Useful tool: annot-regs

Common task:

- find overlaps in two sets of genomic regions (e.g. two CNV callsets)
- .. while matching records by a common column (e.g. sample name)
- .. and transferring one or more fields from one file to another for matching records

Requirements:

- columns can be named differently and can appear in arbitrary order
- headers may or may not be present
- can require multiple columns to match (e.g. sample name AND variant type)
- can transfer multiple fields

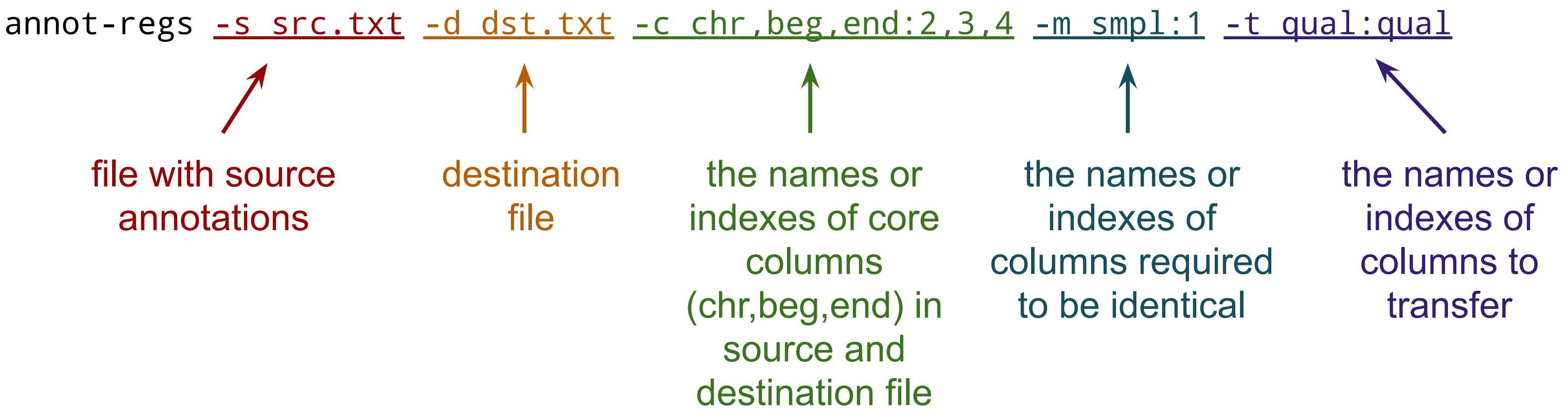
Simple usage

Source file:

```
#chr    beg       end         smpl      type   qual    length  npr
  5  67591246  67591246  DDDP167799  DEL    3.91  19611039  4
  7  42006018  42006018  DDDP154344  DUP   64.02  19580982  29
 12  14825806  14825807  DDDP123456  DEL    7.25     66506  2
 16  30978217  30978217  DDDP345139  DEL   55.76     42248  3
 19  42474652  42474652  DDDP114567  DUP   63.27  19581359  28
```

Destination file:

```
DDDP345139  16  30978217  30978217
DDDP123456  12  14825806  14825807
DDDP154344   7  42006018  42006018
DDDP114567  19  42474652  42474652
DDDP167799   5  67591246  67591246
```



```
annot-regs -s src.txt -d dst.txt -c chr,beg,end:2,3,4 -m smpl:1 -t qual:qual
```

file with source annotations

destination file

the names or indexes of core columns (chr,beg,end) in source and destination file

the names or indexes of columns required to be identical

the names or indexes of columns to transfer

Note: this works also for files with a list of positions, not just regions, just use `-c chr,pos,pos`.

Usage page

```
About: Annotate regions in DST file with texts from overlapping regions in SRC file.
        The transfer of annotations can be conditioned on matching values in one or more
        columns (-m), multiple columns can be transferred (-t).
        In addition to column transfer and adding special annotations, the program can simply
        print (when neither -t nor -a is given) or drop (-x) matching lines.
        All indexes and coordinates are 1-based and inclusive.
Usage: annot-regs [OPTIONS] DST
Options:
        --allow-dups              Add annotations multiple times
   -a, --annotate list            Add special annotations:
                                          cnt  .. number of overlapping regions
                                          frac .. fraction of the destination region with an overlap
                                          nbp  .. number of source base pairs in the overlap
   -c, --core src:dst             Core columns [chr,beg,end:chr,beg,end]
   -d, --dst-file file            Destination file
   -H, --ignore-headers           Use numeric indexes, ignore the headers completely
   -m, --match src:dst            Require match in these columns
        --max-annots int          Adding at most int annotations per column to save time in big regions
   -o, --overlap float            Minimum required overlap (non-reciprocal, unless -r is given)
   -r, --reciprocal               Require reciprocal overlap
   -s, --src-file file            Source file
   -t, --transfer src:dst         Columns to transfer. If src column does not exist, interpret
                                  as the default value to use. If the dst column does not exist,
                                  a new column is created. If the dst column exists, its values are
                                  overwritten when overlap is found and left as is otherwise.
        --version                 Print version string and exit
   -x, --drop-overlaps            Drop overlapping regions (precludes -t)
```

Usage examples

```
# Header is present, match and transfer by column name
annot-regs -s src.txt.gz -d dst.txt.gz -c chr,beg,end:chr,beg,end -m type,sample:type,smpl -t tp/fp:tp/fp

# Header is not present, match and transfer by column index (1-based)
annot-regs -s src.txt.gz -d dst.txt.gz -c 1,2,3:1,2,3 -m 4,5:4,5 -t 6:6

# If the dst part is not given, the program assumes that the src:dst columns are identical
annot-regs -s src.txt.gz -d dst.txt.gz -c chr,beg,end -m type,sample -t tp/fp

# One of source or destination files can be streamed to stdin
gunzip -c src.txt.gz | annot-regs -d dst.txt.gz -c chr,beg,end -m type,sample -t tp/fp
gunzip -c dst.txt.gz | annot-regs -s src.txt.gz -c chr,beg,end -m type,sample -t tp/fp

# Print matching regions as above but without modifying the records
gunzip -c src.txt.gz | annot-regs -d dst.txt.gz -c chr,beg,end -m type,sample
```