MARBL diagnostics for vertical and horizontal nutrient distribution for ocean history files

In [1]:

```
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
import os
from glob import glob
from collections import OrderedDict
import xarray as xr
import numpy as np
import esmlab
import matplotlib.pyplot as plt
import cartopy
import cartopy.crs as ccrs
import seawater as sw
from seawater.library import T90conv
from scipy import stats
```

Define year range and get the CESM data

In [2]:

```
start_yr = 306
num_years = 61
endyr = start_yr + num_years
case = 'g.e22.GOMIPECOIAF_JRA-1p4-2018.TL319_g17.SMYLE.005'
user = 'klindsay'
```

In [3]:
```
files = []
for year in range(start_yr,endyr):
    yr4="{:04d}".format(year)
    print('doing simulation year', year, '!')
    for month in range(1, 13):

        mo2="{:02d}".format(month)
        files.extend(sorted(glob(f'/glade/scratch/{user}/archive/{case}/
        #files.extend(sorted(glob(f'/glade/scratch/{user}/{case}/run/{ca
```

```
doing simulation year 306 !
doing simulation year 307 !
doing simulation year 308 !
doing simulation year 309 !
doing simulation year 310 !
doing simulation year 311 !
doing simulation year 312 !
doing simulation year 313 !
doing simulation year 314 !
doing simulation year 315 !
doing simulation year 316 !
doing simulation year 317 !
doing simulation year 318 !
doing simulation year 319 !
doing simulation year 320 !
doing simulation year 321 !
doing simulation year 322 !
doing simulation year 323 !
doing simulation year 324 !
doing simulation year 325 !
doing simulation year 326 !
doing simulation year 327 !
doing simulation year 328 !
doing simulation year 329 !
doing simulation year 330 !
doing simulation year 331 !
doing simulation year 332 !
doing simulation year 333 !
doing simulation year 334 !
doing simulation year 335 !
doing simulation year 336 !
doing simulation year 337 !
doing simulation year 338 !
doing simulation year 339 !
doing simulation year 340 !
doing simulation year 341 !
doing simulation year 342 !
doing simulation year 343 !
doing simulation year 344 !
doing simulation year 345 !
doing simulation year 346 !
doing simulation year 347 !
doing simulation year 348 !
doing simulation year 349 !
doing simulation year 350 !
doing simulation year 351 !
doing simulation year 352 !
doing simulation year 353 !
doing simulation year 354 !
doing simulation year 355 !
doing simulation year 356 !
doing simulation year 357 !
doing simulation year 358 !
doing simulation year 359 !
doing simulation year 360 !
doing simulation year 361 !
doing simulation year 362 !
doing simulation year 363 !
doing simulation year 364 !
doing simulation year 365 !
doing simulation year 366 !
```

In [4]:
```
%%time
cesm_mon_ds=xr.open_mfdataset(files,decode_times=False,decode_coords=Fa
```

```
CPU times: user 6min 28s, sys: 12.9 s, total: 6min 41s
Wall time: 24min 12s
```

Keep only the variables we need

In [5]:
```
variables = ['sp_Fe_lim_surf','sp_P_lim_surf','sp_N_lim_surf',
             'diat_Fe_lim_surf', 'diat_P_lim_surf','diat_N_lim_surf','di
             'diaz_P_lim_surf','diaz_Fe_lim_surf','photoC_TOT_zint','pho
             'photoC_diat_zint','photoC_diaz_zint',
             'CaCO3_PROD_zint','SiO2_PROD','POC_FLUX_100m','spCaCO3','di
             'spC','diazC','ALK','SALT']
coords = {'x':'TLONG','y':'TLAT'}
```

In [6]:
```
coords = {'x':'TLONG','y':'TLAT'}
keepthese=['z_t','z_t_150m','time_bound','TAREA','PO4','Fe','NO3','SiO3
keep_vars = keepthese +list(coords.values())+['dz','KMT']
cesm_mon_ds = cesm_mon_ds.drop([v for v in cesm_mon_ds.variables if v no
```

In [7]:
```
cesm_ann_ds=cesm_mon_ds.mean(dim='time')
```

In [3]:
```
file='tmp_mean.nc'
#cesm_ann_ds.to_netcdf(file)
cesm_ann_ds = xr.load_dataset(file,decode_times=False)
```

World Ocean Atlas 2013

In [4]:
```
file = '/glade/work/kristenk/WOA_data/regrid_POP/WOA2013_POPgrid.nc'
```

In [5]:
```
ds_woa = xr.load_dataset(file, decode_times=False, decode_coords=False)
```

In [6]:
```
ds_woa['z_t'] = cesm_ann_ds.z_t
```

In [7]:
```
NO3_diff = cesm_ann_ds.NO3 - ds_woa.NO3
PO4_diff = cesm_ann_ds.PO4 - ds_woa.PO4
SiO3_diff = cesm_ann_ds.SiO3 - ds_woa.SiO3
```

