

Начало работы с GMT

[Обсудить в форуме](#) Комментариев — 5

Эта страница опубликована в основном списке статей сайта по адресу <http://gis-lab.info/qa/gmt.html>

Описание работы с GMT на конкретном примере.

Введение

GMT (Generic Mapping Tools) — набор из 60-и консольных инструментов, направленных на обработку географических данных и на создание высококачественных Encapsulated PostScript (EPS) иллюстраций этих данных от простых x-y графиков, до искусственно освещенных карт рельефа и 3-х мерных изображений моделей поверхностей. Начиная в 1988 году с нескольких несложных картографических программ в дипломной работе двух студентов, сейчас GMT это мощная, полнофункциональная ГИС широко распространенная по всему миру в научной сфере. GMT поддерживает около 30 проекций, имеет данные о береговых линиях континентов, рек и политических границах, которые использует для построения карт. Несмотря на внушительный набор возможностей по обработке данных, основная цель GMT — это уменьшение количества времени, затрачиваемых на подготовку высококачественных иллюстраций для публикаций в научных журналах, проектах или слайдов для презентаций.

Основным отличием GMT от большинства ГИС является отсутствие графического интерфейса, что многим кажется большим неудобством. Однако, при большом объеме обрабатываемых данных и большом количестве карт, которые требуется получить на выходе, это является скорее достоинством. Можно легко написать скрипт, который сам будет извлекать требуемую порцию данных, соответствующим образом их обрабатывать, и оформлять все это дело в карту.

Почти все GMT-инструменты посылают на стандартный выход код на языке Postscript. Этот код является каким либо элементом карты. Стандартный выход мы можем просто перенести и записать в нужный нам файл или приписать к уже существующему ("> file" или ">> file" после вызова утилиты). Этим достигается необычайная гибкость - одну карту мы рисуем последовательно вызывая утилиты, каждая из которых добавляет в файл свою порцию данных (рамка, изолинии, маршруты, точки, надписи, масштабная линейка, легенда и т.п.). По этой причине, последовательность вызовов утилит принято оформлять в shell-скрипт, незначительно отредактировав который мы можем перерисовать карту, или нарисовать новую.

Пример

Рассмотрим процесс оформления карты подробнее.

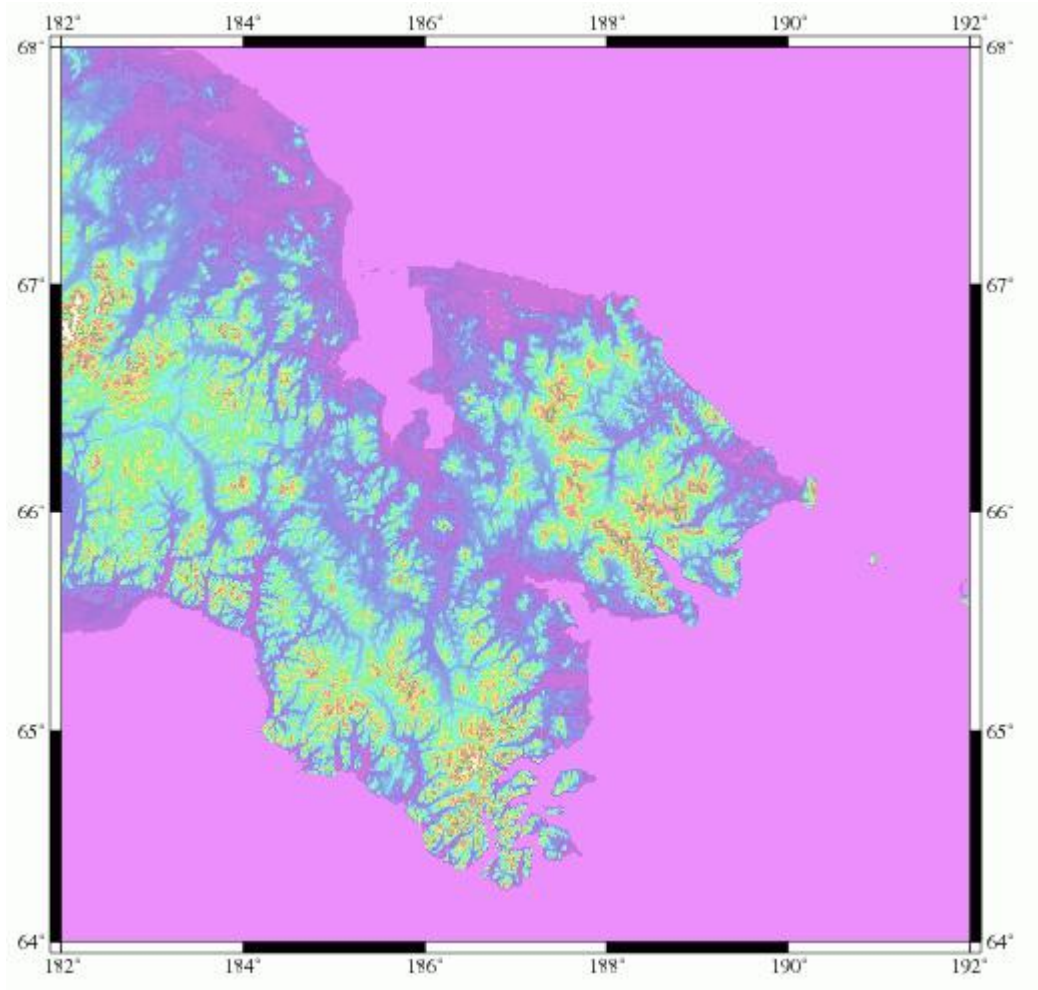
Допустим, мы имеем грид ЦМР на Чукотский полуостров ([скачать](#)). Чтобы отобразить его с помощью GMT, создадим вот такой скрипт:

```
#!/bin/bash
makecpt -Cтопо -T1/1300/1 > map.cpt
grdimage elev.grd -R-178/-168/64/68 -JM16c -Cmap.cpt -K > map.ps
psbasemap -R -J -B2/1 -O >> map.ps
```

Сохраним его под именем gmt.sh. Для запуска скрипта откроем терминал (консоль) и перейдем в папку где он хранится. Разрешим выполнение файла скрипта набрав: `chmod u+x gmt.sh` Теперь мы можем его запустить: `./gmt.sh`

После запуска скрипта, в его каталоге появиться наша карта - файл map.ps. Его можно просмотреть с помощью

любого просмотрщика, поддерживающего формат PostScript. Разработчики рекомендуют использовать ghostview.



Посмотрим на работу скрипта шаг за шагом.

Строка 1 - стандартное начало любого Unix-скрипта - путь к программе-интерпретатору.

Строка 2- создаем цветовую палитру (файл map.cpt) для карты командой makecpt.

-Ctopo

-T1/1300/1

Имя палитры (одной из 20 палитр, имеющихся в GMT) которую мы возьмем за основу

Интервал значений, для которого следует создавать таблицу. Интервал значений грида можно узнать командой grdinfo.

