# Sequencing, data cleaning and assembling

Swiss Institute of Bioinformatics (SIB)

26-30 November 2001

# Some historical landmarks

| Organism | Genome size (bp) | Completed |
|---|---|---|
| Bacteriophage $\phi$X174 | $5386$ | 1982 |
| *Haemophilus influenzae* | $1,830,138$ | 1995 |
| *Saccharomyces cerevisiae* | $12 \times 10^6$ | 1996 |
| *Caenorhabditis elegans* | $95.5 \times 10^6$ | 1998 |
| *Arabidopsis thaliana* | $1.17 \times 10^8$ | 1999 |
| *Drosophila melanogaster* | $1.8 \times 10^8$ | 1999 |
| *Homo sapiens* | $3.3 \times 10^9$ | 2000 |

# How is possible to sequence full genomes?

Fred Sanger developed the first DNA sequencing method in 1977.

In the last 10 years a number of improvements have been made to the original Sanger method

- Enhancements in the biochemical components required for the sequencing reaction, such as thermostable polymerases:
- Capillary-based sequencing instruments (500-800 bp of high quality sequence per reaction)
- More robust fluorescent dye systems
- Advances in the laser-based instrumentation for fluorescent labeled DNA detection
- Robotic systems for automation of a number of steps (sub-cloning, clones storage, DNA purification, sequencing reactions, ...)

The effects of these improvements are:

- Better DNA sequence quality
- Higher throughput of DNA sequences
- Decrease in the costs
- Automation

# The sequencing process: I. Clone-by-clone shotgun sequencing

This method is also referred as hierarchical shotgun sequencing or map-based shotgun sequencing.

This strategy follows a 'map first, sequence second' strategy:

- Map construction
  - Pieces of genomic DNA are cloned in BACs (100-200 kb), and in some rare cases in PACs (100-200 kb), or YACs (up to 1 Mb).
  - Restriction enzyme digest-based fingerprints are derived for each BAC.
  - The fingerprints are used to infer clone overlaps and to assemble BAC contigs.
  - Supplementary mapping data are generated (for example STSs and genetic markers) that can be used for positioning of BAC contigs in the genome.
- Clone selection
  - Minimally overlapping clones are selected for shotgun sequencing.
- Subclone library construction
  - For each selected BAC, the cloned DNA is purified and fragmented.
  - Fragments are subcloned (plasmid- or M13-based vectors).

# The sequencing process: I. Clone-by-clone shotgun sequencing

- Random shotgun
  - Randomly selected subclones are selected and sequenced.
  - Sequence reads are computationally assembled into sequence contigs.
- Finishing
  - Highly accurate sequences are produced to solve problems as discontinuities between sequence contigs (gaps), regions of low sequence quality, ambiguous bases, and contig misassembly.
- Sequence authentication
  - Check for the presence and correct order of known sequence-based markers (STSs, genetic markers, genes, ...).
  - Check for concordance with clone restriction enzymes-based fingerprints.

For a detailed description of the method for the human genome see International Human Genome Sequencing Consortium (2001), Nature 409, 860-921.

# The sequencing process: II. Whole-genome shotgun sequencing

This method involves the assembly of sequence reads generated in a random, genome-wide fashion.

Requires higher redundant sequence coverage.

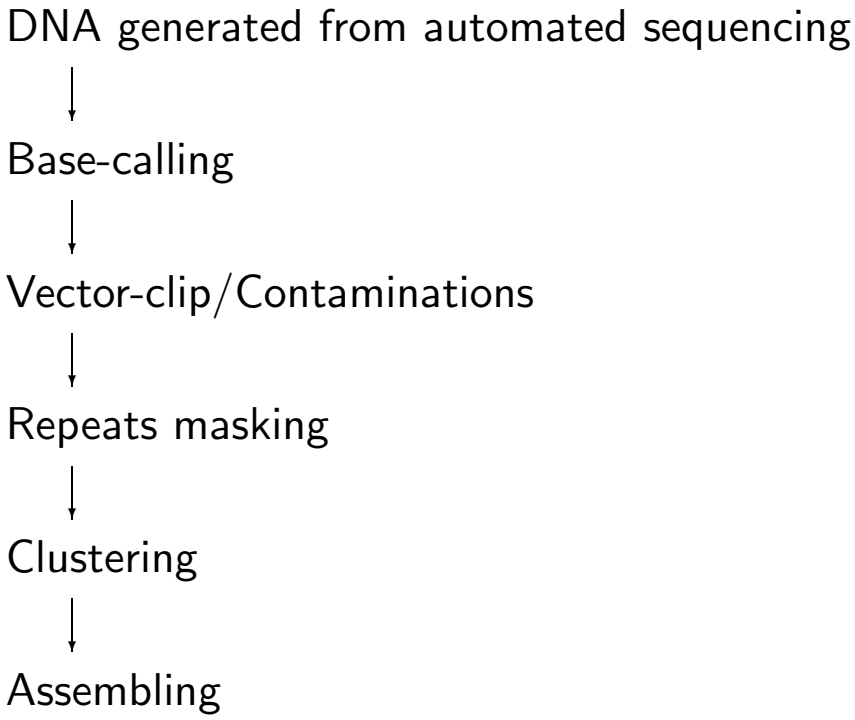
Bypass the need for a clone-based physical map (true?).

