Climate Code Foundation

Who are we?

A non-profit organisation founded in August 2010; our goal is to promote the public understanding of climate science,

by increasing the visibility and clarity of the software used in climate science, and by encouraging climate scientists to do the same;

by encouraging good software development and management practices among climate scientists;

by encouraging the publication of climate science software as open source.

Nick Barnes
nb@climatecode.org
+44 7739 461975

David Jones
drj@climatecode.org
+44 7792 748613

Photo credits: www.serenaatkins.com (nb); Philippa Davey (drj)

What can we do?

Joint projects with academic partners: improve your software, bolster your publications; become field leaders in software quality and transparency.

Presentations and consultation on improving software practices: tools, techniques, and strategies.

Create inter-disciplinary networks focused on software issues.

Work with funding bodies, institutions, and publishers, on software repositories and resources.

Clear Climate Code

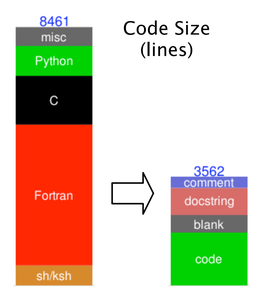
a CCF project

Leading by example to produce **clear** climate science code. Making code clearer leads to greater understanding, and allows us to more easily reason about the scientific principles underlying the code.

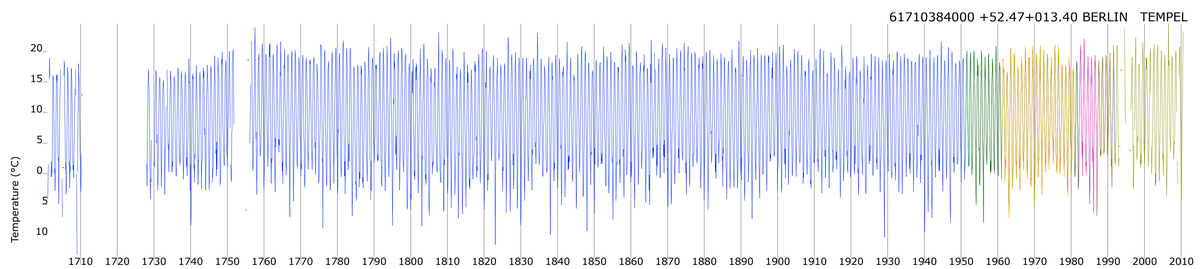
Our flagship product is **ccc-gistemp**, a reimplementation in Python of the NASA GISS GISTEMP surface temperature reconstruction algorithm.

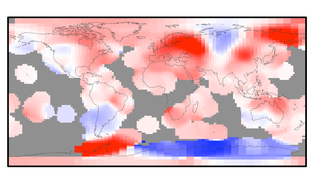
```
#!/bin/ksh
if [[ $# -lt 1 ]] ; then echo "Usage: $0 v2.mean [7 to skip oustr.source) [1 to skip US-USHCN " ; exit ; fi

infile=input_files/$1 ; skip_US=${3:-0}    ; # default: use all US station data
if [[ ! -s $infile ]] ; then echo "$infile not found" ; exit ; fi

mkdir temp_files   2> /dev/null
mkdir work_files   2> /dev/null

fortran_compile=$FC
if [[ $FC = '' ]]
then echo "set an environment variable FC to the fortran_compile_command like f90"
     echo "or do all compilation first and comment the compilation lines"
     exit
fi

echo "Bringing Antarctic tables closer to ${infile} format"
./antarc_to_v2.sh $2

echo "adding extra Antarctica station data to ${infile}"
${fortran_compile} antarc_comb.f -o antarc_comb.exe
antarc_comb.exe v2_antarct.dat ${infile}
echo "created v2.meanx from v2_antarct.dat and ${infile}"
rm -f v2_antarct.dat antarc_comb.exe

# echo "removing pre-1880 data:"
echo ; echo "GHCN data:"
${fortran_compile} dump_old.f -o dump_old.exe
dump_old.exe v2.meanx v2.meany 1880 ; echo "created v2.meany from v2.meanx"
rm -f v2.meanx

echo " "
echo "  reformat USHCN to v2.mean format"
./get_USHCN ; read last_year < USHCN.last_year
if [[ $skip_US -eq 0 ]]
then cp input_files/Ts.strange.RSU.list.IN_full temp_files/Ts.strange.RSU.list.IN ; fi

echo "finding offset caused by adjustments" ; echo "extracting US data from GHCN set"
grep ^425 v2.meany > ghcn_us
dump_old.exe ghcn_us ghcn_us_end 1980
echo "getting USHCN data:"
grep -v -e -9999-9999-9999-9999-9999-9999-9999-9999-9999-9999-9999-9999  USHCN.v2.mean_noFIL > xxx
dump_old.exe xxx yyy 1880  ; rm -f xxx
sort -n yyy > USHCN.v2.mean_noFIL ; rm -f yyy
${fortran_compile} dif.ushcn_ghcn.f -o dif.ushcn_ghcn.exe
dif.ushcn_ghcn.exe ${last_year}  ;  echo "created ushcn-ghcn_offset_noFIL "
${fortran_compile} cmb2.ushcn.v2.f -o cmb2.ushcn.v2.exe
cmb2.ushcn.v2.exe $skip_US > ushcn.log   ;  echo "created  v2.meanz"

echo "replacing Hohenpeissenberg data in $1 by more complete data (priv.comm.)"
echo "disregard pre-1880 data:"
tail +100 input_files/t_hohenpeissenberg_200306.txt_as_received_July17_2003 > t_hohenpeissenberg
${fortran_compile} hohp_to_v2.f -o hohp_to_v2.exe
${fortran_compile} cmb.hohemp.v2.f -o cmb.hohemp.v2.exe
hohp_to_v2.exe ;  cmb.hohemp.v2.exe

# cleanup
mv   v2.mean temp_files/v2.mean_comb
mv -f *.exe v2.meanz* ghcn* ushcn* hcn* USHCN.v2.mean_noFIL ID_US_G *hohenpeiss* work_files/. 2> /dev/null

echo ; echo "created v2.mean_comb"
echo "move files from temp_files/. to .../STEP1/temp_files/. "
echo "and execute in the STEP1 directory the command:"
echo "   do_comb_step1.sh v2.mean_comb"
```

```
def step0(input):
    """An iterator for Step 0.  Produces a stream of
    'giss_data.Series' instances.  *input* should be an object that has
    'ushcn', 'ghcn', 'scar', 'hohenpeissenberg' properties, each one
    being a data source (typically, this input object is made by the
    tool.giss_io.step0_input() function).

    """

    # Read each data input into dictionary form.
    data = {}
    for source in ['ghcn', 'ushcn', 'scar']:
        if source in parameters.data_sources:
            print "Load %s records" % source.upper()
            data[source] = dsdict(getattr(input, source))

    # If we're using GHCN (and we usually are) then we...
    if 'ghcn' in parameters.data_sources:
        # ... adjust the Hohenpeissenberg data;
        if 'hohenpeissenberg' in parameters.data_sources:
            correct_Hohenpeissenberg(data['ghcn'], input.hohenpeissenberg)
        # ... adjust the USHCN data;
        if 'ushcn' in parameters.data_sources:
            adjust_USHCN(data['ushcn'], data['ghcn'])
        # ... optionally exclude contiguous US stations.
        if not parameters.retain_contiguous_US:
            discard_contig_us(data['ghcn'])

    # Join all data sources together.
    records = {}
    for source in ['ghcn', 'ushcn', 'scar']:
        records.update(data.get(source, {}))

    # We sort here - as does GISTEMP - so that all the records for a
    # given 11-digit station ID are grouped together, ready for
    # combining in the next step.
    for _, record in sorted(records.iteritems()):
        if record:
            yield record
```



Code Size (lines)



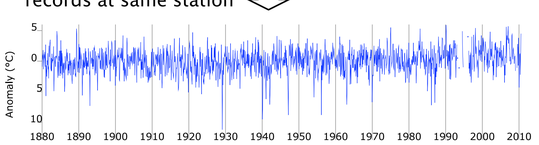


Land station analysis
Temperature anomaly
2010-07

grid
interpolate

ccc-gistemp combines records at same station

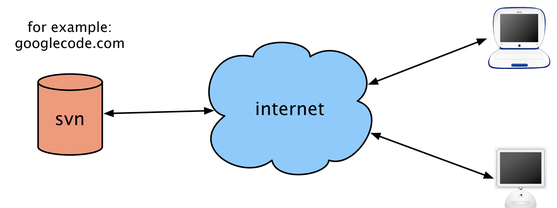
(displayed as anomalies)



Open Climate Code

a CCF project

Distributed open source collaboration is hugely successful for software tools: making better code, quicker. Open Climate Code promotes open source collaboration in the science community.



Open Source, Open Access, Creative Commons

Do you publish your source code?

Do you publish in Open Access journals?

Can we help you be more open?

The Climate Code Foundation is a company limited by guarantee, incorporated in England and Wales on 16th August 2010, company number 7346328.

All content available under CC-BY license: http://creativecommons.org/licenses/by/3.0/
Station Data: GHCN / SCAR READER / USHCN; Map Data: ISLSCP II; Code: GISS / Clear Climate Code