Строка 3 - отрисовка грида командой **grdimage**. С этой командой необходимо использовать большое количество флагов в которых с первого раза легко запутаться (для GMT это нормально). Разберем их подробнее:

| | | | | |
|-------------------------|---|------------------------------------|---------------------------------------|--|
| elev.grd | -R-178/-168/64/68 | -JM16c | -Cmap.cpt | -K |
| имя грида для отрисовки | Размер региона (minx/maxx/miny/maxy), который будет отрисовываться (может | Проекция Меркатор. Ширина экватора | Использовать цветовую палитру map.cpt | После текущей команды ожидается следующая. |

| | |
|--|---|
| быть больше или меньше размеров рисуемого грида) | нарисованной карты составит 16 сантиметров. |
|--|---|

Теперь назначение флагов стало более или менее понятно, кроме последнего -K. Этот флаг отсекает добавление в файл нашей карты финализирующей части postscript кода, для того чтобы следующие gmt-инструменты могли добавить к карте новые детали.

Строка 4 - создание рамки и координатной сетки командой **psbasemap**. Описание использованных флагов:

| -R | -J | -B2/1 | -O |
|---|--|--|---------------------------------|
| Использовать тот же размер, что и при вызове предыдущей команды | Использовать ту же проекцию, что и при вызове предыдущей команды | Расставить подписи координатной сетки на оси X через 2 градуса, на оси Y через 1 градус. | Добавить к существующему файлу. |

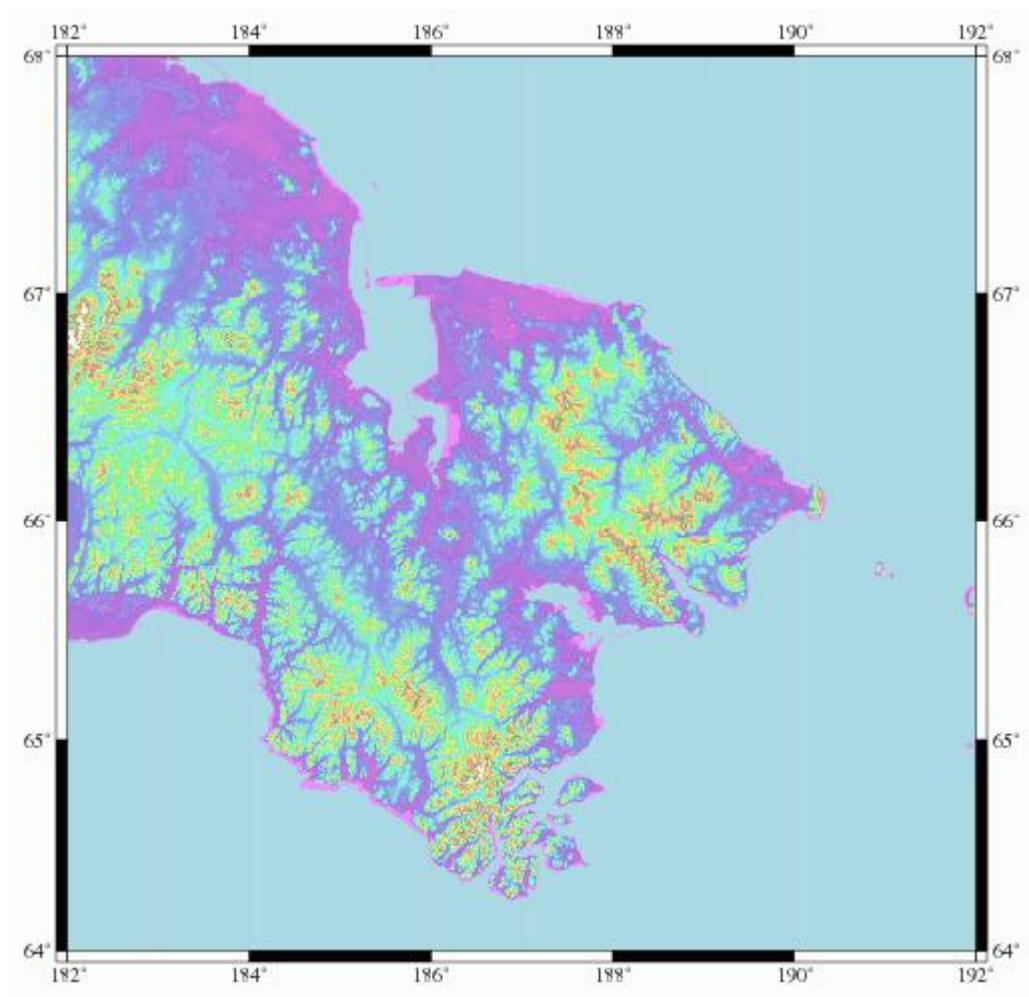
Знакомые нам флаги -R и -J, задающие регион и проекцию на этот раз используются без параметров. GMT догадается использовать такие же параметры региона и проекции как и при выполнении прошлой команды. Флаг -O аналогичен флагу -K, но действует наоборот - отсекает вступительную часть postscript кода, для того чтобы корректно добавить новый фрагмент карты в уже существующий файл. Далее по тексту, в таблицах объяснений параметров, объяснения параметров -J, -R, -O, -K пропускается.

Как видим, пока ничего сложного. Только вот на карте отсутствует береговая линия, и океан. Нарисуем океан вставив перед **psbasemap** следующую строку:

```
pscoast -R -J -Slightblue -O -K -Df >> map.ps
```