For a detailed description of the method for the human genome see Venter et al. (2001), Science 291, 1304-1351.

# The sequencing process

# Assembling pipeline

A typical analysis and assembling pipeline:

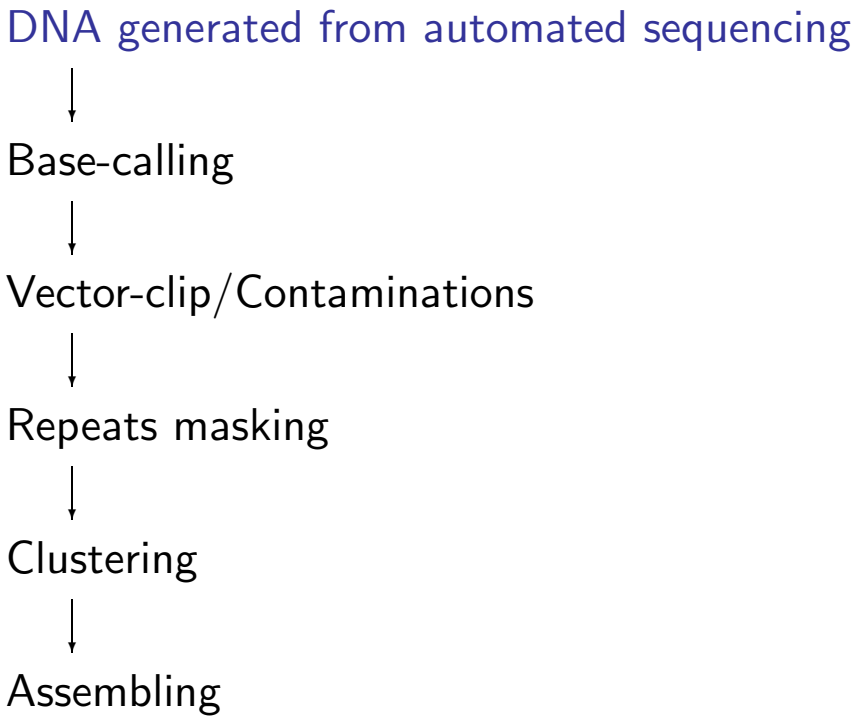



Specific software has been developed to process each step of the pipeline.

There are specific problems for each step of the pipeline.

# Assembling pipeline

The first step: Sequencing and read of the gel to produce a chromatogram.

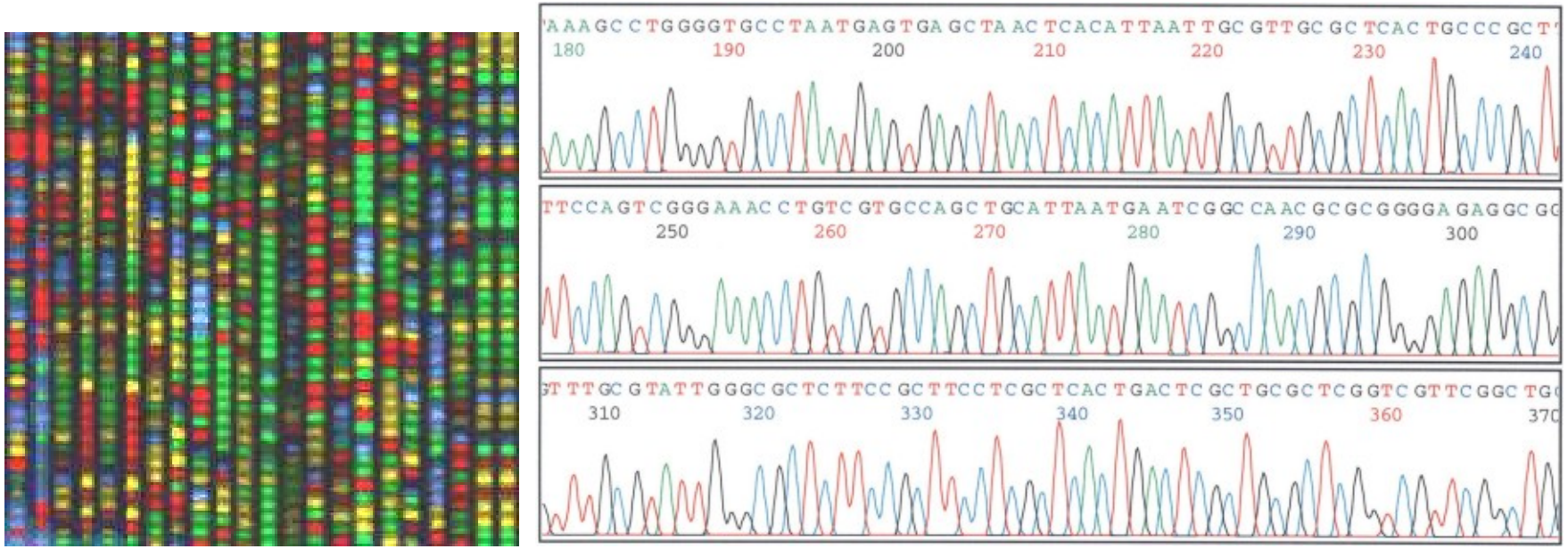


# Automated sequencing

The different steps in automated sequencing:

