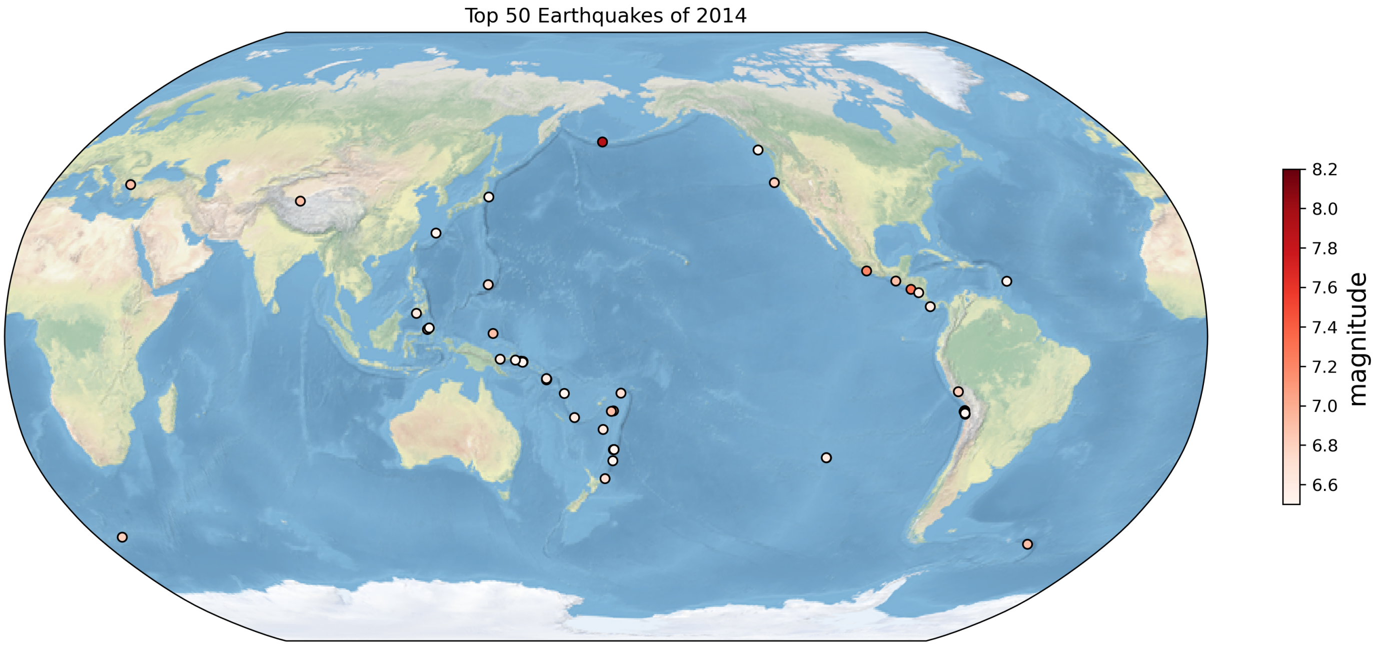
In [1]:

```python
# A4
# All of the codes and data accessing processes are independently made by Shao Shi
import numpy as np
import xarray as xr
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.ticker as mticker
import cartopy.crs as ccrs
import cartopy.feature as cfeature
from cartopy.mpl.gridliner import LONGITUDE_FORMATTER, LATITUDE_FORMATTER
```

In [2]:

```python
# Q1
# data processing
data = pd.read_csv("usgs_earthquakes.csv")
data['time']
top50 = data.sort_values(by=['mag'], ascending = False).head(50)
lon = top50['longitude'].values
lat = top50['latitude'].values
mag = top50['mag'].values

# start plotting
fig = plt.figure(figsize=(16, 12), dpi=300)
proj = ccrs.Robinson(central_longitude=180)
ax = plt.axes(projection=proj)
ax.stock_img()
sc = plt.scatter(lon, lat,c=mag, s =30, cmap='Reds',linewidths = 1,edgecolors = 'k', tra
plt.title('Top 50 Earthquakes of 2014')
cb = plt.colorbar(sc,shrink=0.3, ax = ax)
cb.ax.set_ylabel('magnitude',fontsize=15)
plt.show()
```



