Phrap is part of the phred/phrap package that is designed to analyze and assemble sequences from large genomes. It was developed by Phil Green and Brent Ewing at the Department of Molecular Biotechnology, Washington University, and is in the public domain for academic purposes. Phred processes trace data files, generating relevant information used by phrapt to assemble shotgun DNA sequences. Phrap is also designed for stand-alone use.

This guide doesn’t replace for the entire documentation for phrap, but can be used as reference for those who want to use the whole potential of the program. For additional information, please read the Phrap documentation.

Phrap command line

The Phrap command line has only one obligatory parameter: the FASTA file containing the sequences to be assembled, but there are several options that can be applied. A common Phrap command line is:

```
phrap seq_file -penalty -9 -ace > phrap.out
```

where seq_file is the FASTA file containing the sequences, “-penalty” a modifier of the penalty for base mismatch, “-ace” a modifier generating the file seq_file.ace and the standard output is redirected to phrap.out file.

Phrap also checks for a corresponding quality file suffixed with .qual (seq_file.qual in the example above). In the quality file, each position in the FASTA file corresponds to a quality value between 0 and 97 separated by spaces. The special quality values 98 and 99 are used in visual inspection (manual edition) to indicate inaccurate bases that must be ignored (98) and highly accurate bases, being used to break false joins made by Phrap (99). Every entry in a quality file must match an entry in the corresponding FASTA file.

The easiest way to generate both the FASTA and its corresponding quality file is with phd2fasta (available in the Consed package) over PHD files generated by Phred.

Phrap naming convention

In addition to the sequence and quality data, Phrap needs to know three things for each read:

(i) The subclone or other template from which the read is derived.
(ii) Read orientation (forward or reverse) within the subclone.
(iii) The chemistry used to generate the read.

This information is obtained from the sequence name (the string between the ‘>’ symbol and the first space in the header) using the St. Louis naming convention: the portion of the read name up to the first ‘>‘ (or whatever value used in subclone delim option – see below) identifies the subclone and the first letter following it indicates the orientation of the read within the subclone and its chemistry, as follows:

<table>
<thead>
<tr>
<th>Orientation</th>
<th>Strand</th>
<th>Chemistry</th>
</tr>
</thead>
<tbody>
<tr>
<td>s</td>
<td>Fwd</td>
<td>Single</td>
</tr>
<tr>
<td>r</td>
<td>Fwd</td>
<td>Double</td>
</tr>
<tr>
<td>z</td>
<td>Fwd</td>
<td>Single</td>
</tr>
<tr>
<td>x</td>
<td>Fwd</td>
<td>Double</td>
</tr>
<tr>
<td>y</td>
<td>Rev</td>
<td>Double</td>
</tr>
<tr>
<td>i</td>
<td>Fwd</td>
<td>Single</td>
</tr>
<tr>
<td>b</td>
<td>Fwd</td>
<td>Double</td>
</tr>
<tr>
<td>g</td>
<td>Rev</td>
<td>Double</td>
</tr>
</tbody>
</table>

And other special codes:

```
Feature                     17 cDNA
T3 cDNA
SP cDNA
Consensus pieces
Assembly pieces
```

Phrap command line options

The Phrap command line options are divided into two main groups: optional modifiers and flags. The optional modifiers are used to change default values in Phrap (require a value), and flags are used to enable or disable specific features. Some redundant modifiers are not shown here for a complete list, please read the Phrap documentation.

The scoring options change the way Phrap calculates the score for an alignment between sequences, referring to values used in the comparison matrix (matching residues are always rewarded by +1). They are:

- `--penalty`: penalty for base mismatch. Diminish to increase identity. (Default = -2)
- `--gap_init`: penalty to initiate a gap. Decrease to reduce number of potential gaps. (Default = penalty-2)
- `--gap_ext`: penalty to extend a gap. Decrease to reduce length of gaps. (Default = penalty-1)

And the flags are:

- `--raw`: don’t penalize low-complexity regions
- `--maxmatch`: maximum match length. Decrease under lack of memory at cost of speed. (Default = 30)
- `--minmatch`: minimum match length. Increase for speed at cost of stringency. (Default = 14)

The filtering options cause Phrap to remove selected matches from the assembly. They are:

- `--minscore`: minimum score for alignment. Increase for stringency. (Default = 80)
- `--vector_bound`: potential vector bases at beginning of each sequence. (Default = 80)

The input interpretation options change the way Phrap processes input data. They are:

- `--subclone delim`: delimitates subclone name. (Default = ‘>‘)
- `--group delim`: delimitates groupname(check preassemble flag below). (Default = ‘>‘)
- `--trim_start`: bases to remove at start. (Default = 0)

The assembly options change the way Phrap merges and assembles sequences. They are:

- `--forcelevel`: stringency relaxation (0 to 10). (Default = 0)
- `--maxgap`: maximum gap size for unmatched region when merging contigs. (Default = 30)

And the flags:

- `--revise greedy`: shatters and reattaches pre-contigs on low quality regions to increase overall score. Enable to correct some types of assemblies.
- `--shatter greedy`: same as above, without reattaching.
- `--preassemble`: preassembles reads within groups. The group names are defined by the first part of the sequence name prior to group delim.
- `--force_high`: ignores edited high-quality discrepancies.