- DNA fragments are labeled with fluorescent dyes.
- Electrophoresis (slab or capillary gels).
- Laser detection.
- Laser measures are translated into a DNA sequence:
  - ▷ Lane tracking: gel line boundaries are identified (not necessary with capillary technology).
  - ▷ Lane profiling: each of the 4 signals is summed across the line width to create a profile (trace).
  - ▷ Trace processing: signal processing methods to deconvolve and smooth the signal and reduce the noise. This step produces the final chromatogram.
  - ▷ Base-calling: The chromatogram is translated into the nucleotides sequence.

# Automated sequencing



AGGGATGGACGTNGAGCTCCAAGAAAGGAAAAATGGGGTGNACACCATGGGATTGGATACGTGGGACCAG
CNCCATGAAGTGAAGGAGACTAATGAACAGAACTTCTCAAAATAGCCACTGAACTTTTACTTACAGAAAG
AGCTTATGTCAGCCGGCTCGACCTCCTAGATCAGGTATTTTATTGCAAACTATTAGAAGAAGCAAACCGA
GGCTCATTTCCTGCAGAGATGGTGAATAAAATCTTTTCTAACATTTCATCAATAAATGCCTTCCATAGTA
AATTCCTATTACCTGAGCTGGAGAAACGAATGCAAGAATGGGAAACTACACCCAGAATTGGAGATATCCT
GCAAAAGTTGGCGCCATTCCTTAAGATGTATGGAGAATACGTGAAGGGATTTGATAATGCAGTGGAACTG
GTTAAAACCATGACAGAGCGTGTTCCCCAGTTTAAATCAGTGACTGAAGAGATTCAGAAACAGAAGATCT
ATATCTACAGCAGCAAGCCATTCTAATAGTGC

# Chromatograms

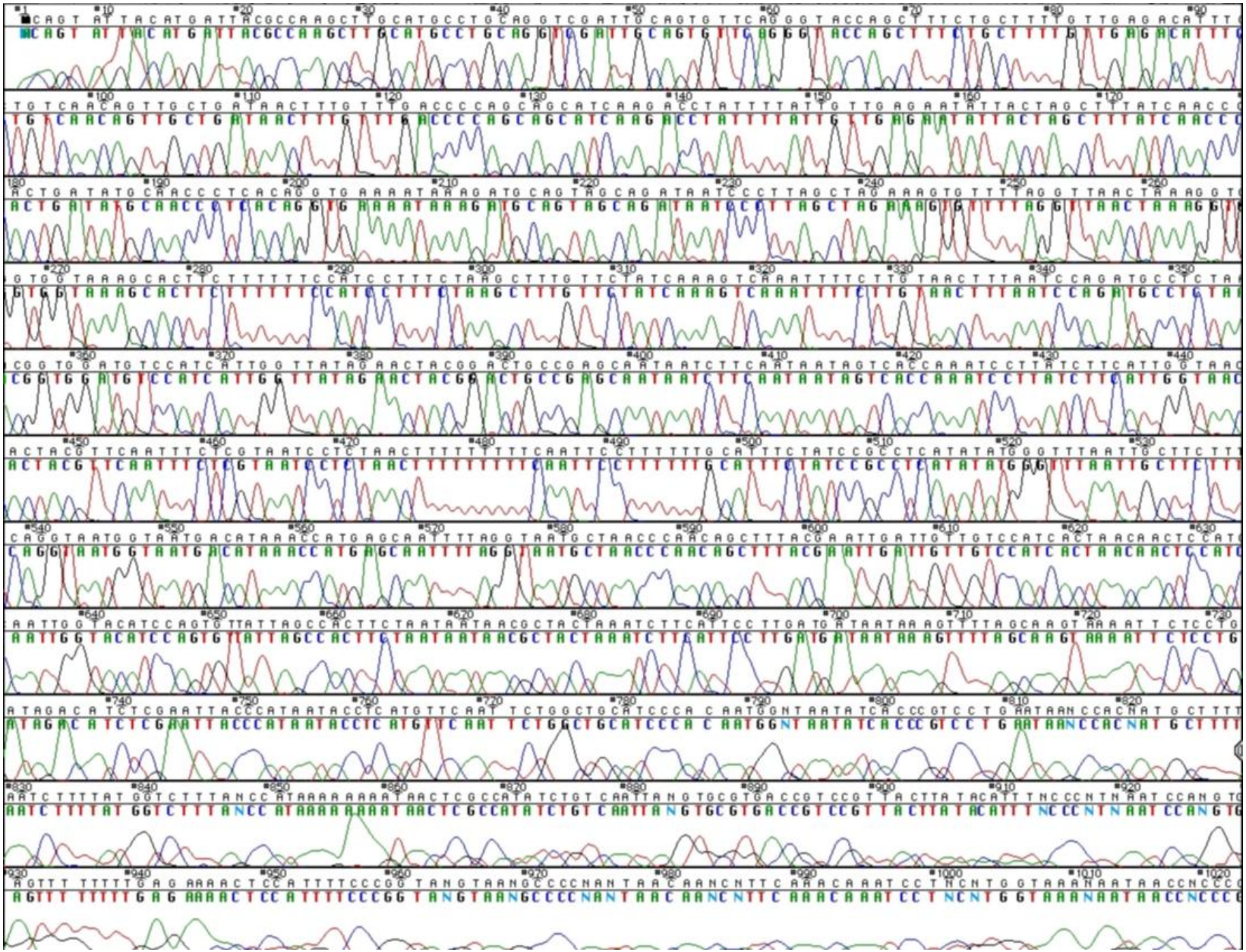
## Ideal trace:

- Non-overlapping peaks.
- Peaks are equally spaced.
- Good signal intensities for each nucleotide read.

## Real trace:

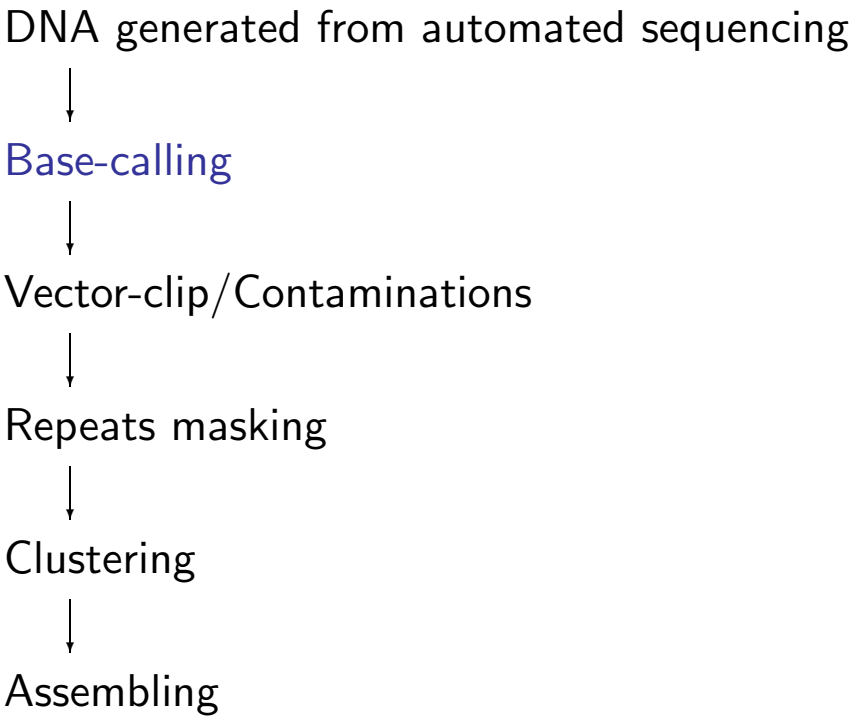
- Imperfections due to:
  - ▷ sequencing reactions;
  - ▷ gel electrophoresis;
  - ▷ trace processing.
- The first 50 peaks of a trace are noisy and unevenly spaced.
- Toward the end of the trace the peaks become progressively less evenly spaced (diffusion effects increase, relative mass difference between successive fragments decreases).
- Compressions result in unevenly spaced and overlapped peaks.
- Polymerase affinity problems lead to dramatic changes on the signal.

# Chromatograms



# Assembling pipeline

The second step: Read of the chromatogram to get a high quality sequence.



# Base-calling

The goal of base-calling is to produce a sequence as accurate as possible from a chromatogram.

A number of software attempting to produce the best quality sequence have been developed:

- Phred (*Ewig et al., 98*);
- ABI (*Connell et al., 87*);
- Sax (*Berno, 95*);
- A base-calling library (*Giddings et al., 93*);
- ...

Phred is one of the most used programs for base-calling in a number of projects.

# Phred algorithm

The algorithm to translate a chromatogram to a DNA sequence is based on 4 phases:

- Idealized peak location (peak prediction) attempts to find idealized locations of the base peaks, using simple signal processing methods (Fourier methods).
  - Estimate period from high quality regions;
  - Extrapolate to low quality regions;
  - Repeat until idealized trace is smooth.
- Locating observed peak
  - Decide what is a peak for the four trace arrays based on the area of the signal. Some peaks may be split in later steps.
- Matching observed and predicted peaks.
  - Assign observed peaks from the second step to the predicted peaks from the first step using dynamic programming algorithm. This involves shifting peaks around (spacing) as well splitting peaks.
  - Typically all predicted peaks have an observed peak assigned to them through this procedure.