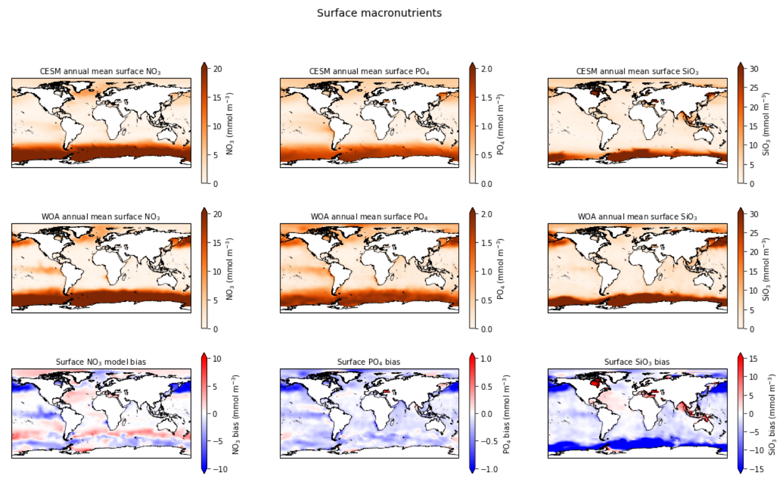
In [8]:
```
lons=ds_woa.TLONG
lats=ds_woa.TLAT
area=ds_woa.TAREA
depths=ds_woa.z_t * 0.01
```

In [14]:
```
rmse_global = xr.Dataset({v: cesm_ann_ds[v] for v in ['z_t']})
rmse_global['NO3']=esmlab.statistics.weighted_rmsd(cesm_ann_ds.NO3, ds_w
rmse_global['PO4']=esmlab.statistics.weighted_rmsd(cesm_ann_ds.PO4, ds_w
rmse_global['SiO3']=esmlab.statistics.weighted_rmsd(cesm_ann_ds.SiO3, ds
```

Surface nutrients

```
In [65]: fig = plt.figure(figsize=(18,10))
plt.suptitle('Surface macronutrients', fontsize=14)

####################NO3
#COLUMN 1  - NO3
#---- CESM panel
ax = fig.add_subplot(3,3,1, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('CESM annual mean surface NO$_3$', fontsize=10)
pc1=ax.pcolormesh(lons, lats,
                 cesm_ann_ds.NO3.isel(z_t=0), vmin=0, vmax=20, cmap='Ora
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc1, ax=ax,extend='max',label='NO$_3$ (mmol m$^{-3}

#---- OBS panel
ax = fig.add_subplot(3,3,4, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('WOA annual mean surface NO$_3$', fontsize=10)
pc2=ax.pcolormesh(lons, lats,
                 ds_woa.NO3.isel(z_t=0), vmin=0, vmax=20, cmap='Oranges
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc2, ax=ax,extend='max',label='NO$_3$ (mmol m$^{-3}

#---- DIFF panel
ax = fig.add_subplot(3,3,7, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('Surface NO$_3$ model bias', fontsize=10)
pc3=ax.pcolormesh(lons, lats,
                 NO3_diff.isel(z_t=0), vmin=-10, vmax=10, cmap='bwr',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc3, ax=ax,extend='both',label='NO$_3$ bias (mmol m



####################PO4
#---- CESM panel
ax = fig.add_subplot(3,3,2, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('CESM annual mean surface PO$_4$', fontsize=10)
pc1=ax.pcolormesh(lons, lats,
                 cesm_ann_ds.PO4.isel(z_t=0), vmin=0, vmax=2, cmap='Oran
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc1, ax=ax,extend='max',label='PO$_4$ (mmol m$^{-3}

#---- OBS panel
ax = fig.add_subplot(3,3,5, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('WOA annual mean surface PO$_4$', fontsize=10)
pc2=ax.pcolormesh(lons, lats,
                 ds_woa.PO4.isel(z_t=0), vmin=0, vmax=2, cmap='Oranges',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc2, ax=ax,extend='max',label='PO$_4$ (mmol m$^{-3}

#---- DIFF panel
ax = fig.add_subplot(3,3,8, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('Surface PO$_4$ bias', fontsize=10)
pc3=ax.pcolormesh(lons, lats,
                 PO4_diff.isel(z_t=0), vmin=-1, vmax=1, cmap='bwr',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc3, ax=ax,extend='both',label='PO$_4$ bias (mmol m



####################SiO3
#---- CESM panel
ax = fig.add_subplot(3,3,3, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('CESM annual mean surface SiO$_3$', fontsize=10)
pc1=ax.pcolormesh(lons, lats,
                 cesm_ann_ds.SiO3.isel(z_t=0),
                  vmin=0, vmax=30,
                  cmap='Oranges',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc1, ax=ax,extend='max',label='SiO$_3$ (mmol m$^{-3

#---- OBS panel
ax = fig.add_subplot(3,3,6, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('WOA annual mean surface SiO$_3$', fontsize=10)
pc2=ax.pcolormesh(lons, lats,
                 ds_woa.SiO3.isel(z_t=0),
                  vmin=0, vmax=30,
                  cmap='Oranges',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc2, ax=ax,extend='max',label='SiO$_3$ (mmol m$^{-3
```

```
#---- DIFF panel
ax = fig.add_subplot(3,3,9, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('Surface SiO$_3$ bias', fontsize=10)
pc3=ax.pcolormesh(lons, lats,
                 SiO3_diff.isel(z_t=0),
                  vmin=-15, vmax=15,
                  cmap='bwr',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc3, ax=ax,extend='both',label='SiO$_3$ bias (mmol
```