In [3]:

```python
# Q2
# data processing
ds = xr.open_dataset("MERRA2_400.tavg3_3d_cld_Nv.20221031.nc4", engine="netcdf4")
surfacePressure = ds.PS.isel(time=-1)
surfacePressure
```

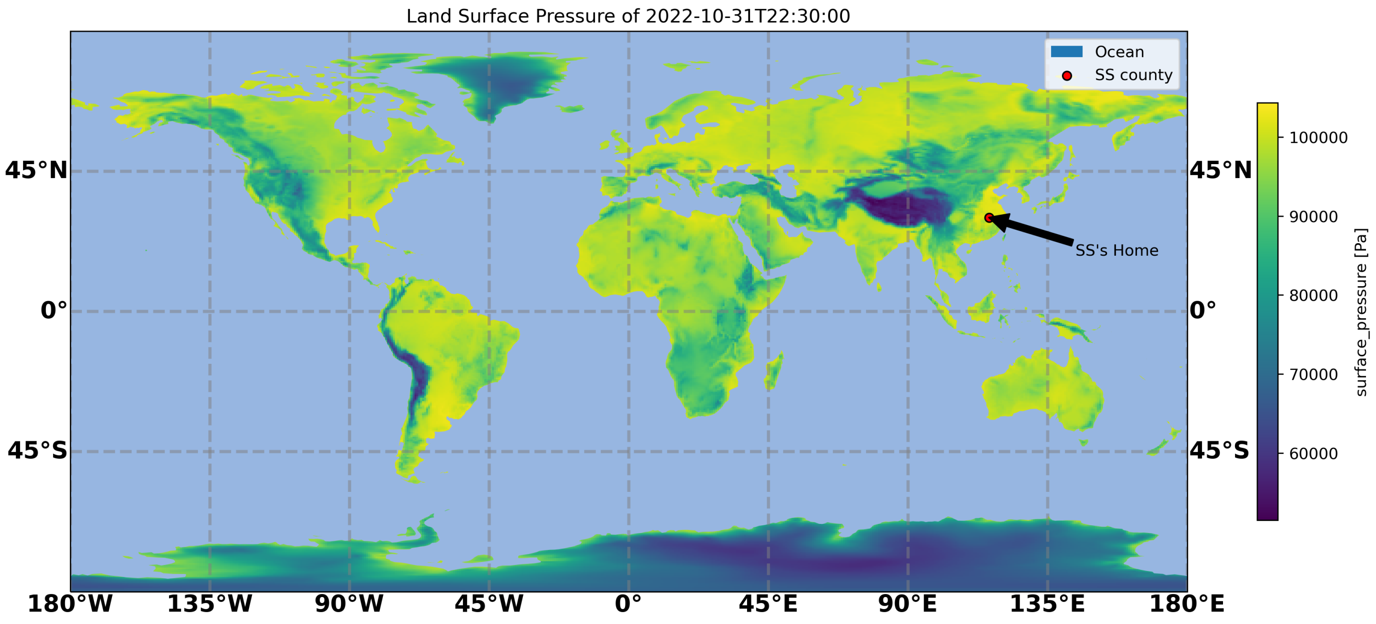
Out[3]:

```
xarray.DataArray   'PS'   (lat: 361, lon: 576)

[207936 values with dtype=float32]
```

```
▼ Coordinates:
  lat        (lat)         float64  -90.0 -89.5 -89.0 ... 89.5 90.0
  lon        (lon)         float64  -180.0 -179.4 ... 178.8 179.4
  time       ()     datetime64[ns]  2022-10-31T22:30:00
▼ Attributes:
  long_name :       surface_pressure
  units :           Pa
  fmissing_value :  1000000000000000.0
  standard_name :   surface_pressure
  vmax :            1000000000000000.0
  vmin :            -1000000000000000.0
  valid_range :     [-1.e+15  1.e+15]
  origname :        PS
  fullnamepath :    /PS
```