# Phred algorithm

- Finding missed peaks. Due to to compressions, extensive noise, or lane processing aberrations, well-resolved observed peaks could not be attributed to predicted peaks.The missed peaks are added to the predicted peaks is the following conditions are verified:
  - the observed peak has the largest of the four signals
  - the observed peak meets a minimum size criterion
  - the observed peak is unsplit
  - the observed peak is flanked by resolved peaks
  - adding the observed peak improves peak spacing

# Phred algorithm

Phred assigns a quality value $q$ to each base-call:

$$q = -10 \times log_{10}(p)$$

where $p$ is the estimated error probability for that base-call, which is calculated using an empirically calibrated algorithm that considers 4 parameters:

- Peak spacing (7-peak window)
- Uncalled/called ratio (7-peak window)
- Uncalled/called ratio (3-peak window)
- Peak resolution

The empirically calibrated algorithm was trained for best predictions using these parameters.

Normally, bad quality regions at the beginning and the end of a sequence are deleted for the following steps of the pipeline.

# Phred algorithm

Phred can read chromatogram files from a number of systems:

- SCF
- ABI model 373 and 373
- MegaBACE ESD

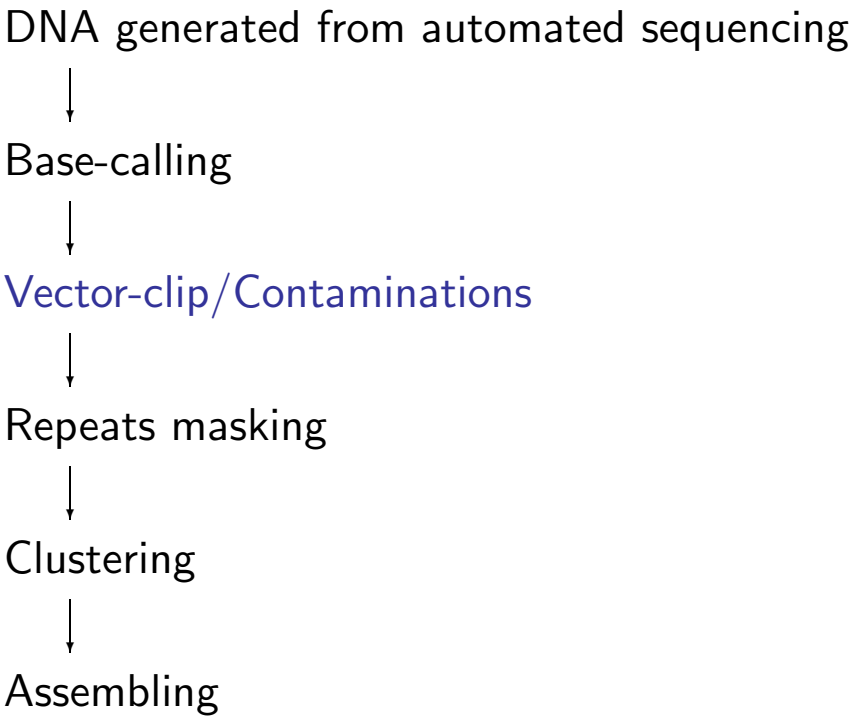
Phred runs on a number of UNIX platforms and its license is free for academic institutions.

Phred is easy to use and produces easily parsable files both for sequences and quality.

The Phred home page: http://www.phrap.org/.

# Assembling pipeline

The third step: Clean sequences from possible contaminations.



# Vector-clipping and contaminations

## Vector-clipping

- Delete 5' and 3' regions corresponding to the vector or adapters used for cloning.

## Contaminations

- Find and delete:
  - Bacterial contaminations;
  - Yeast contaminations;
  - Mitochondrial DNA;
  - ...

Standard pairwise alignment programs are used for the detection of vector and other contaminants (for example cross-match).

The pairwise alignment program is used to scan a sequence against a database of interest (for example: plasmid DNA sequences, bacterial sequences, ...).

# Dynamic programming

Dynamic programming requires substitution matrices.

By definition dynamic programming algorithms always find the best alignment between two sequences given a substitution matrix.

Substitution matrices (Scoring matrices) take into account a number of parameters to construct biological significant alignments:

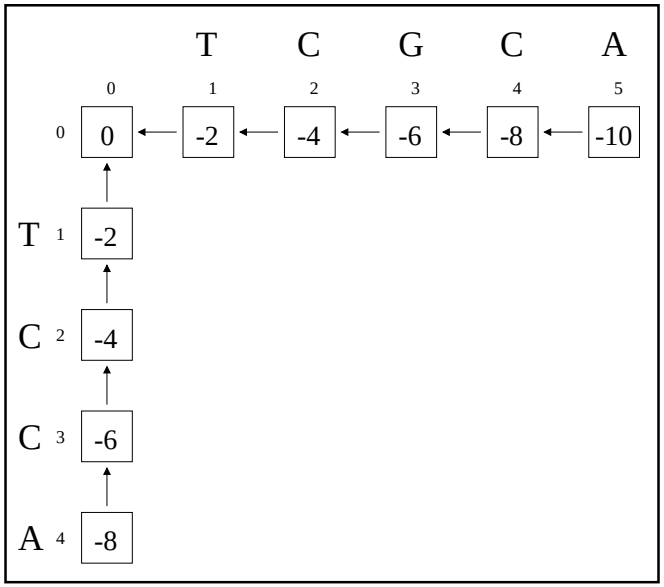
- Chemical properties (ex. changing a Leucine with an Isoleucine is a conservative substitution)
- Genetic code degeneration
- Evolutionary distances

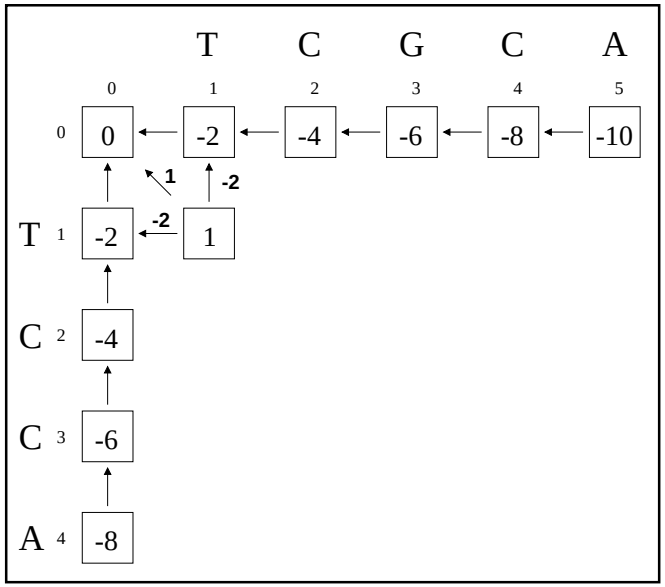
An example: the Needleman-Wunsch algorithm. Given 2 sequences $S_1$ and $S_2$ of lengths $L_1$ and $L_2$ respectively:

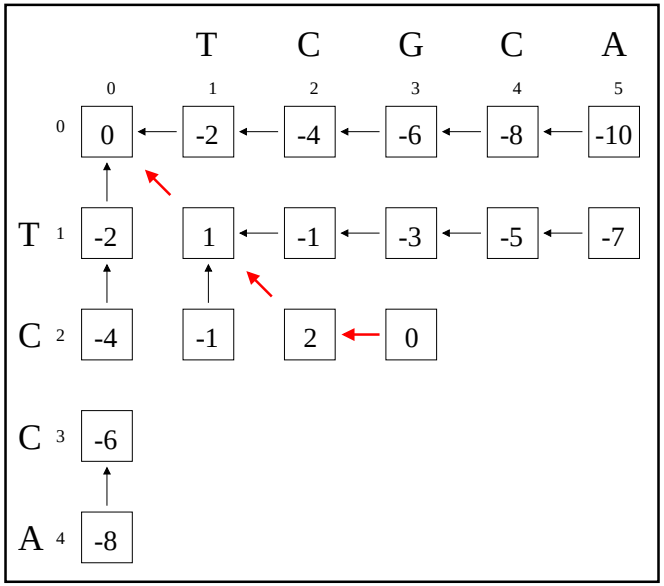
$$s_{i,j} = max \begin{cases} s_{i,j-1} - g \\ s_{i-1,j-1} + p_{i,j} \\ s_{i-1,j} - g \end{cases}$$

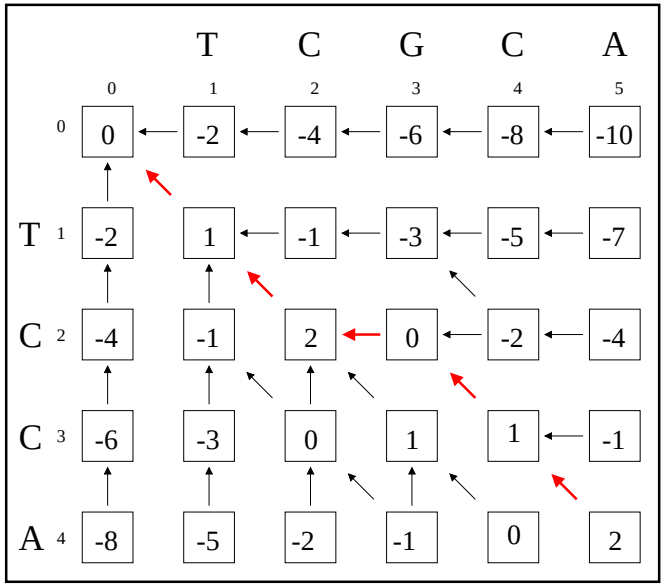
where $0 \leq i \leq L_1$, $0 \leq j \leq L_2$, $g$ is the gap penalty, $p_{i,j}$ is the value read from the substitution matrix for residues $i$ from $S_1$ and $j$ from $S_2$, and $s_{0,0} = 0$.