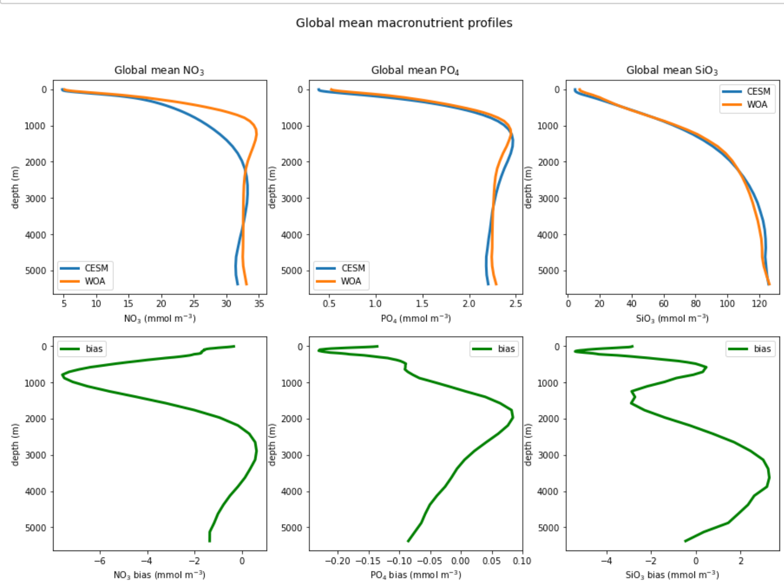


Global nutrient profiles

In [19]:
```
ds_glb = xr.Dataset({v: cesm_ann_ds[v] for v in ['z_t']})
ds_glb['NO3']= esmlab.weighted_mean(cesm_ann_ds['NO3'], weights=area, d:
ds_glb['PO4']= esmlab.weighted_mean(cesm_ann_ds['PO4'], weights=area, d:
ds_glb['SiO3']= esmlab.weighted_mean(cesm_ann_ds['SiO3'], weights=area,
```

In [66]:
```
ds_glb_woa = xr.Dataset({v: cesm_ann_ds[v] for v in ['z_t']})
ds_glb_woa['NO3'] = esmlab.weighted_mean(ds_woa['NO3'], weights=area, d:
ds_glb_woa['PO4'] = esmlab.weighted_mean(ds_woa['PO4'], weights=area, d:
ds_glb_woa['SiO3'] = esmlab.weighted_mean(ds_woa['SiO3'], weights=area,
```

In [67]:
```
fig = plt.figure(figsize=(15,10))

plt.suptitle('Global mean macronutrient profiles', fontsize=14)


#COLUMN 1  - NO3

ax = fig.add_subplot(2,3,1)
ax.set_title('Global mean NO$_3$')
ax.plot(ds_glb['NO3'].values, depths, label='CESM', linewidth=3)
ax.plot(ds_glb_woa['NO3'].values, depths, label='WOA', linewidth=3)
ax.legend()
ax.set(ylabel='depth (m)',xlabel='NO$_3$ (mmol m$^{-3}$)')
plt.gca().invert_yaxis()

#COLUMN 2  - PO4

ax = fig.add_subplot(2,3,2)
ax.set_title('Global mean PO$_4$')
ax.plot(ds_glb['PO4'].values, depths, label='CESM', linewidth=3)
ax.plot(ds_glb_woa['PO4'].values, depths, label='WOA', linewidth=3)
ax.legend()
ax.set(ylabel='depth (m)',xlabel='PO$_4$ (mmol m$^{-3}$)')
plt.gca().invert_yaxis()

#COLUMN 3  - SiO3

ax = fig.add_subplot(2,3,3)
ax.set_title('Global mean SiO$_3$')
ax.plot(ds_glb['SiO3'].values, depths, label='CESM', linewidth=3)
ax.plot(ds_glb_woa['SiO3'].values, depths, label='WOA', linewidth=3)
ax.legend()
ax.set(ylabel='depth (m)',xlabel='SiO$_3$ (mmol m$^{-3}$)')
plt.gca().invert_yaxis()


#COLUMN 1  - NO3 diff

ax = fig.add_subplot(2,3,4)
ax.plot(ds_glb['NO3'].values - ds_glb_woa['NO3'].values, depths, label=
ax.legend()
ax.set(ylabel='depth (m)',xlabel='NO$_3$ bias (mmol m$^{-3}$)')
plt.gca().invert_yaxis()

#COLUMN 2  - PO4 diff

ax = fig.add_subplot(2,3,5)
ax.plot(ds_glb['PO4'].values - ds_glb_woa['PO4'].values, depths, label=
ax.legend()
ax.set(ylabel='depth (m)',xlabel='PO$_4$ bias (mmol m$^{-3}$)')
plt.gca().invert_yaxis()

#COLUMN 3  - SiO3 diff

ax = fig.add_subplot(2,3,6)
ax.plot(ds_glb['SiO3'].values - ds_glb_woa['SiO3'].values, depths, labe
ax.legend()
ax.set(ylabel='depth (m)',xlabel='SiO$_3$ bias (mmol m$^{-3}$)')
plt.gca().invert_yaxis()
```