The consensus construction options directly affect the algorithm that creates the weighted directed graph used to determine consensus. Each node in this graph is a subsequence segment. Higher values reduce memory usage but decrease the accuracy of the consensus sequence found. They are:

- `--node_seg`: minimum segment size to define a node. (Default = 8)
- `--node_space`: minimum space between segments in a single sequence to define a node. (Default = 4)

The output options change the generation of files after the execution of Phrap. They are:

- `--qual show`: minimum quality to show sequence in upper case. (Default = 20)
And the flags are:

- `--tags` tags selected lines in the Phrap output.
- `--screen` masks phrag-inferred vectors and chimeric segments in the .ace file.
- `--old_ace` creates an old style format .ace file.
- `--new_ace` creates a new style format .ace file.
- `--ace` same as `--new_ace`.
- `--view` creates view file for phrapview.
- `--print_extraneous_matches` shows non-local matches between contigs.

Other miscellaneous options are:

- `--max_subclone_size` maximum subclone size (to check forward-reverse pairs). (Default = 5000)
- `--confirm_length` minimum alignment size to confirm an alignment. (Default = 8)
- `--confirm_score` minimum alignment score to confirm an alignment. (Default = 30)

And the flags are:

- `--retain_duplicates` retain exact duplicate reads.

---

### Phrap output

There are three types of output for Phrap: **standard output**, **standard error** and **files**. The **standard output** and **standard error** are the messages shown on the console during the execution of Phrap. These are independent and can be independently redirected to a file (i.e. using `> phrap.out` to redirect standard output into a file named phrap.out and `2> phrap.err` to redirect standard error into a file named phrap.err). The standard output gives a summary of the final assembly, including data anomalies and possible sites of assembly error. The information it provides includes:

(i) Contigs and corresponding reads;
(ii) Read alignments and qualities against contigs;
(iii) Matching regions within and between contigs;
(iv) Suspect reads (probable deletions or chimeras);
(v) Possible assembly errors;
(vi) Forward/reverse consistency checks.

The standard error basically shows execution errors and warnings, and the point reached in the execution of Phred. The output files keep the name from the input file, but with suffixes added according to file type as follows:

- `.contigs` FASTA file containing contig sequences.
- `.contigs.qual` corresponding qualities file for `.contigs`.
- `.singlets` FASTA file containing singlets (unassembled reads).
- `.log` diagnostic only.

### Phrap vector screening

Before running Phrap, vector sequences should be masked or removed, since they may interfere with the assembly. The easiest way to do this is using the program `cross_match`: just create a FASTA file containing all the vector sequences you want to screen for and execute `cross_match` with both screen and sequence files with option `--screen`. E.g., for a FASTA sequence file named `seq_file` and a FASTA vector file named `vector`:

```
$ cross_match seq_file vector --screen
```

This causes `cross_match` to create a file named `seq_file` screen containing all sequences from `seq_file` where all bases matching sequences in `vector` are replaced by ‘X’, and ignored by Phrap during assembly.

### Phred/Phrap integration

The easiest way to execute both Phred and Phrap is using the `phredPhrap` script available in the Consed package. To proceed, create a directory to store your assembly (i.e., ‘assemble’), create your vector sequences file and name it `vector.seq` and put it inside this directory. Then, create three directories in it: ‘chromat_dir’, ‘edit_dir’ and ‘phd_dir’. Put all your trace files (or correspondent symbolic links) in ‘chromat_dir’ and just execute `phredPhrap` under ‘edit_dir’. After execution, all PHD files and assembly results will be available in ‘phd_dir’ and ‘edit_dir’, respectively.

This document was written and designed by Marcos Renato R. Araujo with the help of Eduardo F. Formighieri, Marcelo Falsarella Carazzolle and Gonçalo A. Guimarães Pereira from the Brazilian EMNet node, and distributed by the P&PR Publications Committee of EMNet.

EMNet - European Molecular Biology network - is a network of bioinformatics support centers situated primarily in Europe. Most countries have a national node which can provide training courses and other forms of help for users of bioinformatics software.

http://www.emnet.org/

A Quick Guide To Phrap
First edition © 2004