In [6]:

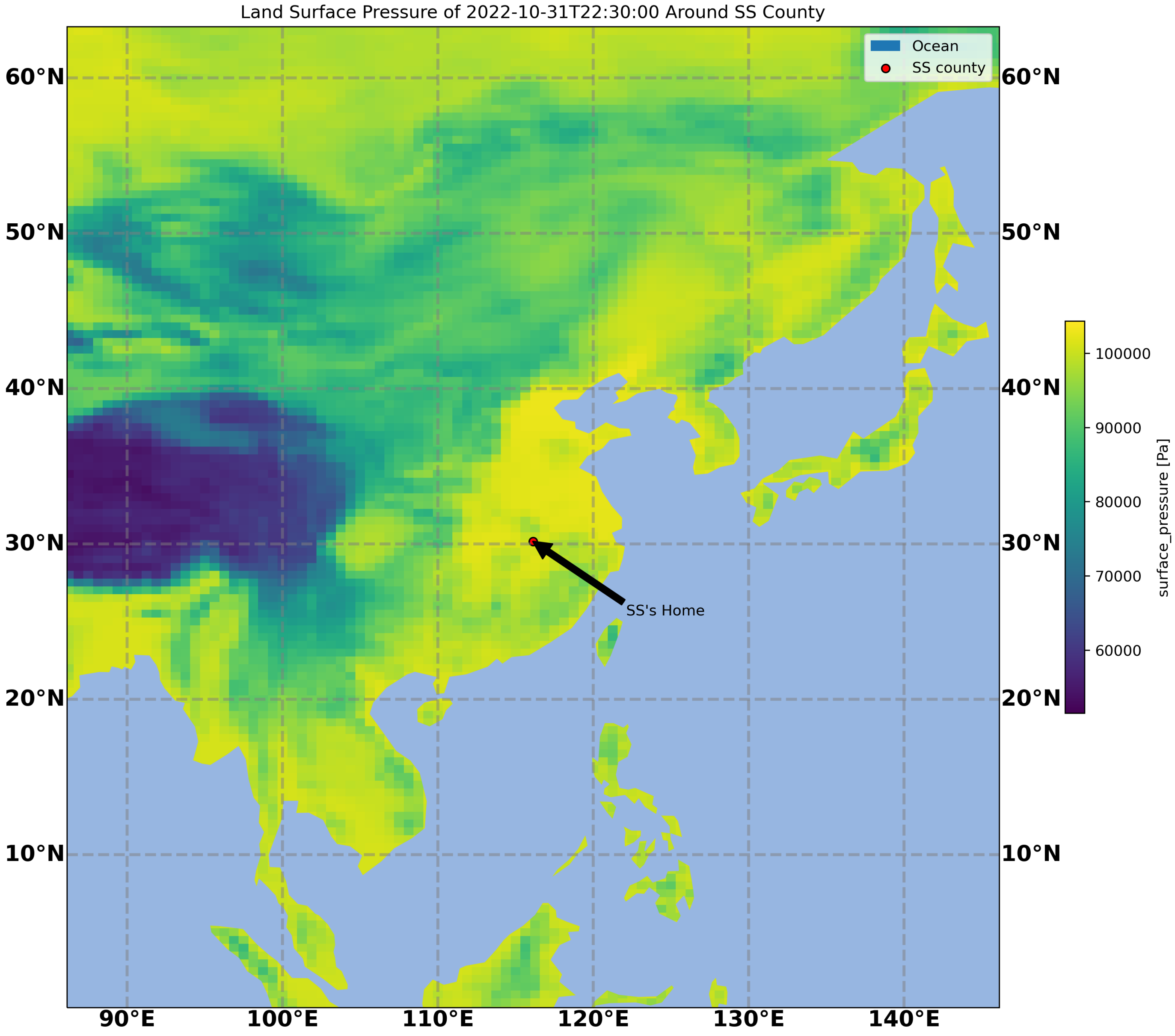
```python
# 2.1
fig = plt.figure(figsize=(16, 12), dpi=300)
# a project
# and colorbar
proj = ccrs.PlateCarree()
ax = plt.axes(projection=proj)
surfacePressure.plot(ax=ax, transform=ccrs.PlateCarree(),cbar_kwargs={'shrink': 0.4})

# title
plt.title('Land Surface Pressure of 2022-10-31T22:30:00')
gl = ax.gridlines(crs=ccrs.PlateCarree(), draw_labels=True,
                  linewidth=2, color='gray', alpha=0.5, linestyle='--')

# grid line
gl.xlines = True
gl.ylines = True

# x label and ticks
gl.xlabels_top = False
gl.xlocator = mticker.FixedLocator([-180,-135, -90, -45, 0, 45, 90, 135, 180])
gl.xformatter = LONGITUDE_FORMATTER
gl.xlabel_style = {'size': 15, 'color': 'black','weight': 'bold'}

# y label and ticks
gl.ylabels_left = True
gl.ylocator = mticker.FixedLocator([-90, -45, 0, 45, 90])
gl.yformatter = LATITUDE_FORMATTER
gl.ylabel_style = {'size': 15, 'color': 'black','weight': 'bold'}


# masks or features
ax.add_feature(cfeature.OCEAN, zorder=1)

scatter = ax.scatter(116.12, 30.15, s=30, linewidths = 1, color = 'r', edgecolors = 'k',

# legend
legend = ax.legend(["Ocean", "SS county"])

# annotation
# and text box
ax.annotate('SS\'s Home', xy=(116.12, 30.15),
             xycoords='data', xytext=(0.9, 0.6),
             textcoords='axes fraction',
             arrowprops=dict(facecolor='black'))

plt.show()
```