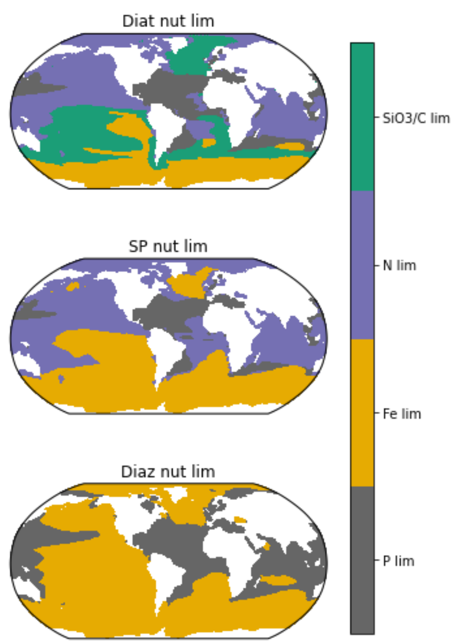


Phytoplankton nutrient limitation at surface

In [37]:
```
#most limiting nutrient - concatenate the limitation terms so that nutr
# 0 = PO4
# 1 = Fe
# 2 = NO3 (only for sp and diat)
# 3 = Si (only for diat)

limarray_sp=xr.concat((cesm_ann_ds.sp_P_lim_surf, cesm_ann_ds.sp_Fe_lim_
limarray_diat=xr.concat((cesm_ann_ds.diat_P_lim_surf, cesm_ann_ds.diat_I
limarray_diaz=xr.concat((cesm_ann_ds.diaz_P_lim_surf, cesm_ann_ds.diaz_I
```

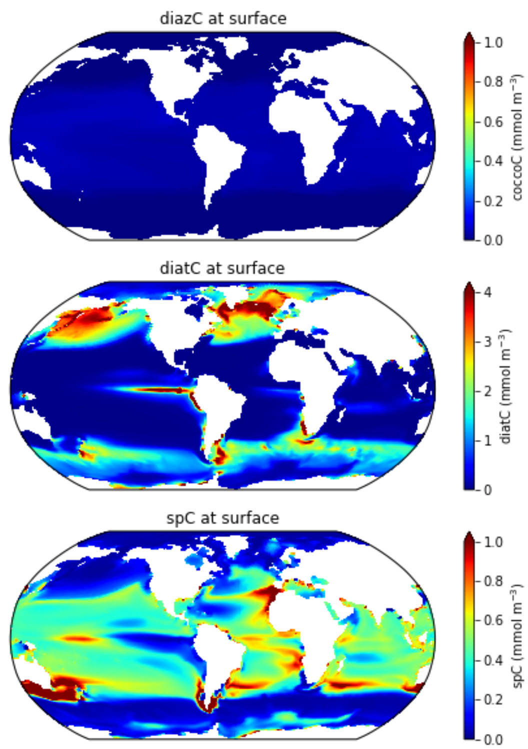
In [38]:
```
most_lim_sp=limarray_sp.argmin(dim='nutrient', skipna=False).squeeze()
most_lim_diat=limarray_diat.argmin(dim='nutrient', skipna=False).squeeze
most_lim_diaz=limarray_diaz.argmin(dim='nutrient', skipna=False).squeeze
mask = np.isnan(cesm_ann_ds.sp_N_lim_surf.squeeze())
```