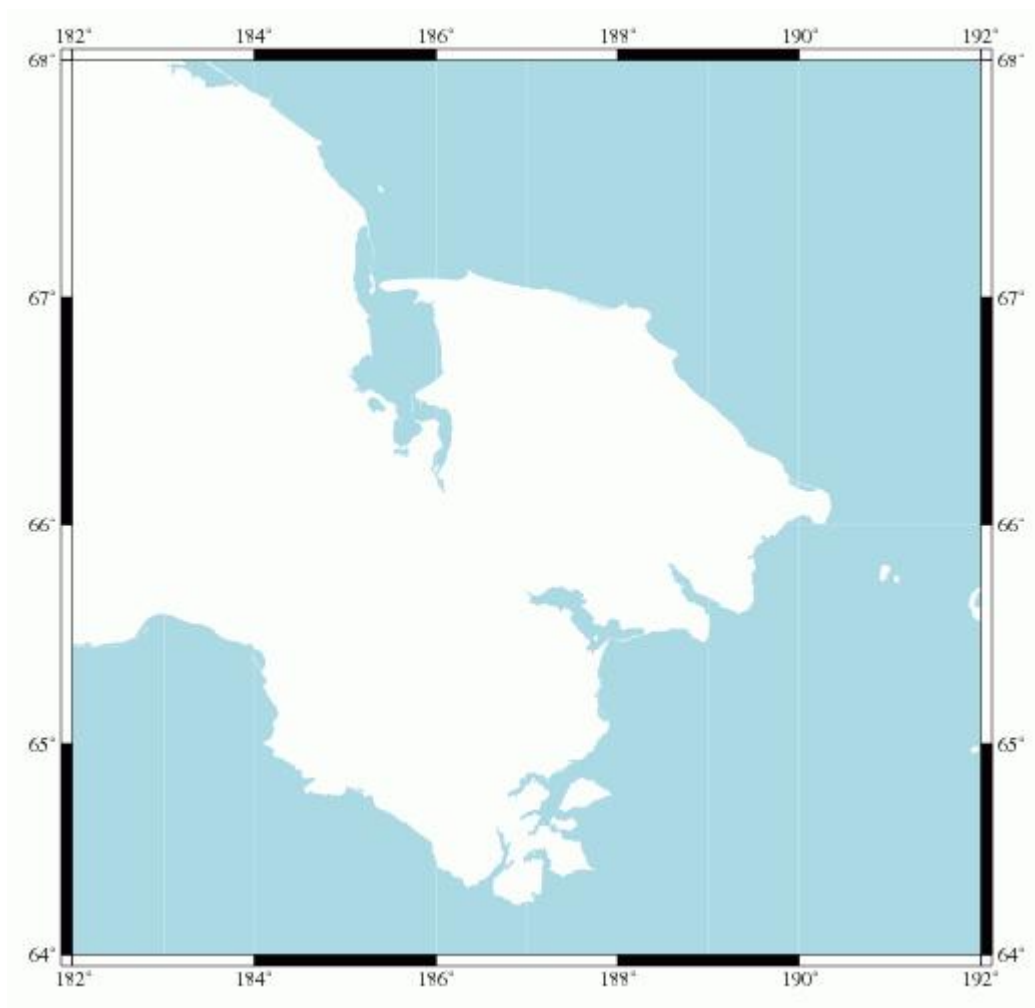
| -Slightblue | -Df |
|--|---|
| Закрасить моря и океаны светло-серым цветом. | Использовать высокое разрешение для данные о береговой линии. |

Pscoast рисует не только моря или океаны, но и континент (если задана опция -G) и береговую линию (опцией -W) и гидросеть (опцией -I) и даже политические границы (опцией -N). Мы задали лишь опцию -S, в результате океан залит светло-серым цветом и наша карта выглядит уже вполне сносно.



Поэкспериментируем с результатом. Чтобы не загружать процессор понапрасну, будем отрисовывать на карте только береговую линию, закомментировав третью строку и слегка изменив четвертую:

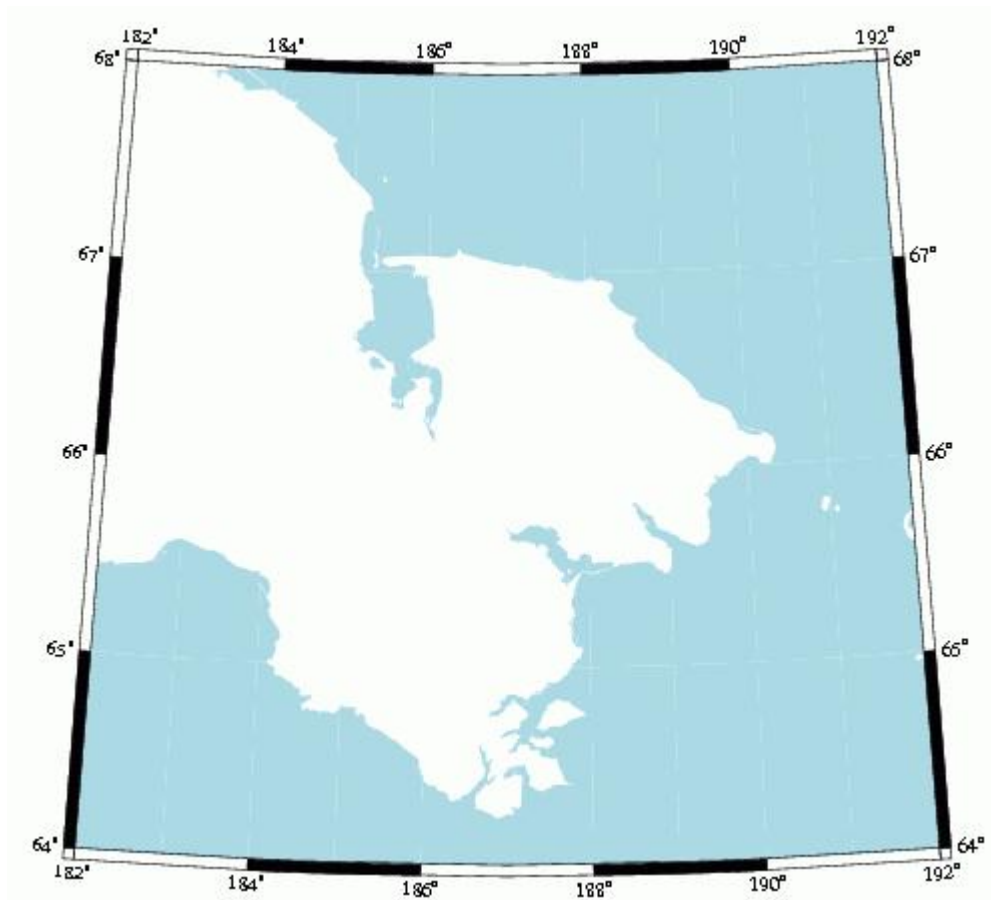
```
#!/bin/bash
makecpt -Ctopo -T1/1300/1 > map.cpt
#grdimage elev.grd -R-178/-168/64/68 -JM16c -Cmap.cpt -K > map.ps
pscoast -R-178/-168/64/68 -JM16c -Slightblue -K -Df > map.ps
psbasemap -R -J -B2/1 -O >> map.ps
```



