/http://sdt-data.s3.amazonaws.com/gmao/merra2/MERRA2_400.tavg1_2d_slv_Nx.20190101.nc4":T2M:1 c:\temp\rr3.mrf
```

Step 2:

- Open .MRF and add in <DataFile>, <IndexFile>, and <CachedSource> to MRF file. This will make it a Raster Proxy pointing to the real NetCDF file.
 - <DataFile>
 - Location on the ArcGIS Server where the cache file will be created and stored
 - <IndexFile>
 - Location on the ArcGIS Server where the cache file will be created and stored
 - <CachedSource><Source>
 - Add the variable to be filtered out of the NetCDF into the source location. You can also use [vsicurl](#) or [vsi3](#) depending on if you ArcGIS Server's AWS IAM permissions.
 - NETCDF:"/vsicurl/https://sdt-data.s3.amazonaws.com/gmao/merra2/MERRA2_400.tavg1_2d_slv_Nx.20190101.nc4":T2M:1
 - NETCDF:"/vsi3/sdt-data/gmao/merra2/MERRA2_400.tavg1_2d_slv_Nx.20190101.nc4":T2M:1

MRF with Added Tags

```
<MRF_META>
  <CachedSource>
    <Source>NETCDF:"/vsicurl/https://sdt-data.s3.amazonaws.com/gmao/merra2/MERRA2_400.tavg1_2d_slv_Nx.20190101.nc4":T2M:1</Source>
  </CachedSource>
  <Raster>
    <Size x="576" y="361" c="1" />
    <PageSize x="576" y="361" c="1" />
    <Compression>LERC</Compression>
    <DataType>Float32</DataType>
    <DataValues NoData="999999986991104.00000000" />
    <DataFile>c:/temp/tavg1_2d_slv_Nx.20190101.nc4_T2M_1.mrfcache</DataFile>
    <IndexFile>c:/temp/tavg1_2d_slv_Nx.20190101.nc4_T2M_1.mrfcache</IndexFile>
  </Raster>
  <GeoTags>
    <BoundingBox minx="-180.31250000" miny="-90.25000000" maxx="179.68750000" maxy=" 90.25000000" />
    <Projection>EPSG:4326</Projection>
  </GeoTags>
</MRF_META>
```

Step 3:

- Create a Database Table in AWS RDS (PostgreSQL) or CSV with the following headings:

```
OBJECTID,Variable,Dimensions,StdTime,StdTime_max,StdZ,Raster,long_name,Name
```



Note

More table headings can be added. Minimum required are Variable, Dimensions, StdTime, and Raster if using a multidimensional dataset.

Step 4:

- Add the Raster Proxy to Table under the "Raster" column.

```
OBJECTID,Variable,Dimensions,StdTime,StdTime_max,StdZ,Raster,long_name,Name
1,T2M,"StdTime,StdZ",2019-01-01 00:30:00,2019-01-01 1:29:59,2,<MRF_META><CachedSource><Source>NETCDF:"/vsis3
/sdt-data/gmao/merra2/MERRA2_400.tavg1_2d_slv_Nx.20190101.nc4":T2M:0</Source></CachedSource><Raster><Size x="
576" y="361" c="1" /><PageSize x="576" y="361" c="1" /><Compression>LERC</Compression><DataType>Float32<
/DataType><DataValues NoData="999999986991104.00000000" /><DataFile>c:/temp/tavg1_2d_slv_Nx.20190101_nc4_T2M_1.
mrftcache</DataFile><IndexFile>c:/temp/tavg1_2d_slv_Nx.20190101_nc4_T2M_1.mrftcache</IndexFile><
/Raster><GeoTags><BoundingBox minx="-180.31250000" miny="-90.25000000" maxx="179.68750000" maxy=" 90.25000000"
/><Projection>EPSG:4326</Projection></GeoTags></MRF_META>,Temperature at 2 Meters,MERRA2_400.tavg1_2d_slv_Nx.
20190101.T2M.0
```

Step 5:

- Create Mosaic Dataset.
- Use the AddRasters to Mosaic tool. Make Raster Type = Table and point to the CSV or table in AWS RDS DB using connection file (.sde).



MDCS cannot handle AWS RDS input table with more than 4 (files) * 24 (hr) rows. It cannot create Mosaic Dataset with more than 2,000 rows. ArcGIS Pro 2.5 doesn't have such restriction.

Step 6:

- Right click on Mosaic Dataset.
- Select Modify > Build Multidimensional Info.
- Set Mosaic Dataset Properties as follows
 - Scientific
 - Number of Raster per mosaic
 - LERC Compression
 - Sort by StdTime
 - Set Use Time Start and Max
 - If StdZ Set as well
- Alternatiely, MDCS can be used for building multidimensional info and caculating statistics.

Step 7:

- Publish Mosaic Dataset using "Share as Web Layer" in ArcGIS Pro.
- Or use upload.py ArcPy script.
- If you add more rasters later using MDCS, restart service using ArcGIS Server Manager to see the new rasters.
 - Alternatiely, you can check "Has Live Data" option in the properties of the service.
 - Then, adjust time slider setting manually in Portal using "Advanced" option to view the newly added data.



ArcPy script cannot publish WMS/WCS. It can publish only ImageService. Use ArcGIS Pro to publish WMS/WCS.