In [39]:
```
fig = plt.figure(figsize=(5,9))

ax = fig.add_subplot(3,1,1, projection=ccrs.Robinson(central_longitude=
ax.set_title('Diat nut lim', fontsize=12)
pc=ax.pcolormesh(lons, lats, most_lim_diat.where(~mask), cmap=plt.cm.get
colorbar_specs = {'ticks' : np.arange(0,4,1)}

ax = fig.add_subplot(3,1,2, projection=ccrs.Robinson(central_longitude=
ax.set_title('SP nut lim', fontsize=12)
pc=ax.pcolormesh(lons, lats, most_lim_sp.where(~mask), cmap=plt.cm.get_c
colorbar_specs = {'ticks' : np.arange(0,4,1)}

ax = fig.add_subplot(3,1,3, projection=ccrs.Robinson(central_longitude=
ax.set_title('Diaz nut lim', fontsize=12)
pc=ax.pcolormesh(lons, lats, most_lim_diaz.where(~mask), cmap=plt.cm.get
colorbar_specs = {'ticks' : np.arange(0,4,1)}


fig.subplots_adjust(right=0.8)
cbar_ax = fig.add_axes([0.85, 0.15, 0.05, 0.7])
cbar = fig.colorbar(pc, cax=cbar_ax,**colorbar_specs)
cbar.ax.set_yticklabels(['P lim', 'Fe lim', 'N lim','SiO3/C lim']);
```



Look at phyto carbon pools

In [27]:
```
fig = plt.figure(figsize=(8,10))

ax = fig.add_subplot(3,1,1, projection=ccrs.Robinson(central_longitude=:
ax.set_title('diazC at surface', fontsize=12)
pc=ax.pcolormesh(lons, lats, cesm_ann_ds.diazC.isel(z_t_150m=0), cmap=':

cbar1 = fig.colorbar(pc, ax=ax,extend='max',label='coccoC (mmol m$^{-3}$

ax = fig.add_subplot(3,1,2, projection=ccrs.Robinson(central_longitude=:
ax.set_title('diatC at surface', fontsize=12)
pc=ax.pcolormesh(lons, lats, cesm_ann_ds.diatC.isel(z_t_150m=0), cmap=':

cbar1 = fig.colorbar(pc, ax=ax,extend='max',label='diatC (mmol m$^{-3}$)

ax = fig.add_subplot(3,1,3, projection=ccrs.Robinson(central_longitude=:
ax.set_title('spC at surface', fontsize=12)
pc=ax.pcolormesh(lons, lats, cesm_ann_ds.spC.isel(z_t_150m=0), cmap='jet

cbar1 = fig.colorbar(pc, ax=ax,extend='max',label='spC (mmol m$^{-3}$)':
```

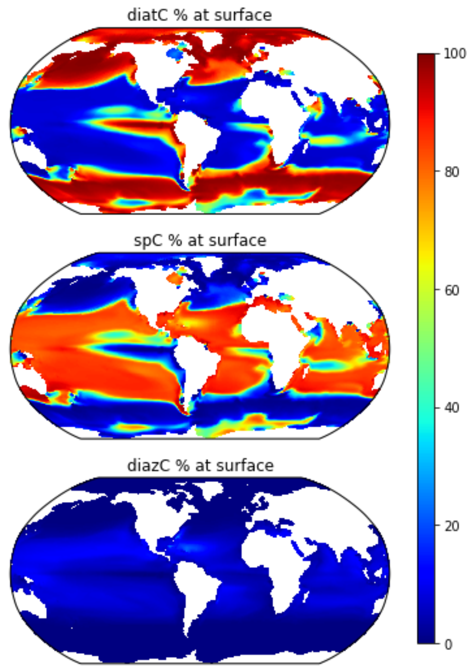


Look at percent phytoC

In [30]:
```
phytoC = cesm_ann_ds.spC + cesm_ann_ds.diatC + cesm_ann_ds.diazC
perc_sp = cesm_ann_ds.spC / (phytoC) * 100.
perc_diat = cesm_ann_ds.diatC / (phytoC) * 100.
perc_diaz = cesm_ann_ds.diazC / (phytoC) * 100.
```

In [62]:

```
fig = plt.figure(figsize=(6,9))

ax = fig.add_subplot(3,1,1, projection=ccrs.Robinson(central_longitude=:
ax.set_title('diatC % at surface', fontsize=12)
pc=ax.pcolormesh(lons, lats, perc_diat.isel(z_t_150m=0), cmap='jet',vmin

ax = fig.add_subplot(3,1,2, projection=ccrs.Robinson(central_longitude=:
ax.set_title('spC % at surface', fontsize=12)
pc=ax.pcolormesh(lons, lats, perc_sp.isel(z_t_150m=0), cmap='jet',vmin=(

ax = fig.add_subplot(3,1,3, projection=ccrs.Robinson(central_longitude=:
ax.set_title('diazC % at surface', fontsize=12)
pc=ax.pcolormesh(lons, lats, perc_diaz.isel(z_t_150m=0), cmap='jet',vmin

fig.subplots_adjust(right=0.8)
cbar_ax = fig.add_axes([0.85, 0.15, 0.03, 0.7])
fig.colorbar(pc, cax=cbar_ax);
```