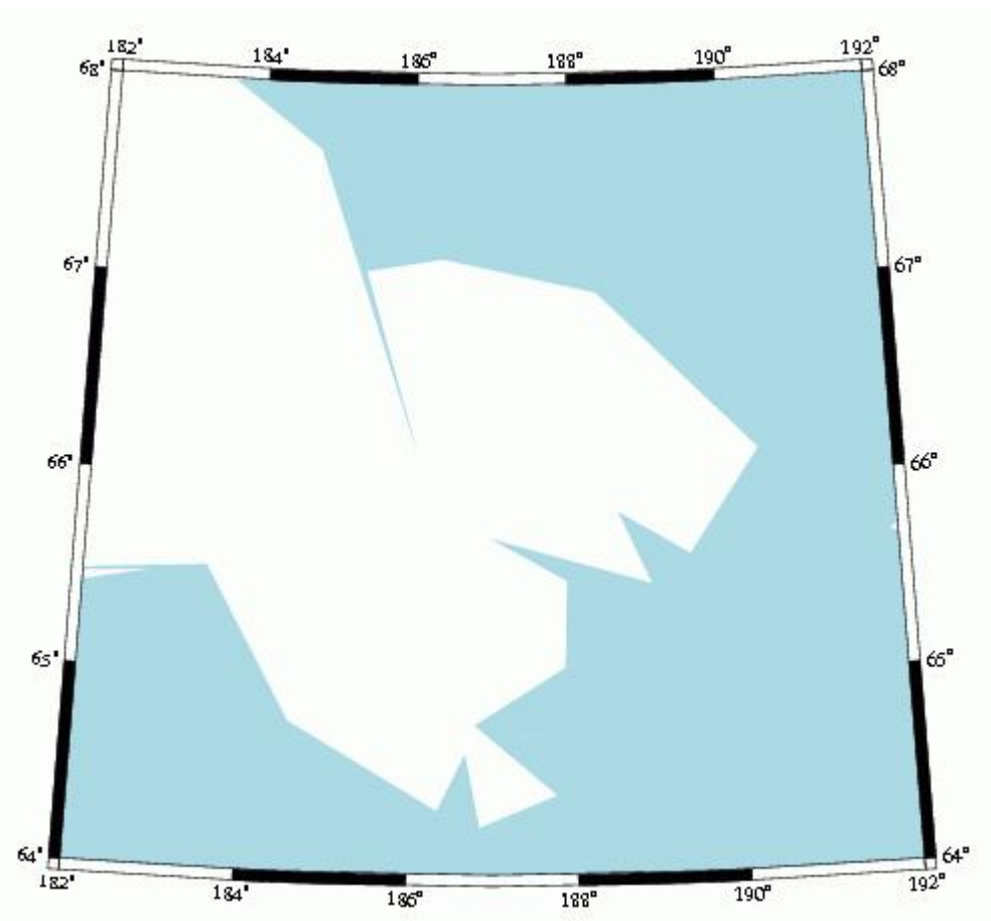
Попробуем в третьей строчке вместо **-JM16с** написать **-JM16i** изменив тем самым ширину экватора с 16 сантиметров на 16 дюймов. Запустим скрипт и увидим, что в нашем файле уместился только край карты.



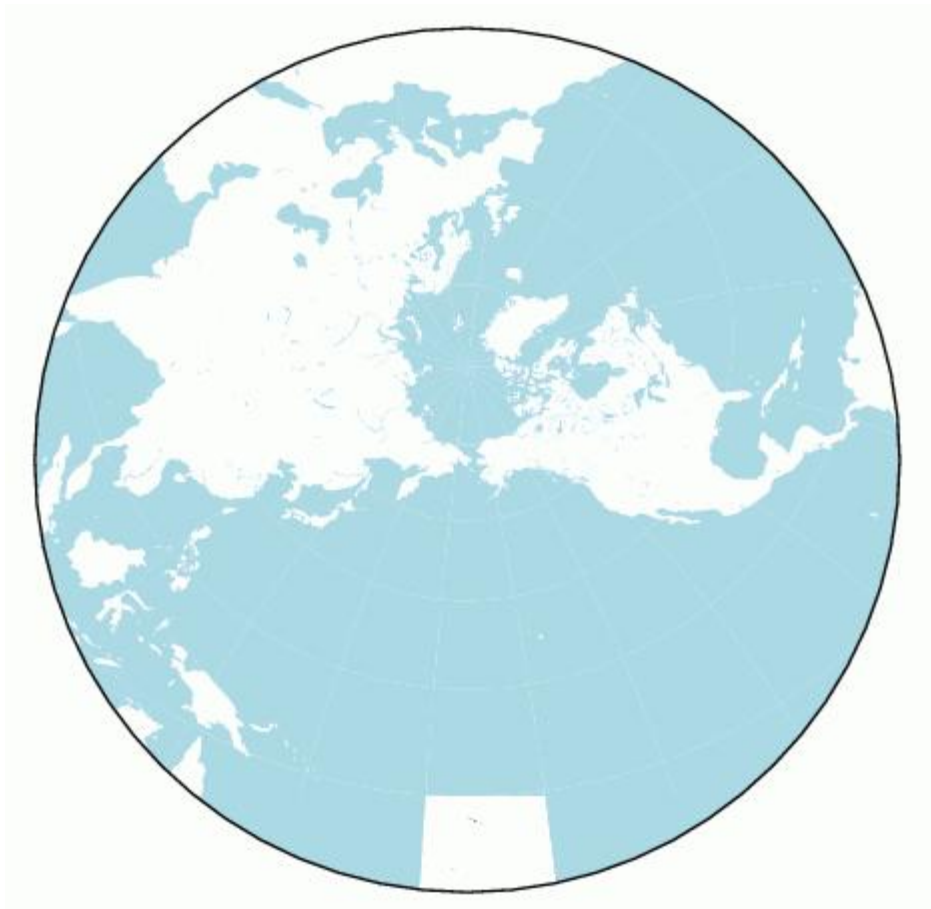
Вернуть все назад можно написав **-JM6с**, и размер карты будет совпадать с размером 16-и сантиметровой карты. От цилиндрической проекции перейдем к конической написав **-JB-173/66/64/68/6i** (Albers projection - JBlon0/lat0/lat1/lat2/width).



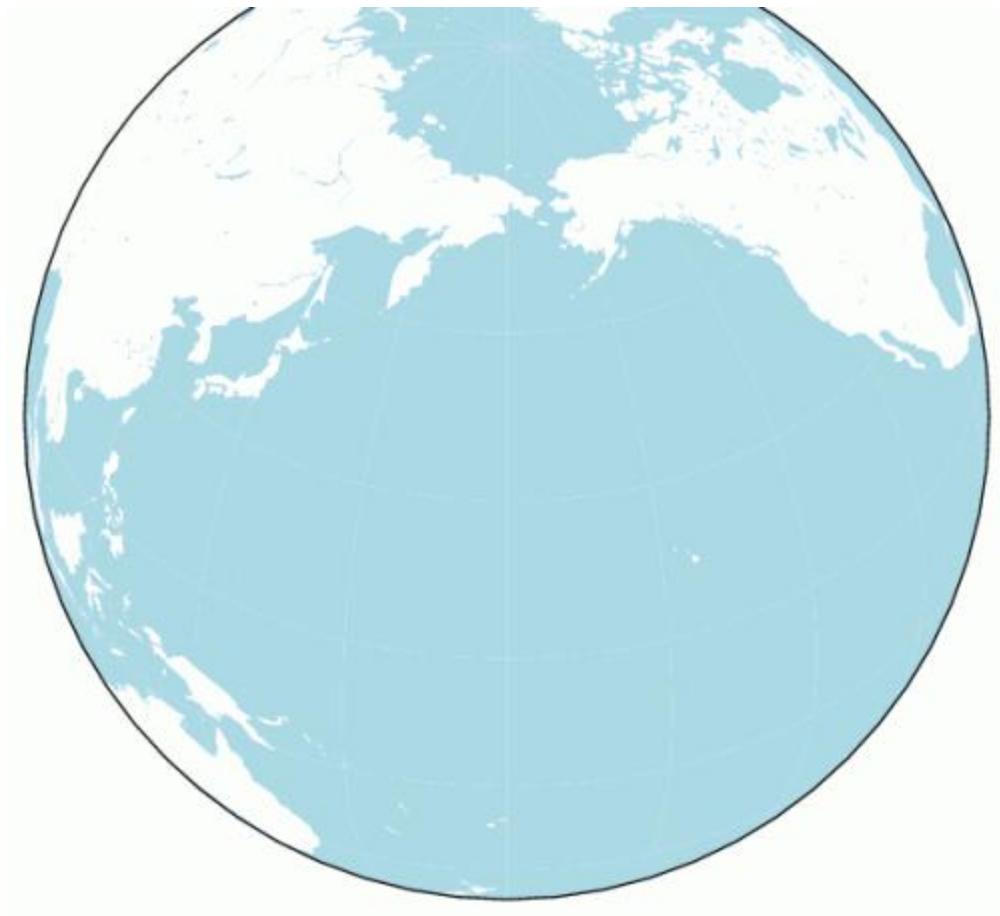
Уменьшим разрешение береговой линии с полного (full) до минимального (crude) изменив параметр -D команды pscoast с **-Df** на **-Dc**.



Граница сильно генерализована, но для крупномасштабной карты такая степень генерализации в самый раз. Азимутальная проекция **-JS-173/66/6i** (General Stereographic - JSlon0/lat0/width).

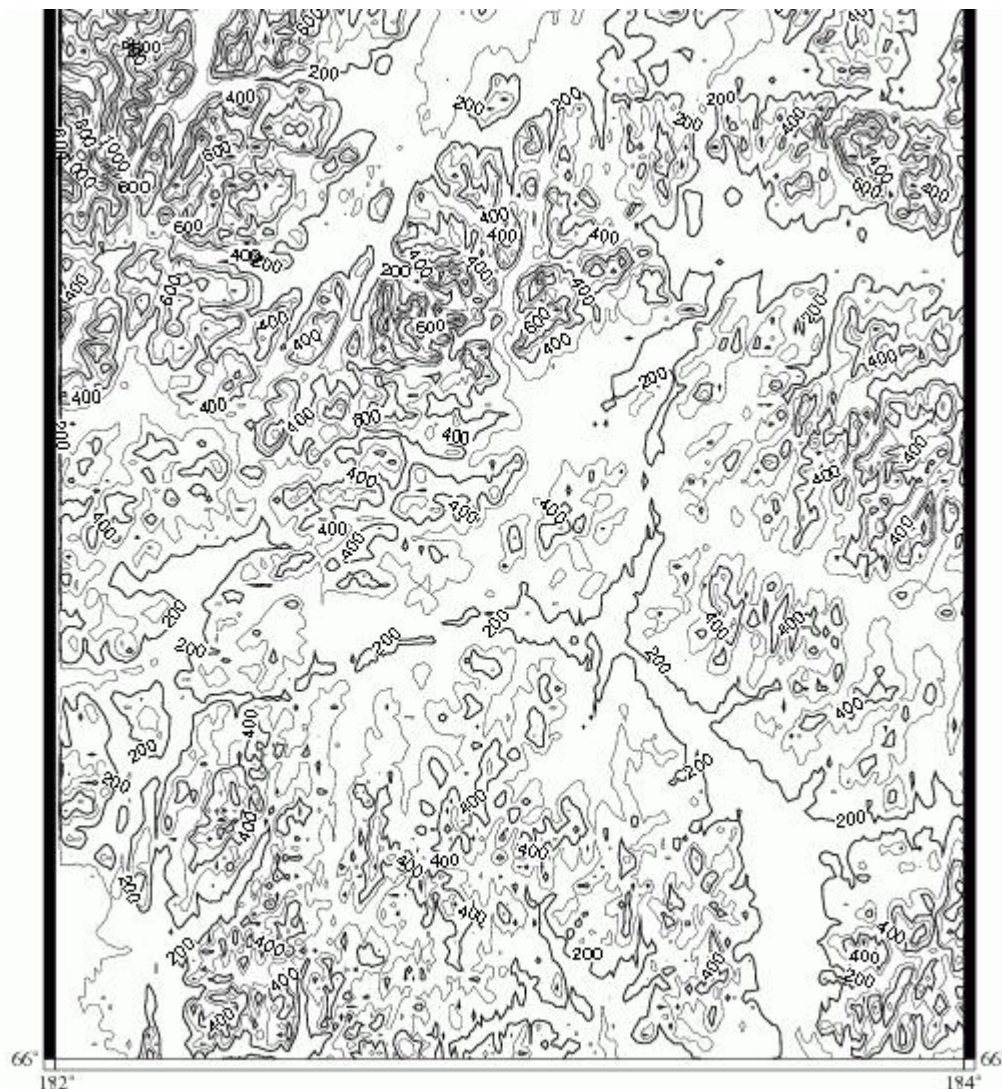


Изменим проекцию, а заодно и размеры региона с -R-180/180/-90/90 на -JG-180/40/20с.



Установим следующие параметры: -R-178/-176/66/67-JM16с, и перед **psbasemap** вставим строку:

```
grdcontour elev.grd -R -J -C100 -A200+kblack -O -K >> map.ps
```



Команда `grdcontour` отображает изолинии поверхности грида и имеет следующие параметры:

elev.grd

-C100

-A200+kblack

грид, для которого
надо нарисовать
изолинии

Интервал изолиний
100 м.

Через каждые 200 м будет
нарисована сплошная изолиния,
и подписана шрифтом черного
цвета.

Вернем наш скрипт к первоначальному варианту и изменим проекцию на цилиндрическую:

```
#!/bin/bash
makecpt -Ctopo -T1/1300/1 > map.cpt
grdimage elev.grd -R-178/-168/64/68 -JB-173/66/64/68/6i -Cmap.cpt -K > map.ps
pscoast -R -J -Slightblue -O -K -Df >> map.ps
psbasemap -R -J -B2/1 -O >> map.ps
```

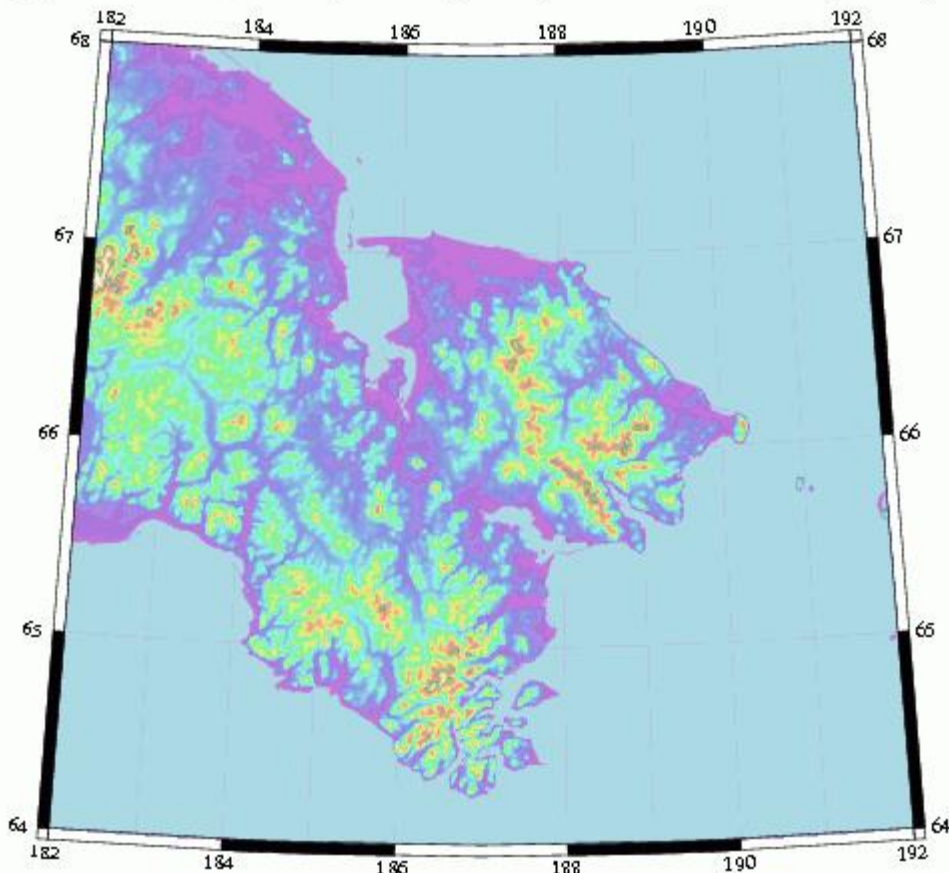
Неплохо было как-нибудь бы озаглавить нашу карту. Чуть-чуть исправим строчку `psbasemap`:

```
psbasemap -R -J -B2/1:."Цифровая модель рельефа Чукотского полуострова": -O -V >>
map.ps
```

Для нормального отображения кириллицы в ps-файле, файл нашего скрипта нужно сохранить в кодировке ISO-8859-5, а в начале скрипта, перед вызовом остальных `gmt` инструментов, добавить сразу после `#!/bin/bash`:

```
gmtset CHAR_ENCODING ISO-8859-5
```


Цифровая модель рельефа Чукотского полуострова



Команда **gmtset** устанавливает глобальные параметры для GMT, хранящиеся в файле `.gmtdefaults4` в директории `$GMTHOME` (каталоге установки GMT). Перед тем как производить какие-либо изменения в этом файле, `gmtset` копирует его в текущую директорию, и изменяет копию. Все последующие `gmt`-инструменты обнаружив `.gmtdefaults4` в текущей директории, будут брать настройки из него. Поэтому, для того чтобы не возникало недоразумений (особенно, когда в одной директории хранится сразу несколько скриптов), перед завершением скрипта лучше удалить этот файл, а заодно и `.gmtcommands`, в котором хранится история введенных параметров, общих для всех `gmt`-инструментов (-R, -J, -P, и др.):

```
rm .gmt*
```

Примечание: К сожалению, поддержка русского языка в GMT слабая. Если вы запустите этот скрипт, то в терминале увидите сообщения **psbasemap** о том что символы для обозначения градусов, минут и секунд отсутствуют, и поэтому он вставит на их место пробелы. Также, если вы конвертируете post-script в pdf (например командой `ps2pdf`) то, в зависимости от просмотрщика, все символы сползут со своих мест. Впрочем, Acrobat Reader 7.0 такие pdf-файлы отображает корректно.

Один из способов решения этой проблемы предложил [Михаил Чернышев](#), и заключается он в установке дополнительных кириллических шрифтов в `ghostscript`, и ссылки на них в `gmt`. После этого шрифты можно использовать в `gmt` под номерами 36 и больше. Однако, стоит отметить, что ps-файлы, полученные таким способом будут корректно отображаться только на компьютере с установленными кириллическими шрифтами в `ghostscript`.

Изменим шрифт заголовка с Times на Helvetica:

```
gmtset CHAR_ENCODING ISO-8859-5 HEADER_FONT 1 HEADER_FONT_SIZE 20
```

Цифровая модель рельефа Чукот



Теперь, добавим поселки и их названия. Отрисовка векторных элементов в GMT осуществляется командой **psxy** для двухмерной карты и **psxyz** для псевдо-трехмерного изображения. Эти команды визуализируют векторные объёты, передаваемые им в формате .xy (или .xyz), который является простым текстовым файлом, в каждой строке которого содержится координаты x и y (и z, в формате .xyz). Например:

```
-173.19806733    64.45356668
-172.84372109    64.50418757
-172.26769026    64.41341907
-171.72657038    65.50962321
-171.01264126    65.58991841
-169.79599432    66.15896705
-171.89763408    66.96541022
-173.00954815    67.03523214
-174.95932524    67.42972598
-175.84780916    67.83294757
-175.82511704    65.00515984
-175.41316772    64.80442182
```

Сохраним список в файл points.xy. Следующая команда нарисует на месте поселков белые кружочки.

```
psxy points.xy -J -R -Sc0.14 -W2black -Gwhite -O -K -V >> map.ps
```

points.xy

-Sc0.14

-W2black

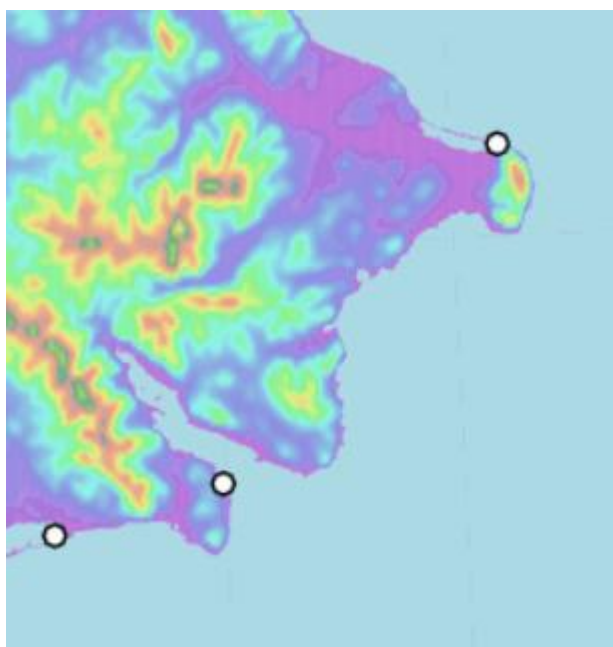
-Gwhite

xy-файл, содержащий
координаты поселков

Рисуем кружок
радиусом 0.14
см.

Кружок нарисует
карандашом
черного цвета и
толщиной 2

Зальем кружок
белым цветом



Чтобы напротив каждого поселка написать его название воспользуемся командой **pstext**. Pstext размещает на карте в заданных местах надписи, определенного размера, и начертания, построчно записанные в простом текстовом файле. В каждой строке такого файла написаны x-y координаты (как в .xy файле), размер надписи, наклон, номер ps-шрифта, выравнивание по вертикали (M-по середине, T-по верху, B-по низу) и выравнивание по горизонтали (L-лево, C-центр, R-право), и, собственно, надпись. Вот как такой файл будет выглядеть для надписей названий поселков ([скачать](#)):

```
-173.19806733    64.45356668    8 0 2 TR Провидения
-172.84372109    64.50418757    8 0 2 BL Нов. Чаплино
-172.26769026    64.41341907    8 0 2 BL Чаплино
-171.72657038    65.50962321    8 0 2 TL Лорино
```

| | | |
|---------------|-------------|----------------------|
| -171.01264126 | 65.58991841 | 8 0 2 TL Лаврентия |
| -169.79599432 | 66.15896705 | 8 0 2 BL Уэлен |
| -171.89763408 | 66.96541022 | 8 0 2 BL Энмурино |
| -173.00954815 | 67.03523214 | 8 0 2 BL Нешкан |
| -174.95932524 | 67.42972598 | 8 0 2 BL Нутэпельмен |
| -175.84780916 | 67.83294757 | 8 0 2 BL Ванкарем |
| -175.82511704 | 65.00515984 | 8 0 2 TR Энмелен |
| -175.41316772 | 64.80442182 | 8 0 2 TR Нунлинггран |

По сути дела, это дополненный файл points.xy. Мы смело можем удалить его, а на его месте сохранить наш новый файл. **psxy** после этого также будет рисовать кружочки, а **pstext** из этого же файла надписи к ним:

```
pstext points.xy -J -R -Dj0.1c/0.1c -Gblack -K -O >> map.ps
```

points.xy

-Dj0.1c/0.1c

-Gblack

файл, содержащий
координаты поселков
и их названия

Смещение координат каждой
надписи для того чтобы надписи
не налезали на символы
поселков.

Поселки
подписывать
шрифтом черного
цвета

Надписи заливов и морей сохраним в файле [names.xy](#):

| | | |
|--------|------|----------------------------|
| -171 | 67.5 | 14 -10 0 MC Чукотское море |
| -176.5 | 64.5 | 14 0 0 MC Анадырский залив |
| -168.8 | 66 | 10 80 0 MC Берингов Пролив |
| -171.6 | 65.3 | 8 -10 0 MC Мечигменский |
| -171.6 | 65.2 | 8 -10 0 MC залив |
| -174.2 | 66.8 | 8 -10 0 MC Колючинская |
| -174.2 | 66.7 | 8 -10 0 MC губа |

И отобразим их командой:

```
pstext names.xy -J -R -Gblack -K -O >> map.ps
```



Теперь, неплохо было бы добавить к карте шкалу высот рельефа.

```
psscale -Cmap.cpt -D8c/-2c/8c/0.3ch -B100:"Высота рельефа":/:"метры": -A -O -K >>
map.ps
```

-Cmap.cpt

-D8c/-2c/8c/0.3ch

**-B100:"Высота
рельефа":/:"метры":**

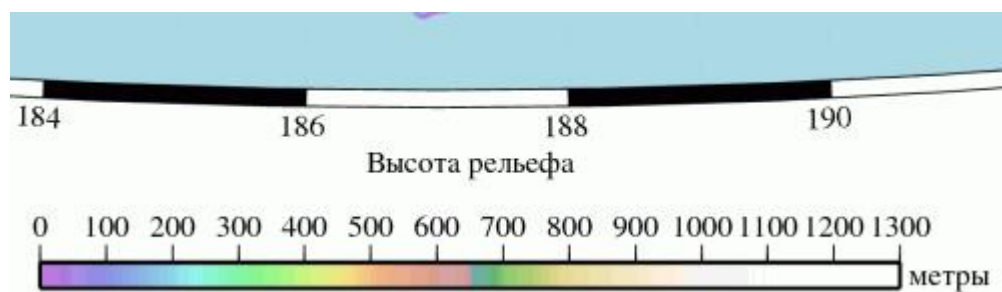
-A

Цветовая палитра, для
которой рисуется
шкала

Разместить шкалу на
странице с центром в
точке на 8 см правее и на
2см ниже левого
нижнего угла карты.
Параметр h рисует
горизонтальную шкалу
(по умолчанию она
вертикальная).

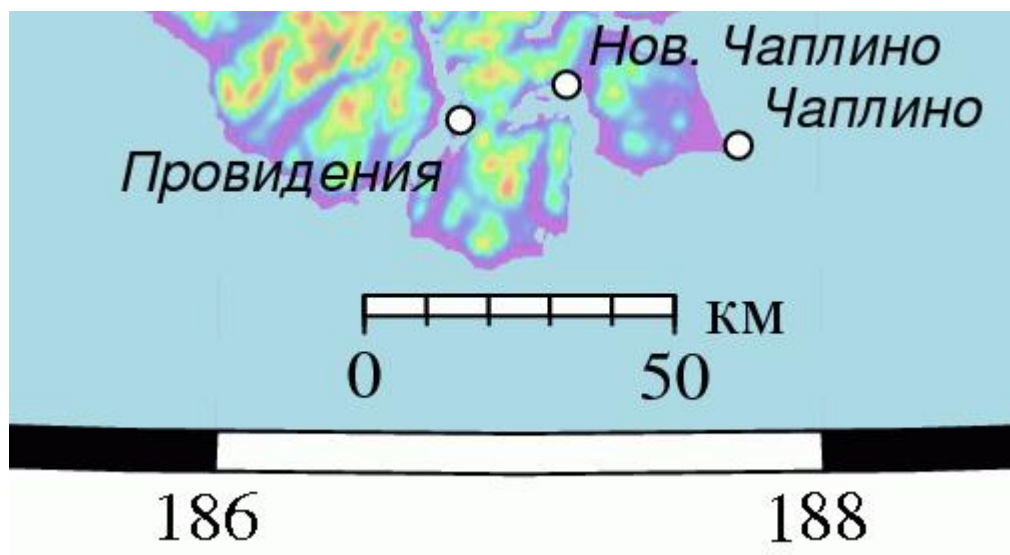
Установить деления
на шкале через 100 м
и сделать подпись
на шкале "Высота
рельефа", а на
делениях - "метры".

Подпись и
деления
поместить над
шкалой.