In [5]:

```
# 2.2
fig = plt.figure(figsize=(16, 12), dpi=300)
# a project
# and colorbar
proj = ccrs.PlateCarree()
ax = plt.axes(projection=proj)
surfacePressure.plot(ax=ax, transform=ccrs.PlateCarree(),cbar_kwargs={'shrink': 0.4})

central_lon, central_lat = 116.12, 30.15 # SS COUNTY
extent_shift = 30
extent = [central_lon-extent_shift, central_lon+extent_shift, central_lat-extent_shift,
ax.set_extent(extent)


# title
plt.title('Land Surface Pressure of 2022-10-31T22:30:00 Around SS County')
gl = ax.gridlines(crs=ccrs.PlateCarree(), draw_labels=True,
                  linewidth=2, color='gray', alpha=0.5, linestyle='--')

# grid line
gl.xlines = True
gl.ylines = True

# x label and ticks
gl.xlabels_top = False
gl.xformatter = LONGITUDE_FORMATTER
gl.xlabel_style = {'size': 15, 'color': 'black','weight': 'bold'}

# y label and ticks
gl.ylabels_left = True
gl.yformatter = LATITUDE_FORMATTER
gl.ylabel_style = {'size': 15, 'color': 'black','weight': 'bold'}


# masks or features
ax.add_feature(cfeature.OCEAN, zorder=1)

plt.scatter(116.12, 30.15, s=30, linewidths = 1, c = 'r', edgecolors = 'k', transform=pr

# legend
legend = ax.legend(["Ocean", "SS county"])
```

```
# annotation
# and text box
ax.annotate('SS\'s Home', xy=(116.12, 30.15),
            xycoords='data', xytext=(0.6, 0.4),
            textcoords='axes fraction',
            arrowprops=dict(facecolor='black'))

plt.show()
```



Land Surface Pressure of 2022-10-31T22:30:00 Around SS County

```
In [ ]:
```