NPP

In [21]:

```
nmols_to_PgCyr = 1e-9 * 12. * 1e-15 * 365. * 86400.
```

In [31]:
```
variables = [f'photoC_{phyto}_zint' for phyto in ['diat', 'sp', 'diaz',

ds_glb = xr.Dataset()
for v in variables:
    ds_glb[v] = esmlab.statistics.weighted_sum(cesm_ann_ds[v], weights=a
    ds_glb[v].attrs = cesm_ann_ds[v].attrs

    #if ds1_annmean[v].units == 'mmol/m^3 cm/s':
    ds_glb[v] = ds_glb[v] * nmols_to_PgCyr
    ds_glb[v].attrs['units'] = 'Pg C yr$^{-1}$'

ds_glb = ds_glb.compute()
ds_glb
```

Out[31]:

xarray.Dataset

► Dimensions:

► Coordinates: (0)

▼ Data variables:

photoC_diat_zint	()	float64	21.37
photoC_sp_zint	()	float64	22.43
photoC_diaz_zint	()	float64	1.607
photoC_TOT_zint	()	float64	45.41

► Attributes: (0)

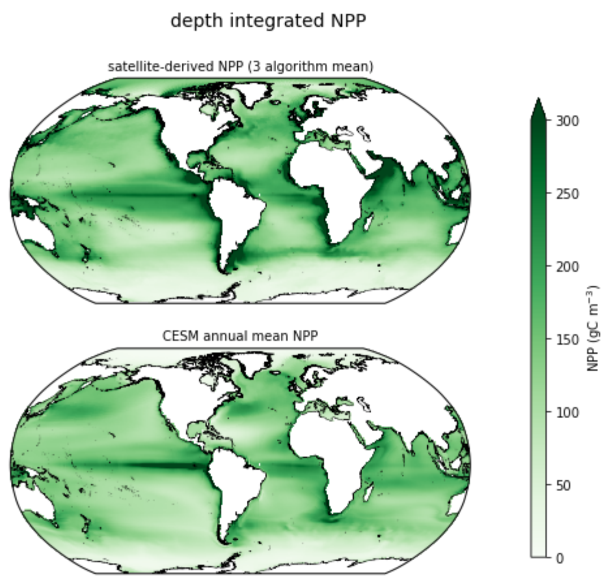
Globally integrated NPP should be between 50 and 60 Pg C yr^{-1} according to satellite derived NPP algorithms

Compare to satellite-derived NPP

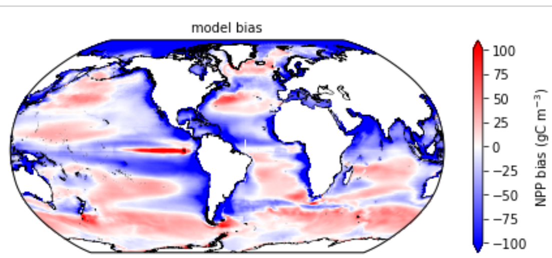
In [9]:
```
file='/glade/work/kristenk/satellite_data/POP_regrid/NPP_3methodmean_mea
ds_npp=xr.open_dataset(file, decode_times=False)
```

In [10]:
```
npp = cesm_ann_ds.photoC_TOT_zint * 0.01 * 86400 * 365 * 0.001 * 12. #co
```

In [33]:
```
#plot depth integrated NPP

fig = plt.figure(figsize=(8,7))
plt.suptitle('depth integrated NPP', fontsize=14)

####################NO3
#COLUMN 1  - NO3
#---- CESM panel
ax = fig.add_subplot(2,1,2, projection=ccrs.Robinson(central_longitude=:
ax.coastlines('10m',linewidth=0.5)
ax.set_title('CESM annual mean NPP', fontsize=10)
pc1=ax.pcolormesh(lons, lats,
                  npp, cmap='Greens',
                   vmin=0, vmax=300,
                  transform=ccrs.PlateCarree())

#---- OBS panel
ax = fig.add_subplot(2,1,1, projection=ccrs.Robinson(central_longitude=:
ax.coastlines('10m',linewidth=0.5)
ax.set_title('satellite-derived NPP (3 algorithm mean)', fontsize=10)
pc2=ax.pcolormesh(lons, lats,
                  ds_npp.NPP, cmap='Greens',
                   vmin=0, vmax=300,
                  transform=ccrs.PlateCarree())


fig.subplots_adjust(right=0.8)
cbar_ax = fig.add_axes([0.85, 0.15, 0.02, 0.7])
fig.colorbar(pc2, cax=cbar_ax,extend='max',label='NPP (gC m$^{-3}$)');
```



In [28]:
```
fig = plt.figure(figsize=(8,3))
#---- diff
npp_diff=npp - ds_npp.NPP

ax = fig.add_subplot(1,1,1, projection=ccrs.Robinson(central_longitude=:
ax.coastlines('10m',linewidth=0.5)
ax.set_title('model bias', fontsize=10)
pc2=ax.pcolormesh(lons, lats,
                  npp_diff, cmap='bwr',
                   vmin=-100, vmax=100,
                  transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc2, ax=ax,extend='both',label='NPP bias (gC m$^{-:
```