# Dynamic programming



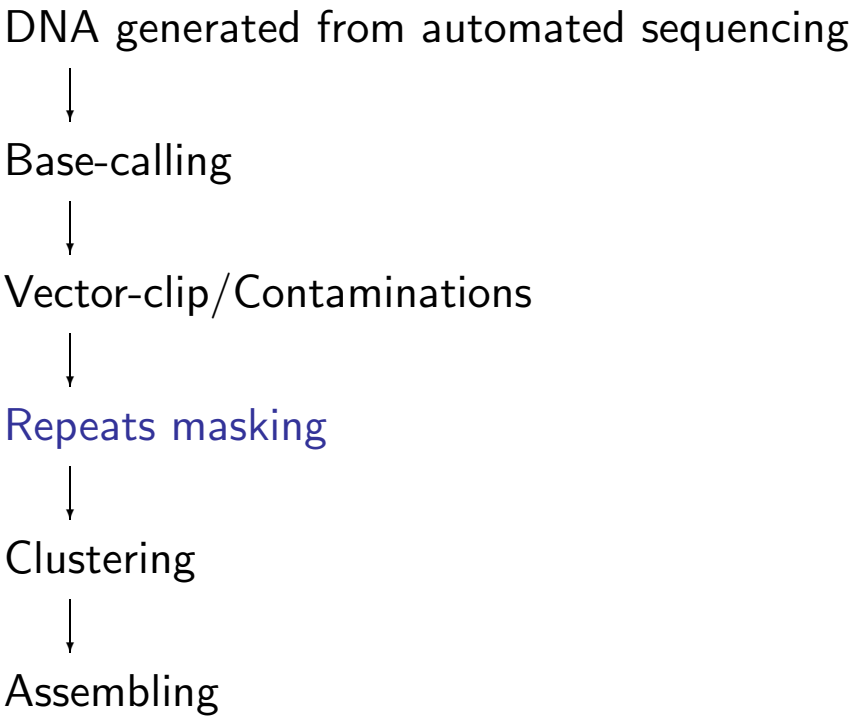



# Assembling pipeline

The fourth step: Repeats must be masked to avoid wrong sequence assembly.



# Repeats masking

Some repetitive elements found in the human genome:

| | Length | Copy number | Fraction of the genome |
|---|---|---|---|
| LINEs (long interspersed elements) | 6-8 kb | 850,000 | 21% |
| SINEs (short interspersed elements) | 100-300 bp | 1,500,000 | 13% |
| LTR (autonomous) | 6-11 kb | 450,000 | 8% |
| LTR (non-autonomous) | 1.5-3 kb | | |
| DNA transposons (autonomous) | 2-3 kb | 300,000 | 3% |
| DNA transposons (non-autonomous) | 80-3000 bp | | |
| SSRs (simple sequence repeats or microsatellite and minisatellites) | | | 3% |

# Repeats masking

Repeated elements:

- They represent a big part of the mammalian genome.
- They are found in a number of genomes (plants, ...)
- They induce false gene clustering and assembling.
- They should be masked, not deleted, to avoid false sequence assembling.
- ... but also interesting elements for evolutionary studies.
- SSRs important for mapping of diseases.

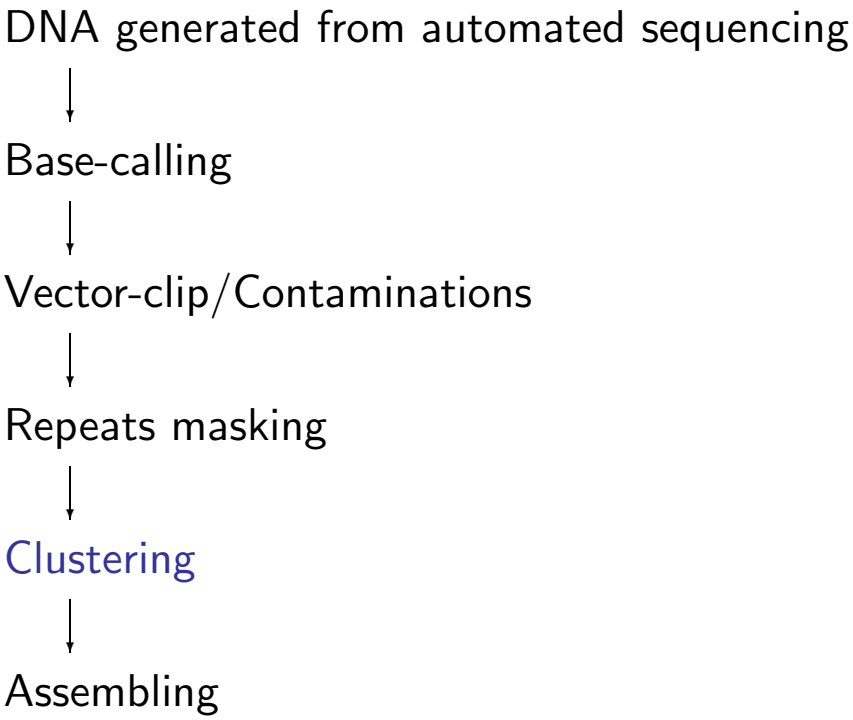
Ribosomal DNA can also be masked.

Tools to find repeats:

- RepeatMasker has been developed to find repetitive elements and low-complexity sequences. RepeatMasker uses the cross-match program for the pairwise alignments (http://repeatmasker.genome.washington.edu/cgi-bin/RepeatMasker).
- MaskerAid improves the performances of RepeatMasker by $\sim 30$ folds using WU-BLAST instead of cross-match (http://sapiens.wustl.edu/maskeraid)
- Repbase is a database of prototypic sequences representing repetitive DNA from different eukaryotic species.: http://www.girinst.org/Repbase_Update.html.

# Assembling pipeline

The fifth step: Clustering similar sequences together.



# Clustering

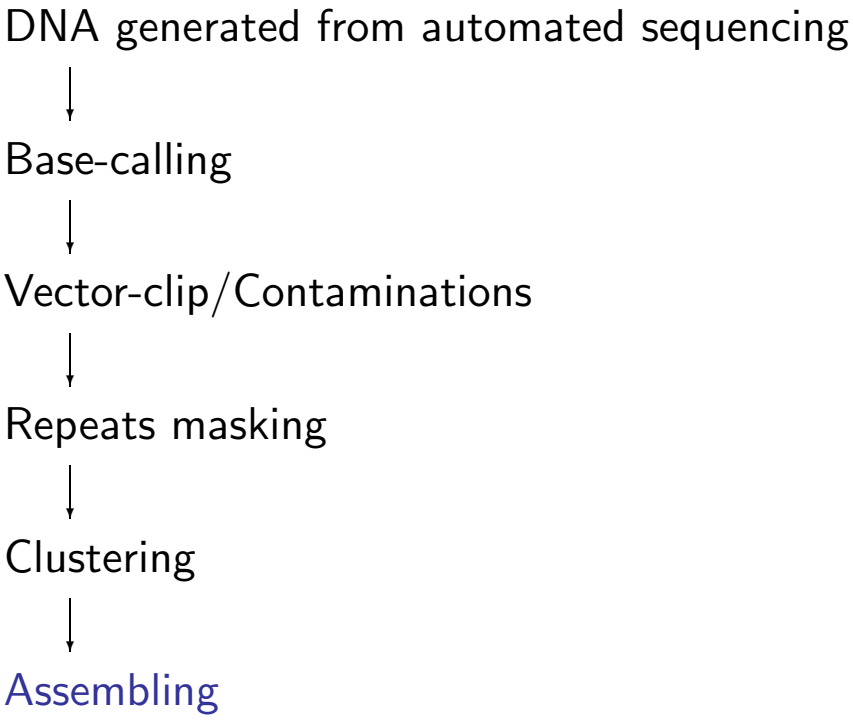
Clustering of the sequences:

- essential to decrease the complexity of the problem for the assembling step;
- uses standard pairwise algorithms:
  - cross-match;
  - BLAST;
  - LASSAP (LArge Scale Sequence compArison Package);
- strict parameters are selected for pairwise comparison;

The clustering step is not required for the clone-by-clone shotgun method, because the origin of the clones is known.

# Assembling pipeline

The final step: Assemble together sequences to build contigs and build the final genomic sequence.



# Assembling: Building contigs

Reconstruct the sequence of a sequence (BAC, ...) from the sequences of the subclone fragments.

Phrap is wide used for assembly of genomic sequences.

Phrap features:

- allows use of entire reads (not only high quality parts);
- uses quality values from phred;
- returns quality values for contig sequences;
- handle large datasets.

Other programs are available:

- CAP3;
- TIGR assembler;
- ...

# Assembling: Building contigs

Phrap algorithm (for details see: http://bozeman.mbt.washington.edu/):

- Find pairs with matching words (swat). Eliminate exact duplicates.
- Find probable vector matches and mark them to avoid their utilization for assembling.
- Find near duplicates reads and self-matches.
- Find matching pairs without a "solid" matching segment.
- Good pairwise matches used to compute revised quality values.
- Compute scores for the matches.
- Find best alignment for each matching pair.
- Identify possible chimeric sequences.
- Construct contig layouts, using consistent pairwise matches in decreasing score order.
- Construct contig sequences and align reads to contigs.

# Assembling: Assemble contigs

Order and catenate contigs produced by the previous steps.

GigAssembler has been developed to perform this last step using a number of different information.

GigAssembler procedure:

- Align mRNAs, ESTs, and BAC ends with the produced contigs.
- Use clones fingerprints to align contigs.
- Merge, order, and orient contigs using data mRNAs, ESTs, ... (GigAssembler).
- Combine contig assemblies into full chromosome assemblies.

For details see Kent and Haussler (2001) Genome research, 11, 1541-1548.