Чтобы нарисовать масштабную линейку к команде **psbasemap** нужно добавить флаг **-Lflat0/lon0/lon/lenght**. Мы добавим так: **-Lf-173/64.2/66/50:"км":r**. После этого на карте в точке с координатами 1173/64.2 отобразится масштабная линейка, отображающая горизонтальный масштаб для широты 66 градусов,

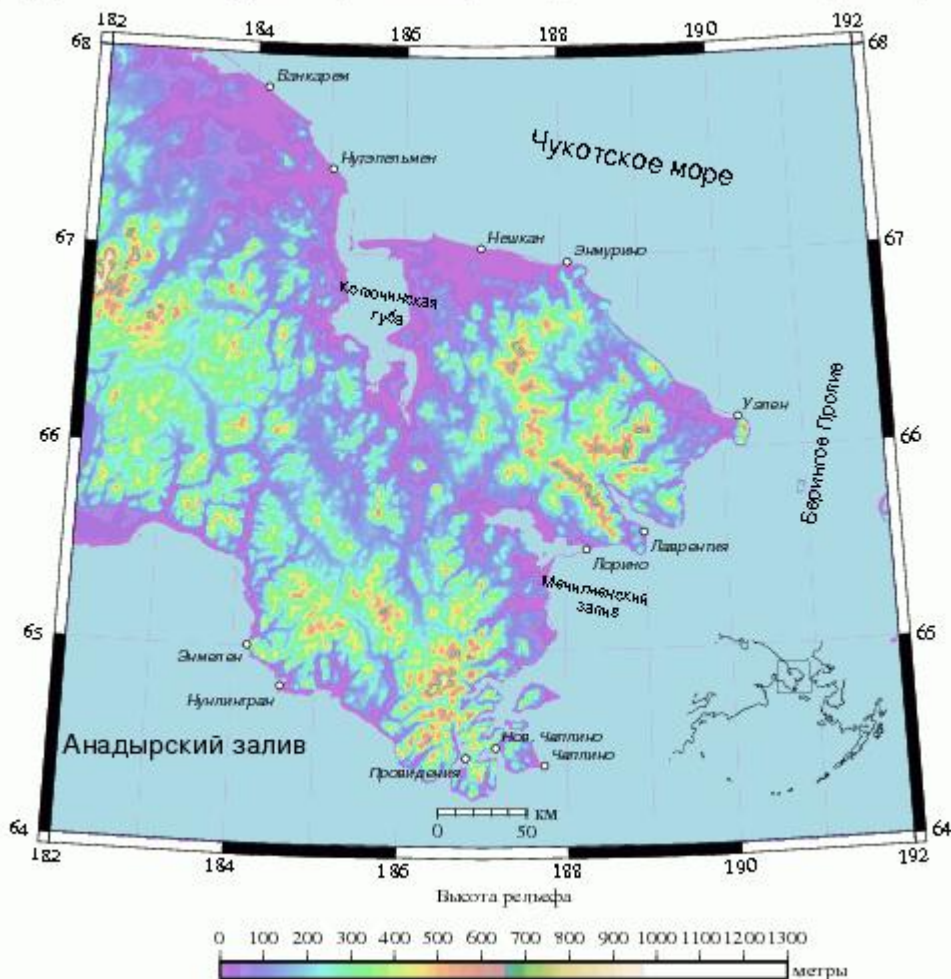
протяженностью в 50 км, с подписью "км", размещенной справа.



В завершение, в левом нижнем углу карты разместим схематичный фрагмент окрестностей мелкого масштаба (карту врезку), на котором выделим наш регион:

```
pscoast -R160/210/45/70 -JB-175/60/50/70/5c -O -K -Wthinnest,black -X10.5c -A400 >>
map.ps
psxy -R -J -L -Wthinnest,black -O <> map.ps
182 64
182 68
192 68
192 64
EOF
```

Цифровая модель рельефа Чукотского полуострова



Параметры команды **pscoast**:

-R160/210/45/70 -JB-175/60/50/70/5c -Wthinest,black -X10.5c

-A400

| | | | | |
|----------------|--|--------------------------------|-----------------------------------|---|
| размер региона | Коническая проекция, и ширина карт по горизонтали - 5 см | Береговая линия черная, тонкая | сместить карту вправо на 10,5 см. | Упростить береговую линию, отсечением элементов, площадь которых меньше 400 м2. |
|----------------|--|--------------------------------|-----------------------------------|---|

Границы региона рисуем командой **psxy**:

-L -Wthinest,black <<EOF

| | | |
|--------------------------|----------------------|--|
| Замкнуть рисуемую кривую | Линия черная, тонкая | координаты кривой находятся не в файле, а следуют после текущей команды, и завершаются строкой 'EOF' |
|--------------------------|----------------------|--|

Содержание результирующего скрипта ([скачать скрипт](#)):

```
#!/bin/bash
gmtset CHAR_ENCODING ISO-8859-5 HEADER_FONT 1 HEADER_FONT_SIZE 20
makecpt -Ctopo -T0/1300/1 > map.cpt
grdimage elev.grd -R-178/-168/64/68 -JB-173/66/64/68/6i -Cmap.cpt -K > map.ps
pscoast -R -J -Slightblue -O -K -Df >> map.ps
psxy points.xy -J -R -Sc0.14 -W2black -Gwhite -O -K -V >> map.ps
pstext points.xy -J -R -Dj0.1c/0.1c -Gblack -K -O >> map.ps
pstext names.xy -J -R -Gblack -K -O >> map.ps
psscale -Cmap.cpt -D8c/-2c/10c/0.3ch -B100:"Высота рельефа":/:"метры": -A -O -K >>
map.ps
psbasemap -R -J -B2/1:."Цифровая модель рельефа Чукотского полуострова": -Lf-
173/64.2/66/50:"км":r -O >> map.ps
pscoast -R160/210/45/70 -JB-175/60/50/70/5c -O -K -Wthinest,black -X10.5c -A400 >>
map.ps
psxy -R -J -L -Wthinest,black -O <> map.ps
182 64
182 68
192 68
192 64
EOF

rm .gmt*
```

Заключение

Как видно, с помощью GMT несложно создавать качественные карты. Простота внутреннего формата данных, а также большое разнообразие поддерживаемых форматов, позволяет использовать GMT в связке с любой ГИС и в любом проекте. А широкий выбор предустановленных параметров позволяет отвлечься от дизайнерской рутины, и сосредоточиться на более важных аспектах работы.

GMT распространяется под лицензией GPL, ее можно свободно скачать с [официального сайта](#). На сайте представлена удобная форма, позволяющая относительно легко получить исходный код и скомпилировать его на компьютере с установленной ОС Linux. Для установки GMT можно ознакомиться с [небольшой инструкцией](#) на русском языке. GMT есть в репозиториях многих дистрибутивов, однако установка программы из них сопровождается некоторыми трудностями. Возможное их решение описано [здесь](#).

Данное описание проверено на работоспособность под UNIX. Разработчики утверждают, что программу можно использовать также и под Windows. На официальном ftp-сервере есть уже скомпилированная версия для win-32. Однако, GMT зависит от ряда классических UNIX утилит (например, awk), так что проще будет установить cygwin, и запустить GMT в нем.

[Обсудить в форуме](#) Комментариев — 5

Последнее обновление: 2014-05-15 01:27

Дата создания: 04.04.2008

Автор(ы): [Михаил Кондратьев](#)