Calcification

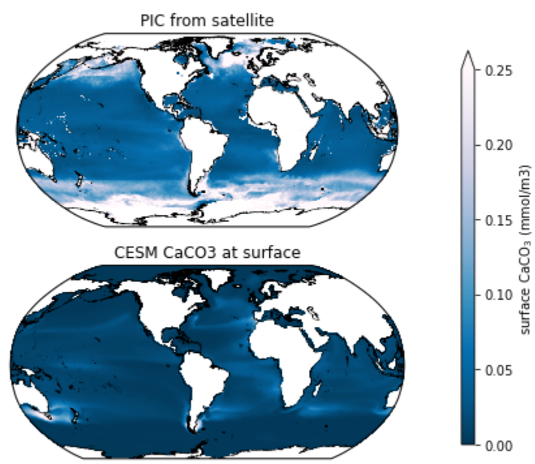
In [22]:
```
## GLOBALLY integrated calcification; observation-based estimates range
## (Feely et al., 2004, Maranon et al., 2016, Smith et al., 2016, Balch
ds_glb['CaCO3_PROD_zint'] = esmlab.statistics.weighted_sum(cesm_ann_ds[
ds_glb['CaCO3_PROD_zint'] = ds_glb['CaCO3_PROD_zint'] * nmols_to_PgCyr
ds_glb['CaCO3_PROD_zint'].values
print('Globally integrated calcification is',ds_glb['CaCO3_PROD_zint'].v
print('(should be between 0.6 to 2.4 Pg C per year)')
```

```
Globally integrated calcification is 0.5967147978656123 Pg C per year
(should be between 0.6 to 2.4 Pg C per year)
```

Compare satellite-derived Particulate Inorganic Carbon (PIC; i.e., $CaCO_3$) to $spCaCO_3$ in the top 10m

In [23]:
```
#read in mean PIC
pic_file = '/glade/work/kristenk/satellite_data/processed/PIC_mean2003to
ds_pic=xr.open_dataset(pic_file, decode_times=False)
#convert to mmol/m3 from mg/m3
ds_pic['PIC']=ds_pic.PIC / 12.011
```

In [35]:
```
fig = plt.figure(figsize=(7,6))

ax = fig.add_subplot(2,1,1, projection=ccrs.Robinson(central_longitude=1
ax.set_title('PIC from satellite', fontsize=12)
ax.coastlines('10m',linewidth=0.5)
pc=ax.pcolormesh(ds_pic.lon.values, ds_pic.lat.values, ds_pic.PIC, cmap=

ax = fig.add_subplot(2,1,2, projection=ccrs.Robinson(central_longitude=1
ax.set_title('CESM CaCO3 at surface', fontsize=12)
ax.coastlines('10m',linewidth=0.5)
pc=ax.pcolormesh(lons, lats, cesm_ann_ds.spCaCO3.isel(z_t_150m=0), cmap=


fig.subplots_adjust(right=0.8)
cbar_ax = fig.add_axes([0.85, 0.15, 0.02, 0.7])
fig.colorbar(pc, cax=cbar_ax,extend='max',label='surface CaCO$_3$ (mmol/
```

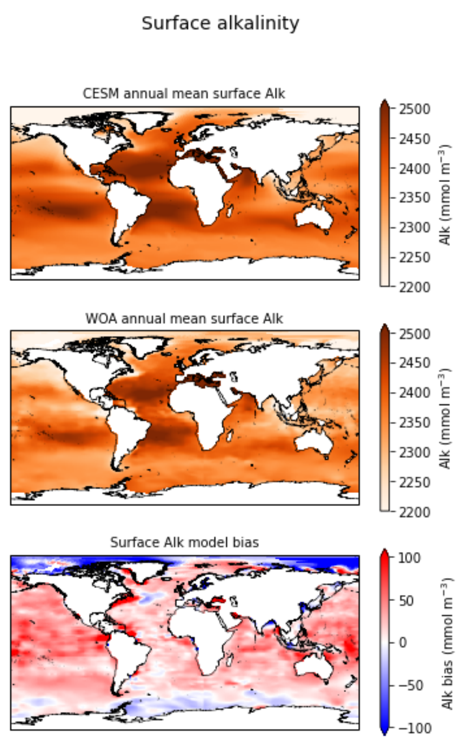


Alkalinity

In [42]:
```
#just getting this for the coords (that don't have nans)
file = '/glade/work/kristenk/GLODAPv2_regridded/glodap_pop_grid/GLODAP_I
ds_glodap = xr.load_dataset(file, decode_times=False, decode_coords=Fals
ds_glodap['z_t']=cesm_ann_ds.z_t
ALK_diff = cesm_ann_ds.ALK - ds_glodap.ALK
#rmse_global['ALK']=esmlab.statistics.weighted_rmsd(cesm_ann_ds.ALK, ds_
#rmse_global.ALK.values
```

In [63]:

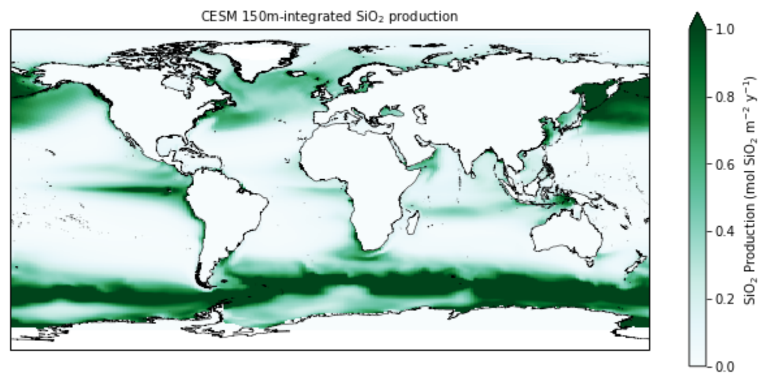
```
#plot surface alkalinity

fig = plt.figure(figsize=(6,9))
plt.suptitle('Surface alkalinity', fontsize=14)

###################NO3
#COLUMN 1  - NO3
#---- CESM panel
ax = fig.add_subplot(3,1,1, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('CESM annual mean surface Alk', fontsize=10)
pc1=ax.pcolormesh(lons, lats,
                  cesm_ann_ds.ALK.isel(z_t=0), cmap='Oranges',
                   vmin=2200, vmax=2500,
                  transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc1, ax=ax,extend='max',label='Alk (mmol m$^{-3}$)

#---- OBS panel
ax = fig.add_subplot(3,1,2, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('WOA annual mean surface Alk', fontsize=10)
pc2=ax.pcolormesh(lons, lats,
                  ds_glodap.ALK.isel(z_t=0), cmap='Oranges',
                   vmin=2200, vmax=2500,
                  transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc2, ax=ax,extend='max',label='Alk (mmol m$^{-3}$)

#---- DIFF panel
ax = fig.add_subplot(3,1,3, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('Surface Alk model bias', fontsize=10)
pc3=ax.pcolormesh(lons, lats,
                  ALK_diff.isel(z_t=0), vmin=-100, vmax=100, cmap='bwr',
                  transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc3, ax=ax,extend='both',label='Alk bias (mmol m$^
```



Silicification

In [70]:
```
# integrate SiO2 production over top 150 m
SiO2_prod_int = cesm_ann_ds['SiO2_PROD'].isel(z_t=slice(0,15)) * 10. #e
SiO2_prod_int = SiO2_prod_int.sum(dim='z_t') #units go from mmolSi/m3/s
SiO2_prod_int = SiO2_prod_int * 86400. * 365. * 0.001 #convert to molSi,
```

In [71]:
```
fig = plt.figure(figsize=(11,5))

#---- CESM panel
ax = fig.add_subplot(1,1,1, projection=ccrs.PlateCarree())
ax.set_extent([-180, 180, -90, 90], ccrs.PlateCarree())
ax.coastlines('10m',linewidth=0.5)
ax.set_title('CESM 150m-integrated SiO$_2$ production', fontsize=10)
pc1=ax.pcolormesh(lons, lats,
                 SiO2_prod_int,
                  vmin=0, vmax=1,
                  cmap='BuGn',
                 transform=ccrs.PlateCarree())
cbar1 = fig.colorbar(pc1, ax=ax,extend='max',label='SiO$_2$ Production (
```



In [73]:
```
### GLOBALLY integrated biogenic SiO2 production is between 166 and 280
gc_si_prod = SiO2_prod_int * area_m2 #molSi/gc/yr

ds_glb['SiO2_PROD'] = gc_si_prod.sum(dim='nlon').sum(dim='nlat') * 1.e-1

ds_glb['SiO2_PROD'].values
```

Out[73]:
```
array(111.15986714)
```

GLOBALLY integrated biogenic SiO2 production is between 166 and 280 mol Si per year, but half is dissolved in the upper 100m (Nelson et al., 1995; Holzer et al., 2014)

POC flux at 100m

In [61]:
```
ds_glb['POC_FLUX_100m'] = esmlab.statistics.weighted_sum(cesm_ann_ds['P
ds_glb['POC_FLUX_100m'] = ds_glb['POC_FLUX_100m'] * nmols_to_PgCyr
ds_glb['POC_FLUX_100m'].values
print('Globally integrated POC flux is',ds_glb['POC_FLUX_100m'].values,
```

```
Globally integrated POC flux is 7.05632704198471 Pg C per year
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: