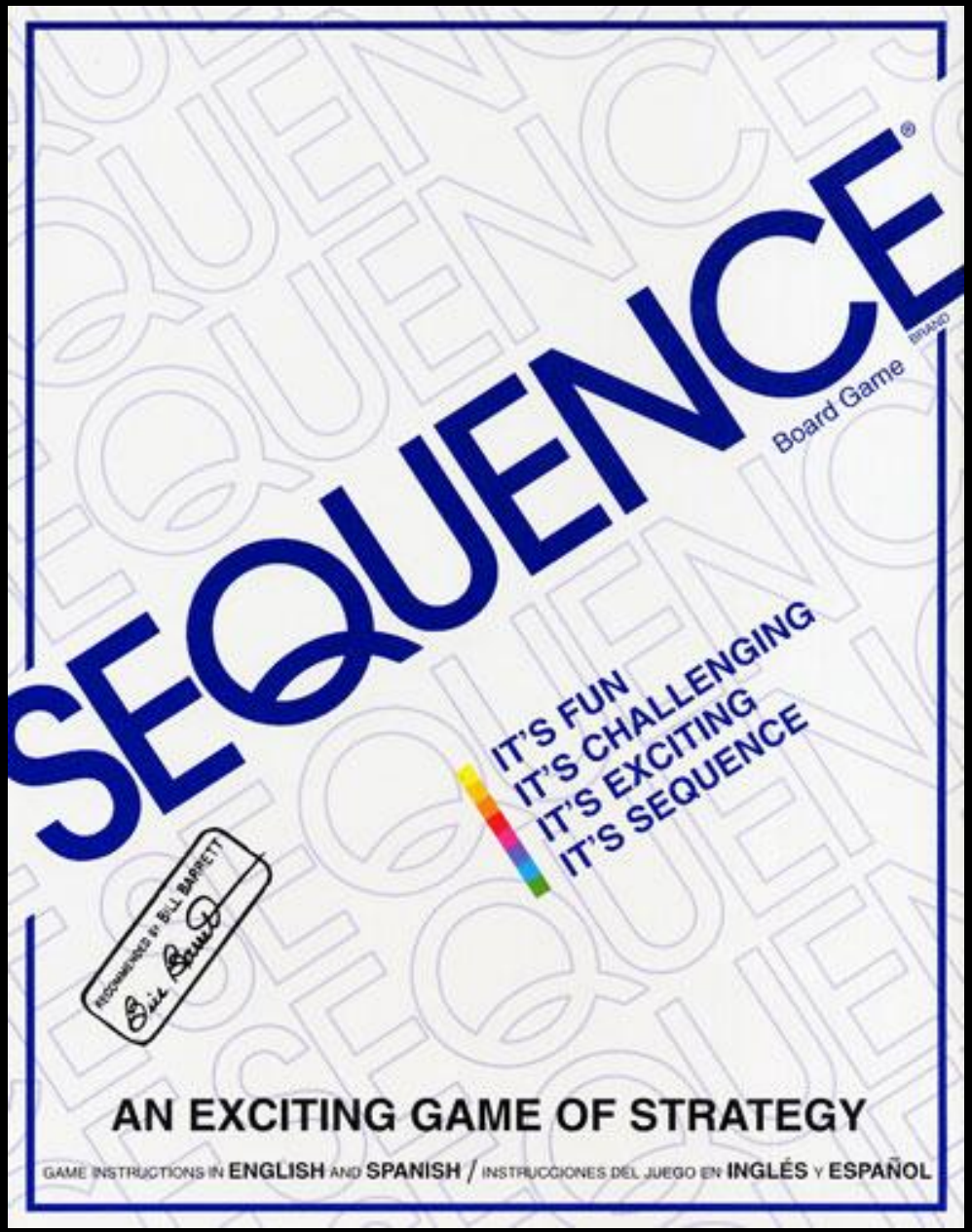
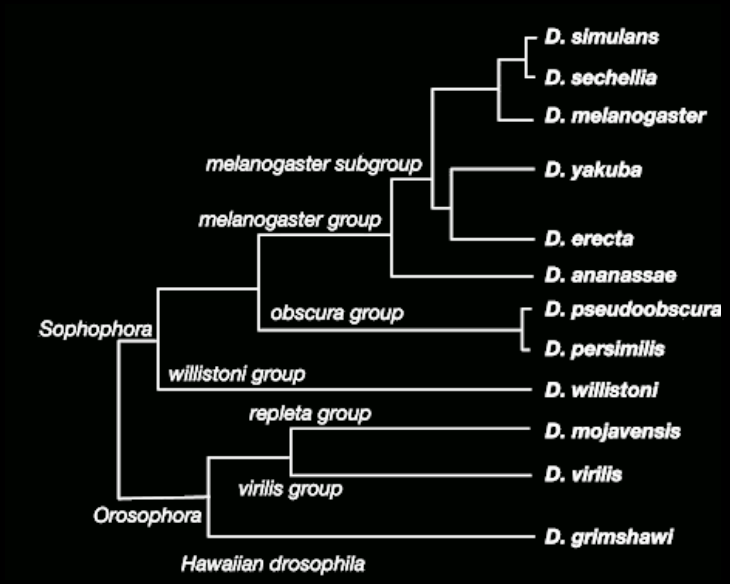


Reading genomes bit by bit

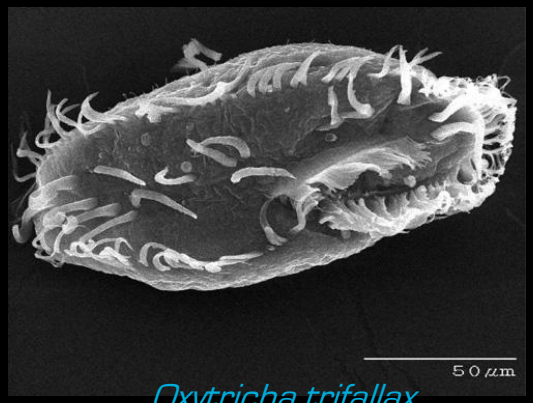
Sean Eddy
HHMI Janelia Farm
Ashburn, Virginia

Why did we sequence so many different flies?



the power of
comparative genome
sequence analysis

Why did we sequence a single-celled pond protozooan?



Oxytricha trifallax

exploiting unusual adaptations
and unusual genomes

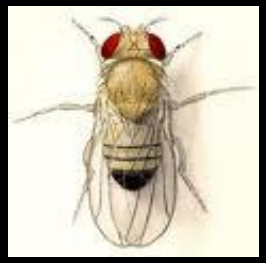
Symbolic texts can be cracked by statistical analysis

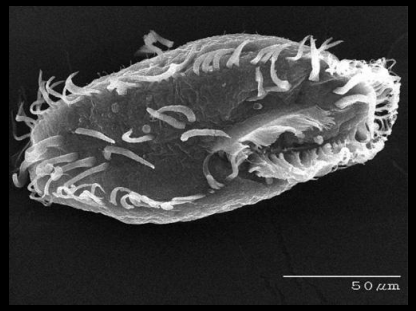
Linear B, from Mycenae
ca. 1500-1200 BC
deciphered by Michael Ventris and John Chadwick, 1953

"Cryptography has contributed a new weapon to the student of unknown scripts.... the basic principle is the analysis and indexing of coded texts, so that underlying patterns and regularities can be discovered. *If a number of instances can be collected*, it may appear that a certain group of signs in the coded text has a particular function...."

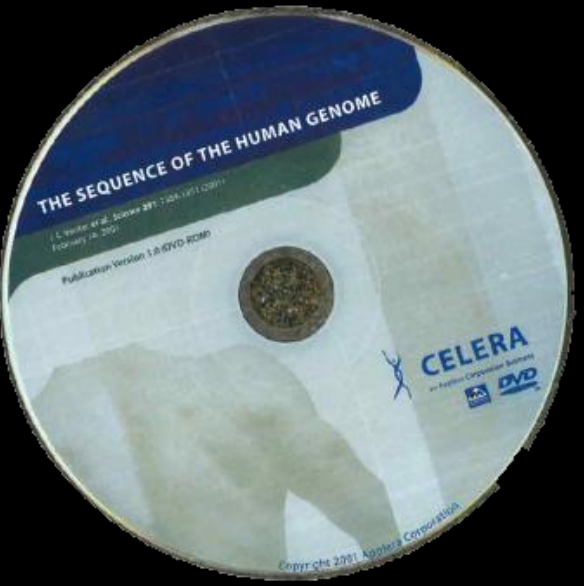
John Chadwick,
The Decipherment of Linear B
Cambridge University Press, 1958

```
[eddys@eddys-wm1 genomes]$ ls -l
total 66
-rwxrwxr-x  1 jonest data0  4227 Mar 31  2010 00README*
drwxrwsr-x  3 jonest data0   146 Apr 12  2010 Algae/
drwxrwsr-x  4 jonest data0    70 Jun 12  2009 Amoebozoa/
drwxr-sr-x  3 jonest data0    36 Mar 29  2009 Amphibians/
drwxrwsr-x 15 jonest data0   603 Aug 30  2007 Archaea/
drwxrwsr-x 32 jonest data0  1297 Aug 30  2007 Bacteria/
drwxr-sr-x  5 jonest data0    88 Dec 11  2009 Basal_animals/
drwxrwsr-x  3 jonest data0    25 Apr  2  2010 Chordates_lower/
drwxrwsr-x  5 jonest data0   155 Apr 28  2010 Chromalveolates/
drwxrwsr-x  5 jonest data0   120 Jun 10  2009 Ciliates/
drwxr-sr-x  5 jonest data0    98 Jun  2  2009 Deuterostomes-lower/
drwxrwsr-x  5 jonest data0   154 Apr 28  2010 Excavates/
drwxrwsr-x 16 jonest data0   570 May 13  2008 Fungus/
drwxrwsr-x 11 jonest data0   265 Jun  4  2009 Insects/
drwxrwsr-x 13 jonest data0   397 Feb  8 16:43 Nematodes/
drwxr-sr-x  7 jonest data0   144 Mar 29  2010 Plants/
drwxr-sr-x  3 jonest data0    85 Sep 30  2009 Platyhelminthes/
drwxrwsr-x  5 jonest data0   116 Aug 26 11:58 Prokaryotes/
drwxr-sr-x  3 jonest data0    37 Feb 13  2009 Reptiles/
drwxrwsr-x  5 jonest data0    98 Apr 20  2010 Vertebrates/
drwxrwsr-x  7 jonest data0   132 Mar 26  2010 Virus/
drwxrwsr-x  6 jonest data0   180 Aug 30  2007 a.thaliana/
drwxrwsr-x  2 eddys  data0   993 Nov 24 16:55 amphimedon_queenslandica/
lrwxrwxrwx  1 jonest data0    26 Jun  3  2009 anopheles.gambiae -> Insects/anopheles.gambiae//
drwxrwsr-x  3 jonest data0    92 Aug 30  2007 brassica.oleracea/
drwxr-sr-x  3 jonest data0    45 Sep 30  2009 bup.Platyhelminthes/
drwxrwsr-x  6 eddys  data0   110 Feb  8 16:41 caenorhabditis/
drwxrwsr-x  3 jonest data0    54 Oct 18  2007 capitella_sp/
drwxrwsr-x  7 jonest data0   220 Mar 23  2010 chicken/
drwxrwsr-x  3 jonest data0    54 Aug 30  2007 chimp/
drwxrwsr-x  4 jonest data0    82 Aug 30  2007 ciona.intestinalis/
drwxrwsr-x  4 jonest data0    84 Aug 30  2007 ciona.savignyi/
lrwxrwxrwx  1 jonest data0    24 Jun 12  2009 dictyostelium -> Amoebozoa/dictyostelium//
drwxrwsr-x  3 jonest data0    54 Aug 30  2007 dog/
lrwxrwxrwx  1 jonest data0    20 Jun  4  2009 drosophilid -> Insects/drosophilid//
-rwxrwxr-x  1 jonest data0 24724 Jan 17  2007 facheck*
drwxrwsr-x  3 jonest data0    51 Aug 30  2007 hedgehog/
drwxrwsr-x  3 jonest data0    54 Oct 18  2007 helobdella_robusta/
lrwxrwxrwx  1 jonest data0    17 Jun  3  2009 honeybee -> Insects/honeybee//
drwxrwsr-x 22 jonest data0   675 Sep  3 11:06 human/
drwxrwsr-x  3 jonest data0    54 Oct 19  2007 lottia_gigantea/
drwxrwsr-x 10 jonest data0   324 Aug 30  2007 mouse/
-rw-r--r--  1 jonest data0  5086 Jan 18  2010 notes
drwxr-sr-x  3 jonest data0    51 Oct 27  2008 oikopleura.dioica/
drwxrwsr-x 15 jonest data0   510 Oct 29 13:02 oxytricha_trifallax/
drwxrwsr-x  3 jonest data0    53 Aug 30  2007 platypus/
drwxrwsr-x  4 jonest data0    84 Sep 16  2008 pristionchus_pacificus/
drwxrwsr-x  4 jonest data0    83 Aug 30  2007 rat/
drwxrwsr-x  3 jonest data0   929 Apr 26  2010 s.lemnae/
drwxrwsr-x  3 jonest data0   684 Aug 30  2007 sargasso_sea/
drwxrwsr-x  4 jonest data0    84 Oct 17  2007 schistosoma.mansoni/
drwxrwsr-x  3 jonest data0    54 Aug 30  2007 schmidtea_mediterranea/
drwxrwsr-x  5 jonest data0   136 Aug 30  2007 takifugu.rubripes/
drwxrwsr-x  7 jonest data0   202 Mar  1  2010 tetrahymena.thermophila/
drwxrwsr-x  3 jonest data0    51 Aug 30  2007 tetraodon/
lrwxrwxrwx  1 jonest data0    28 Jun  3  2009 tribolium.castaneum -> Insects/tribolium.castaneum//
drwxrwsr-x  3 jonest data0    54 Aug 30  2007 trichinella_spiralis/
drwxrwsr-x  3 jonest data0    54 Oct 22  2007 trichoplax_adhaerens/
drwxrwsr-x  6 jonest data0   166 Mar 30  2010 zebrafish/
```





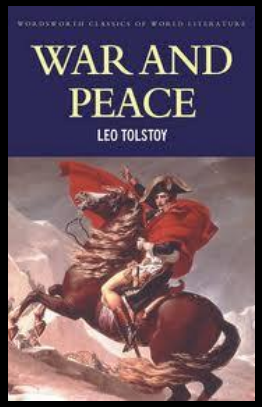
How much data are we talking about, really?



my coffee coaster
3 GB

		STORAGE COST/YEAR	TIME TO DOWNLOAD
raw images	30 TB	$36,000	20 days
unassembled reads	100 GB	$120	1 hr
mapped reads	100 GB	$120	1 hr
assembled genome	6 GB	$7	5 min
differences	4 MB	$0.005	0.2 sec

selab:/misc/data0/genomes
450 GB



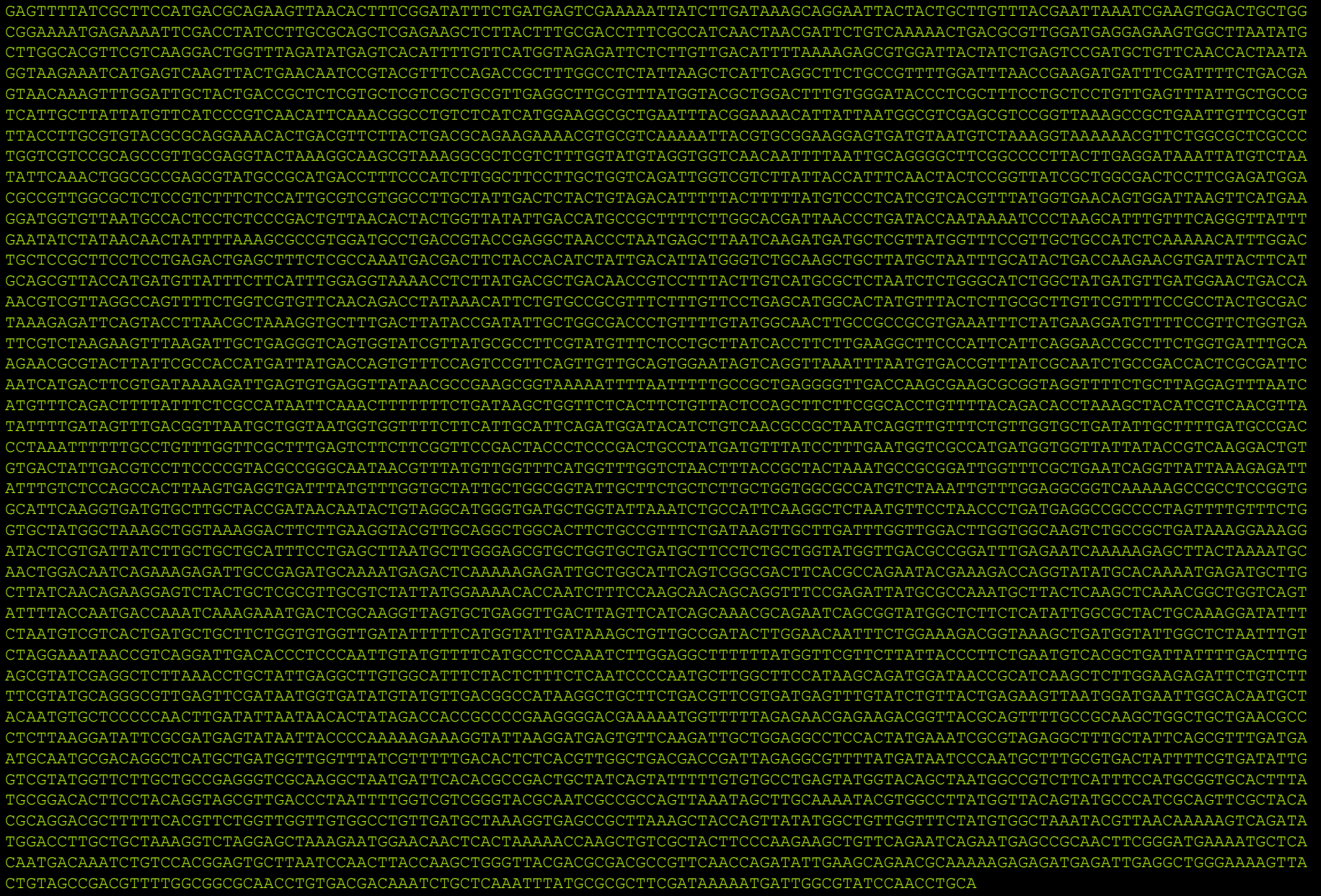
3 MB

selab:~eddys/Music/iTunes
128 albums
15 GB

JFRC computing, available disk
~ 1 petabyte (1000 TB)

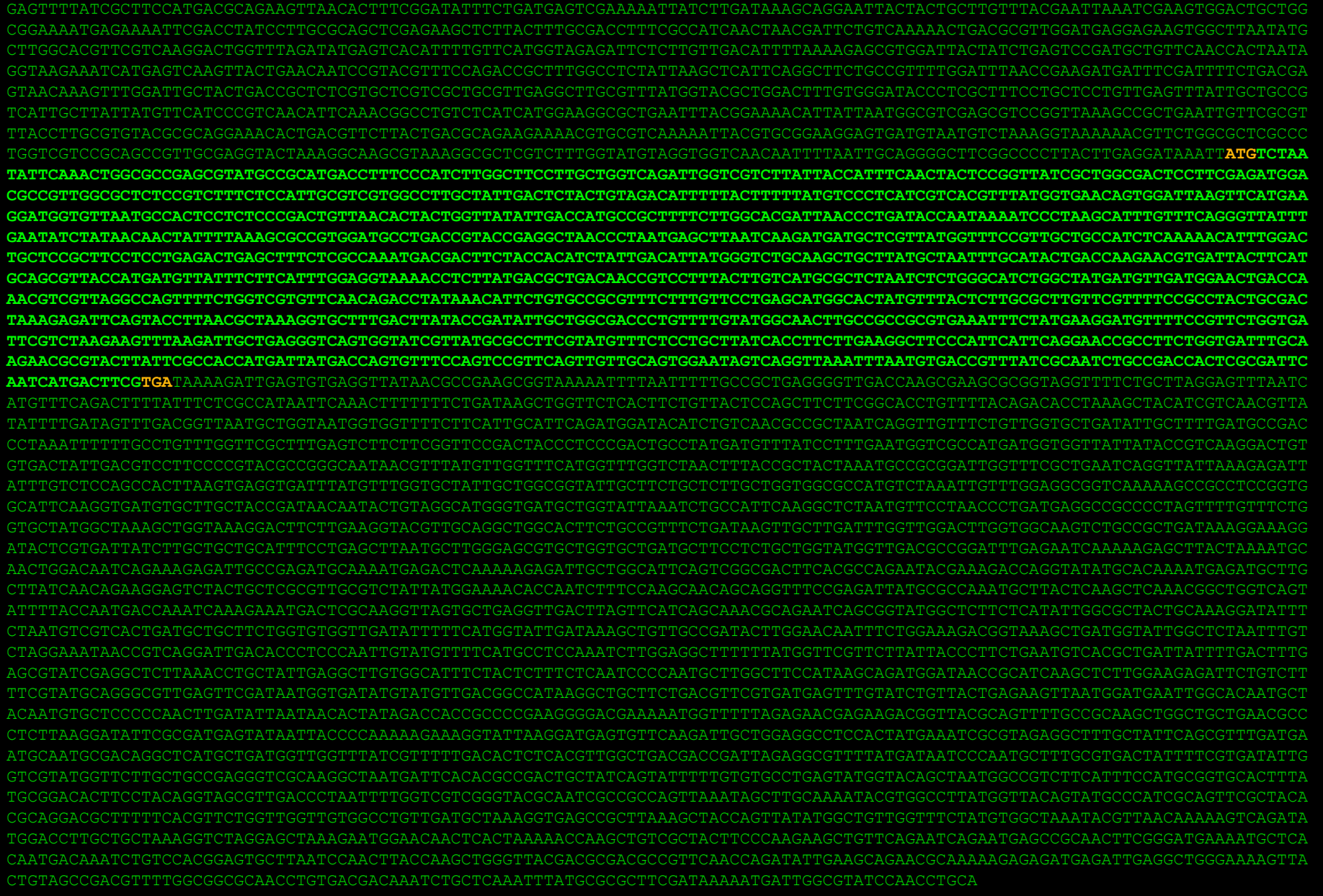
1000 Genomes Project pilot
5 TB (30 GB/genome)

NCBI Short Read Archive
200 TB + 10-20 TB/mo



bacterial virus phiX174

Fred Sanger, Alan Coulson, et al. (1978)



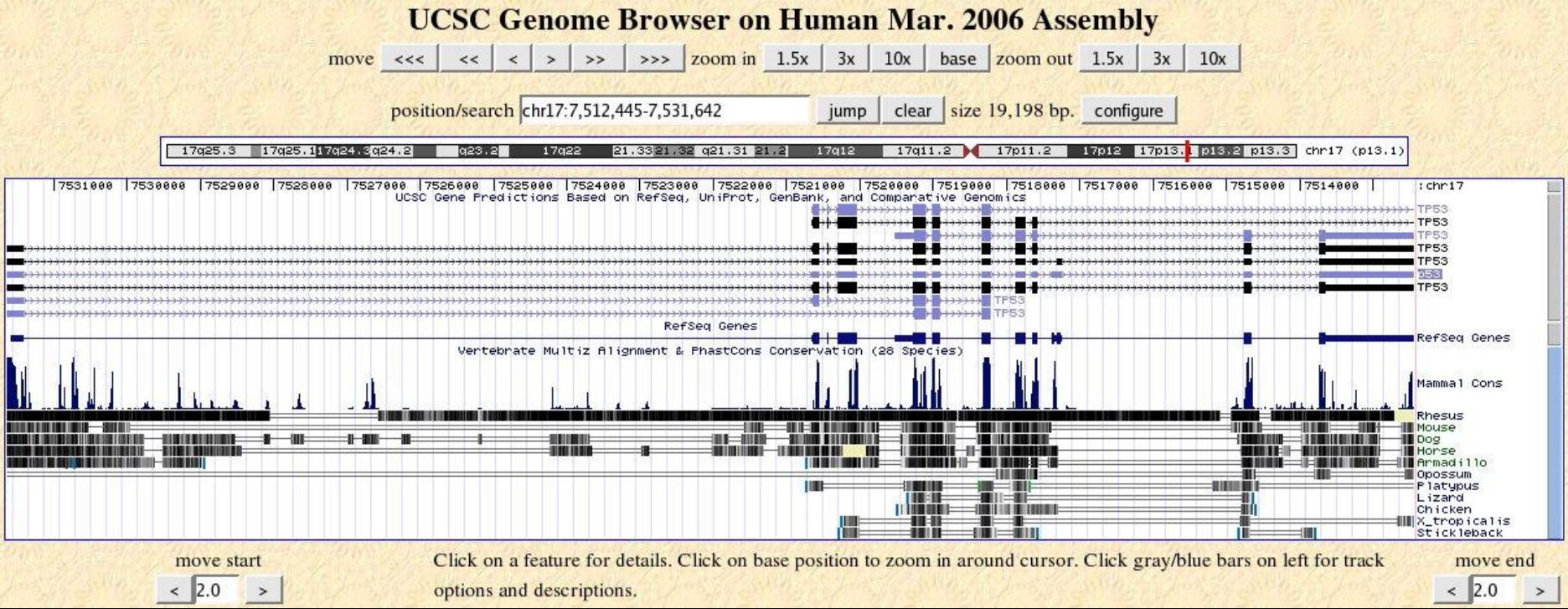
bacterial virus phiX174

Fred Sanger, Alan Coulson, et al. (1978)

bacterial virus phiX174

Fred Sanger, Alan Coulson, et al. (1978)

Sequence conservation is an important signal



a view of 20 kb around the human P53 gene

UCSC Browser: Jim Kent and David Haussler
Ensembl Browser: Ewan Birney
MULTIZ: Webb Miller
PhastCons: Adam Siepel

More genomes = more resolution

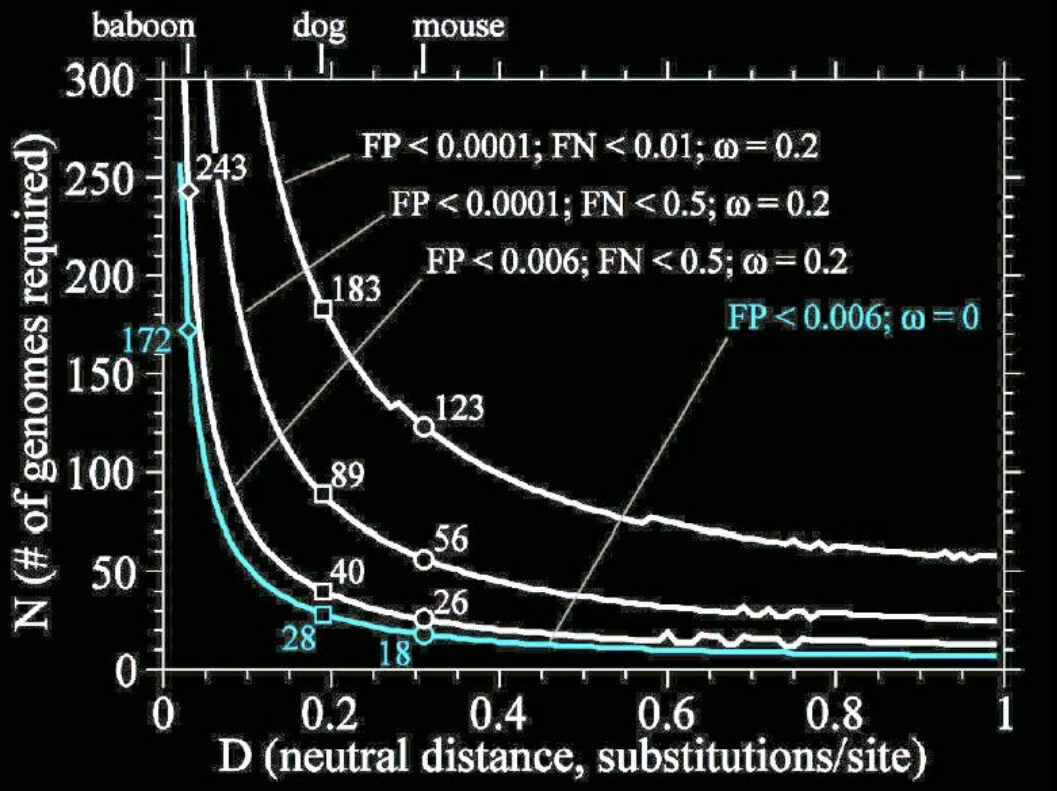


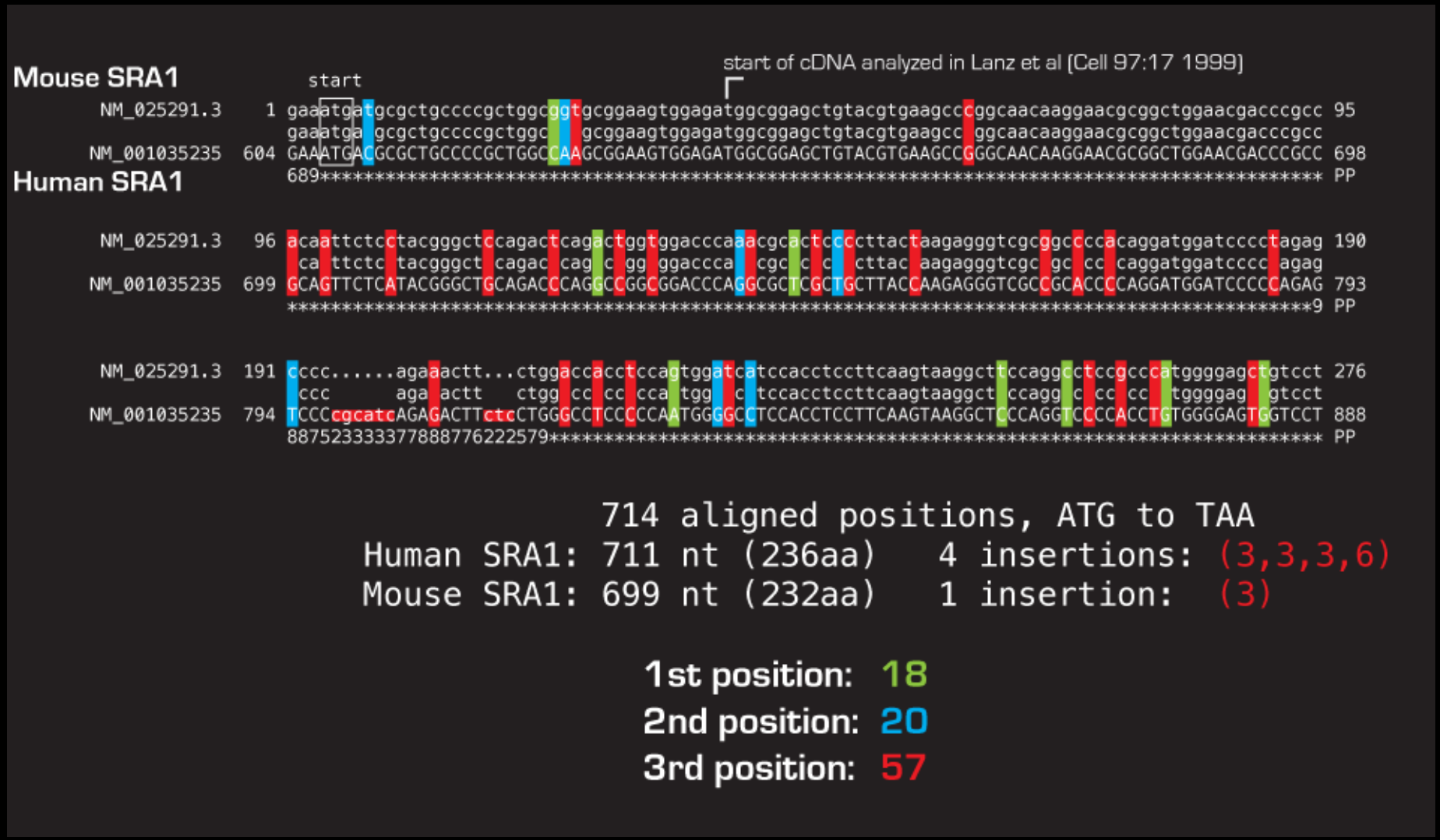
Figure 1. Number of Genomes Required for Single Nucleotide Resolution

It's not that we're so interested in genomes of 100's of species;
We're interested in comparing 100's of genomes to our main model systems.
This is the driver for comparative genome sequencing.

SR Eddy, PLoS Biology 3:e10 (2005)

Greg Cooper, Arend Sidow, others

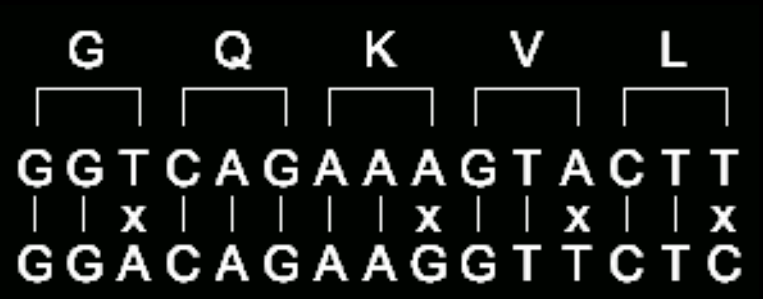
The *pattern* of conservation is also informative



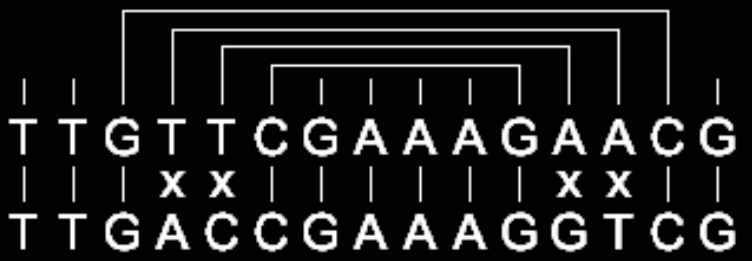
pioneered by Jonathan Badger and Gary Olsen (CRITICA)
many comparative coding region and genefinders, including work from Michael Brent, Irmi Meyer, Manolis Kellis, others

Different sequences evolve under different constraints

OTHER model:
terms P(a,b | OTH)
sequence divergence only



CODING model:
terms P(aaa,bbb | COD)
codon constraints

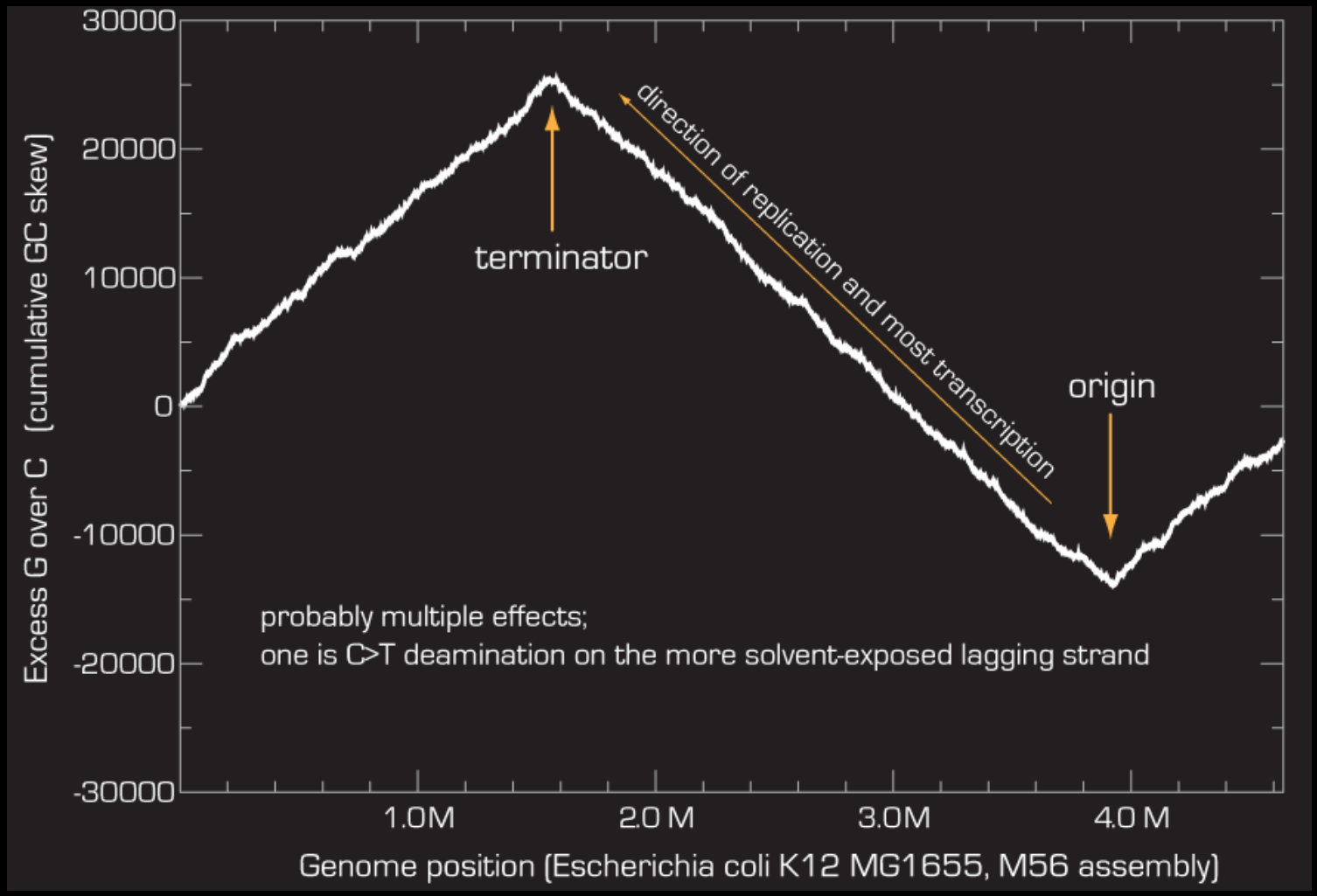


RNA model:
terms P(a-a', b-b' | RNA)
secondary structure constraints

E Rivas and SR Eddy, BMC Bioinf 2:8 (2001)

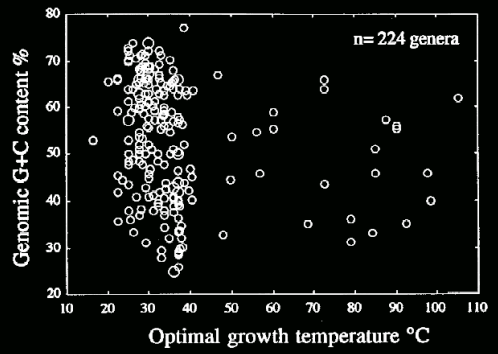
introduced by Elena Rivas (QRNA)
now also Ivo Hofacker et al (RNAz), Jakob Pedersen (EvoFold)

Even some subtle biophysical effects show up in sequence

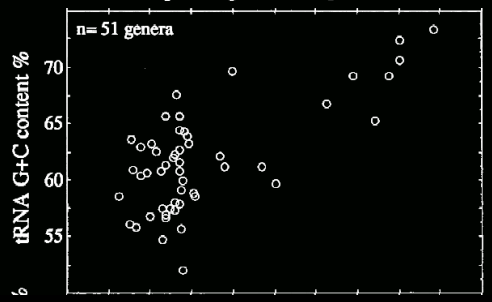


pioneered by Jean Lobry
now also Alexander Grigoriev, others
RNA genefinding applications by Phil Green, Arian Smit

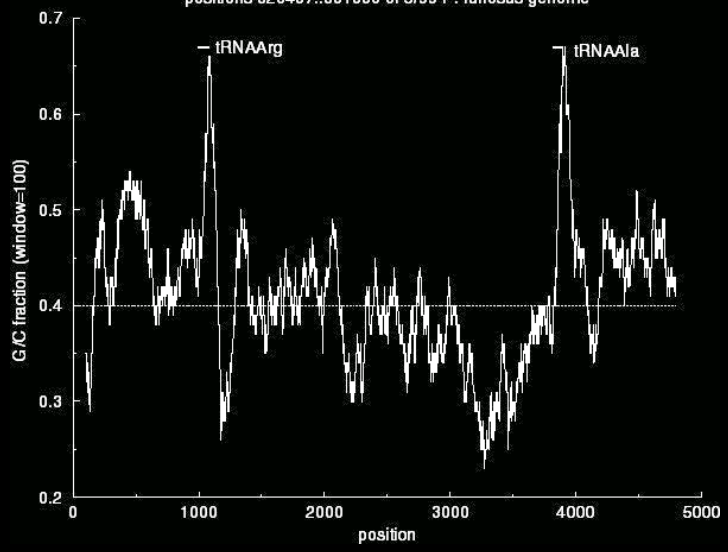
Fetch me an AT-rich hyperthermophile



Genome GC content uncorrelated w/ growth temp



GC content of structural RNAs highly correlated



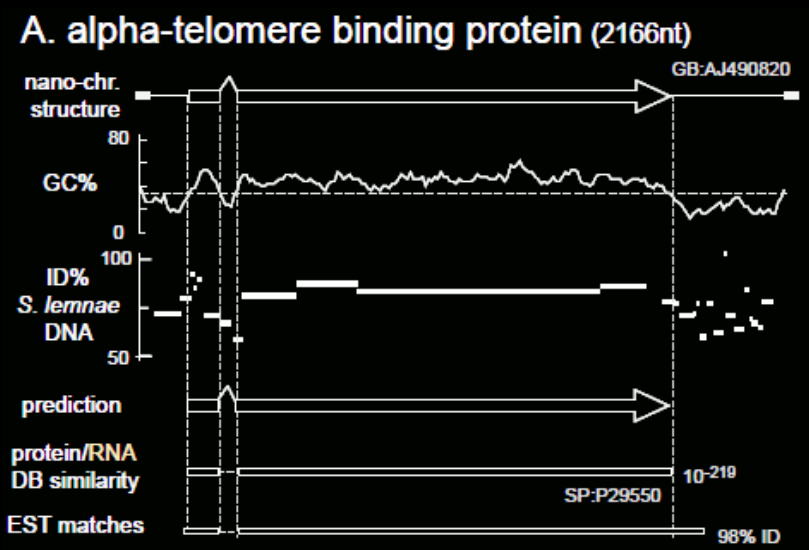
So: in the most extreme AT-rich hyperthermophiles, structural RNA genefinding becomes trivial

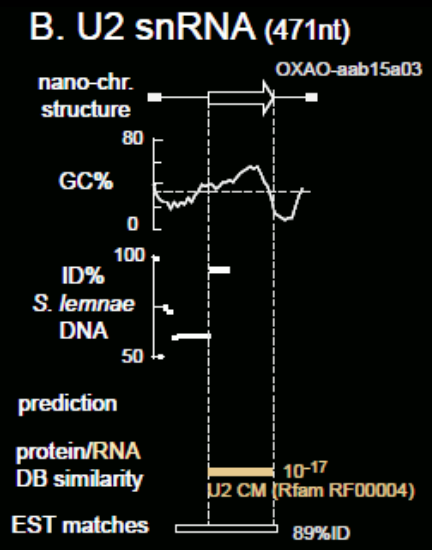
Pyrococcus furiosus – Vulcano Island, Italy. 98C. 60%AT.
P. abyssi, P. horikoshii
Methanococcus jannaschii

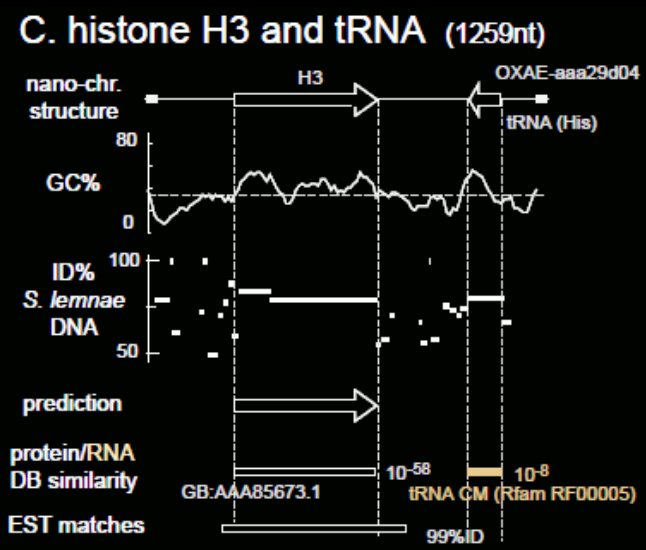
Galtier & Lobry, JME 44:632 (1997); RJ Klein et al PNAS 99:7542 (2002)

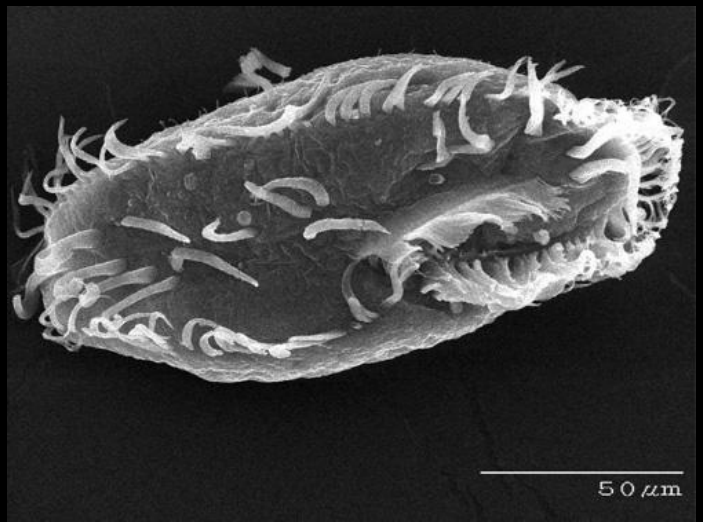
Robbie Klein, Peter Schattner

Fetch me an organism with one gene per chromosome









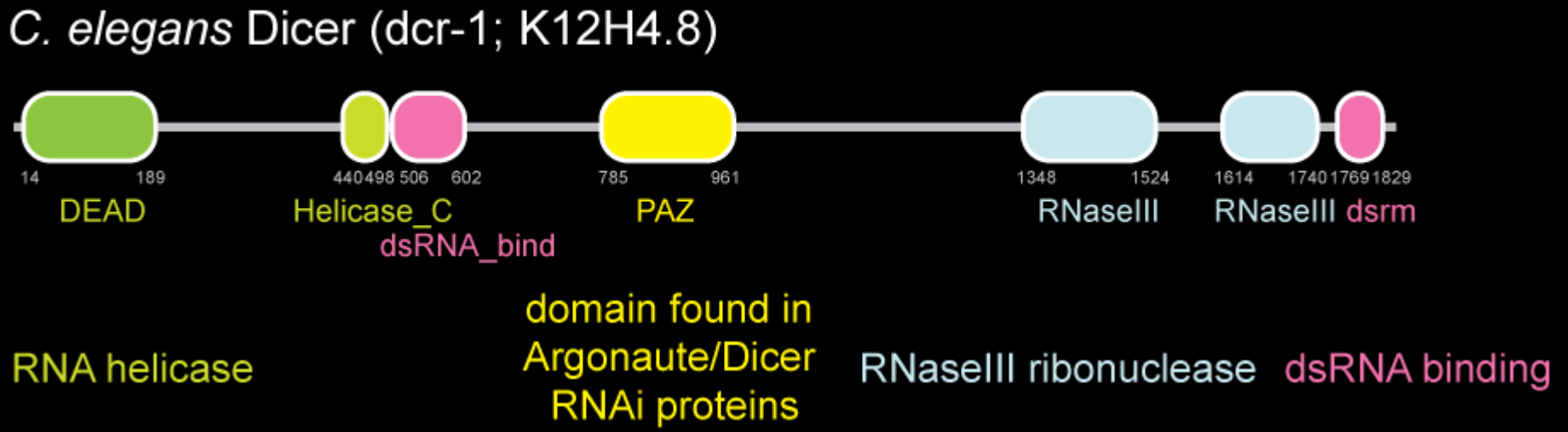
Oxytricha trifallax

Average macronuclear chromosome: 2.2kb

Seolkyoung Jung, Laura Landweber, David Prescott

Sequence analysis means putting labels on residues

... i.e. assigning *hidden states* to *observed symbols* ...



Probabilistic models of biological sequences

hidden Markov models (HMMs)

- linear sequence
- developed for digital signal processing, speech recognition
- protein and DNA analysis

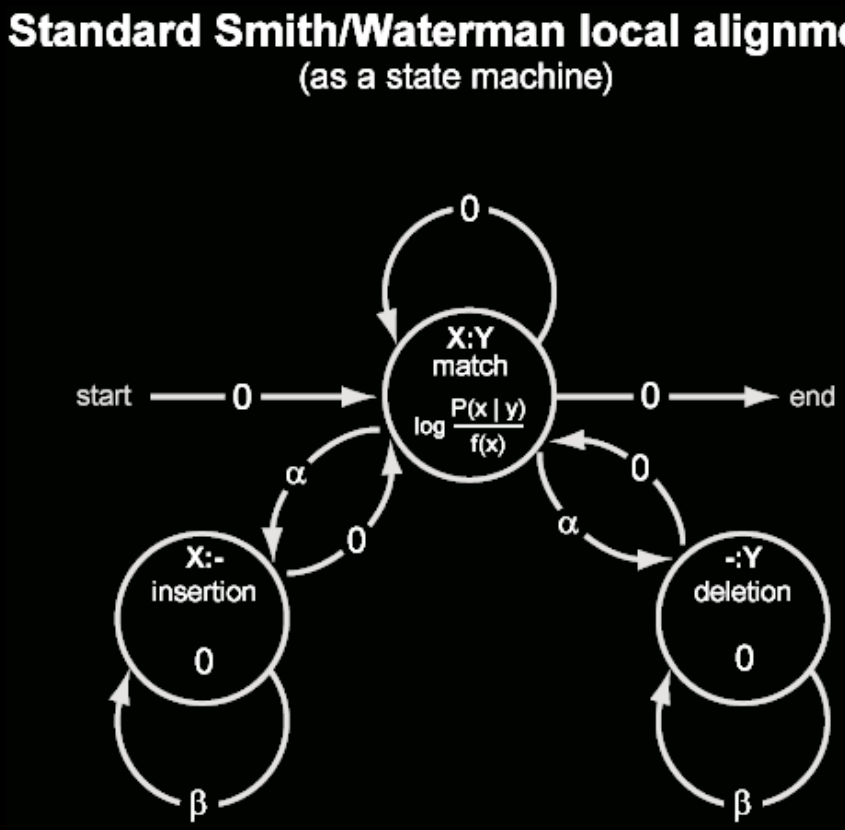
stochastic context-free grammars (SCFGs)

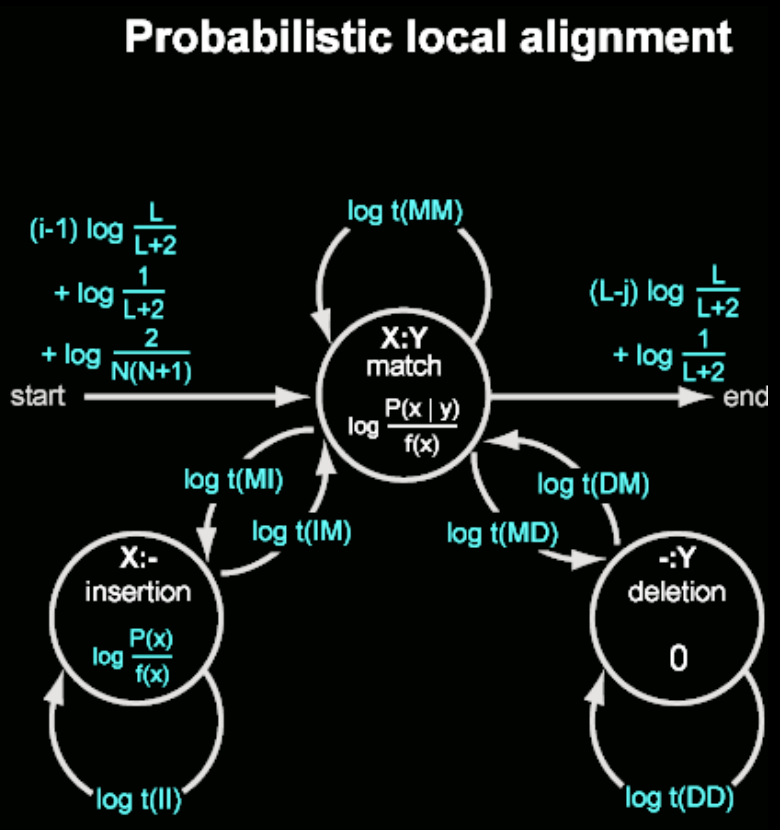
- sequence + nested pairwise correlations
- developed in computational linguistics
- RNA analysis

R Durbin, SR Eddy, GJ Mitchison, A Krogh
Biological Sequence Analysis:
Probabilistic Models of Proteins and Nucleic Acids
Cambridge Univ. Press, 1998

HMMs introduced by : Gary Churchill,
Gary Stormo, Anders Krogh, David Haussler
SCFGs introduced by : Yasu Sakakibara,
Sean Eddy, David Haussler

BLAST is an approximation to a profile HMM





S/W: Temple Smith, Michael Waterman
BLAST: Gene Myers, Warren Gish, David Lipman, Stephen Altschul, Sam Karlin, others
BLOSUM62: Steve and Jorja Henikoff
FASTA: Bill Pearson

"HMM methods are limited by computational complexity"

No; the problem is the amount of work needed to write real software.

BIOINFORMATICS ORIGINAL PAPER

Vol. 23 no. 2 2007, pages 156–161
doi:10.1093/bioinformatics/btl582

Sequence analysis

Striped Smith–Waterman speeds database searches six times over other SIMD implementations

Michael Farrar

Received on June 22, 2006; revised on November 13, 2006; accepted on November 14, 2006
Advance Access publication November 16, 2006
Associate Editor: Nikolaus Rajewsky

HMMER2 (Forward)	7 Mc/s
NCBI BLASTP	9000 Mc/s
HMMER3	4000 Mc/s

} 1000x needed (per core)

One typical database search, 400 aa protein against 10 million proteins:

HMMER2 Forward:	3000 CPU-minutes
NCBI BLASTP:	3 CPU-minutes

Why not 100 milliseconds (*interactive*)? 14 Tc/s: 2,000,000x needed (overall)

HMMER3 servers (hmmer.org) on 144 cpus: now 1 sec (200,000x)

Pfam

11,912 protein families
Cambridge | Janelia | Stockholm
Interpro Consortium: Cambridge | Geneva
pfam.janelia.org

Rfam

1,446 RNA families
Cambridge | Janelia
rfam.janelia.org

HMMER

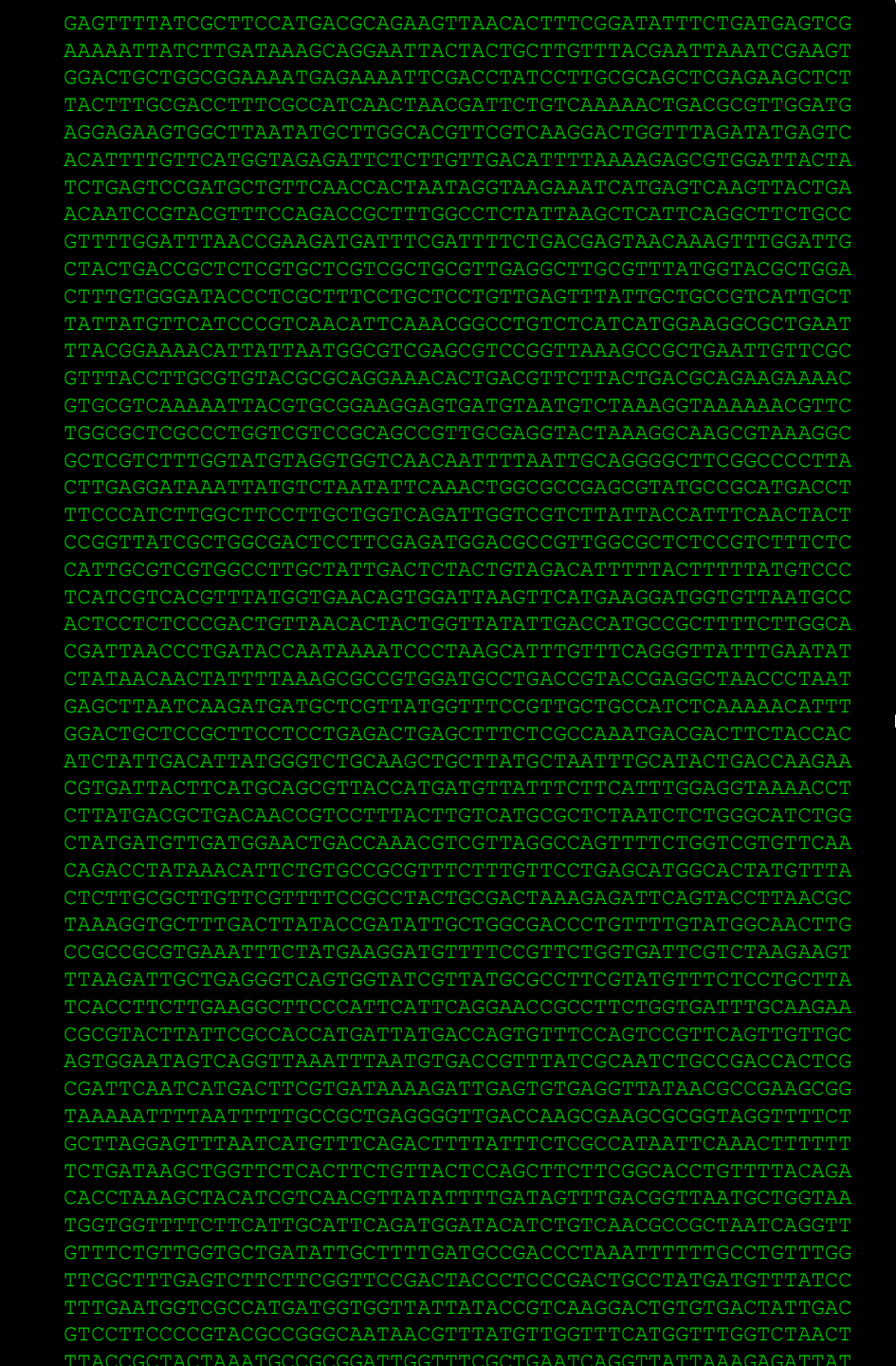
protein domains and DNA elements
sequence homology recognition
linear models: profile HMMs
44K lines ANSI C99
hmmer.org

Infernal

RNA structures
sequence homology recognition
binary tree models: profile SCFGs
104K lines ANSI C99
infernal.janelia.org

Easel

biosequence analysis code library
foundation for most lab code
59K lines ANSI C99



```
GAGTTTTATCGCTTCCATGACGCAGAAGTTAACACTTTCGGATATTTCTGATGAGTCG
AAAAATTATCTTGATAAAGCAGGAATTACTACTGCTTGTTTACGAATTAAATCGAAGT
GGACTGCTGGCGGAAAATGAGAAAATTCGACCTATCCTTGCGCAGCTCGAGAAGCTCT
TACTTTGCGACCTTTCGCCATCAACTAACGATTCTGTCAAAAACTGACGCGTTGGATG
AGGAGAAGTGGCTTAATATGCTTGGCACGTTCGTCAAGGACTGGTTTAGATATGAGTC
ACATTTTGTTCATGGTAGAGATTCTCTTGTTGACATTTTAAAAGAGCGTGGATTACTA
TCTGAGTCCGATGCTGTTCAACCACTAATAGGTAAGAAATCATGAGTCAAGTTACTGA
ACAATCCGTACGTTTCCAGACCGCTTTGGCCTCTATTAAGCTCATTCAGGCTTCTGCC
GTTTTGGATTTAACCGAAGATGATTTCGATTTTCTGACGAGTAACAAAGTTTGGATTG
CTACTGACCGCTCTCGTGCTCGTCGCTGCGTTGAGGCTTGCGTTTATGGTACGCTGGA
CTTTGTGGGATACCCTCGCTTTCCTGCTCCTGTTGAGTTTATTGCTGCCGTCATTGCT
TATTATGTTCATCCCGTCAACATTCAAACGGCCTGTCTCATCATGGAAGGCGCTGAAT
TTACGGAAAACATTATTAATGGCGTCGAGCGTCCGGTTAAAGCCGCTGAATTGTTCGC
GTTTACCTTGCGTGTACGCGCAGGAAACACTGACGTTCTTACTGACGCAGAAGAAAAC
GTGCGTCAAAAATTACGTGCGGAAGGAGTGATGTAATGTCTAAAGGTAAAAAACGTTC
TGGCGCTCGCCCTGGTCGTCCGCAGCCGTTGCGAGGTACTAAAGGCAAGCGTAAAGGC
GCTCGTCTTTGGTATGTAGGTGGTCAACAATTTTAATTGCAGGGGCTTCGGCCCCTTA
CTTGAGGATAAATTATGTCTAATATTCAAACTGGCGCCGAGCGTATGCCGCATGACCT
TTCCCATCTTGGCTTCCTTGCTGGTCAGATTGGTCGTCTTATTACCATTTCAACTACT
CCGGTTATCGCTGGCGACTCCTTCGAGATGGACGCCGTTGGCGCTCTCCGTCTTTCTC
CATTGCGTCGTGGCCTTGCTATTGACTCTACTGTAGACATTTTTACTTTTTATGTCCC
TCATCGTCACGTTTATGGTGAACAGTGGATTAAGTTCATGAAGGATGGTGTTAATGCC
ACTCCTCTCCCGACTGTTAACACTACTGGTTATATTGACCATGCCGCTTTTCTTGGCA
CGATTAACCCTGATACCAATAAAATCCCTAAGCATTTGTTTCAGGGTTATTTGAATAT
CTATAACAACTATTTTAAAGCGCCGTGGATGCCTGACCGTACCGAGGCTAACCCTAAT
GAGCTTAATCAAGATGATGCTCGTTATGGTTTCCGTTGCTGCCATCTCAAAAACATTT
GGACTGCTCCGCTTCCTCCTGAGACTGAGCTTTCTCGCCAAATGACGACTTCTACCAC
ATCTATTGACATTATGGGTCTGCAAGCTGCTTATGCTAATTTGCATACTGACCAAGAA
CGTGATTACTTCATGCAGCGTTACCATGATGTTATTTCTTCATTTGGAGGTAAAACCT
CTTATGACGCTGACAACCGTCCTTTACTTGTCATGCGCTCTAATCTCTGGGCATCTGG
CTATGATGTTGATGGAACTGACCAAACGTCGTTAGGCCAGTTTTCTGGTCGTGTTCAA
CAGACCTATAAACATTCTGTGCCGCGTTTCTTTGTTCCTGAGCATGGCACTATGTTTA
CTCTTGCGCTTGTTCGTTTTCCGCCTACTGCGACTAAAGAGATTCAGTACCTTAACGC
TAAAGGTGCTTTGACTTATACCGATATTGCTGGCGACCCTGTTTTGTATGGCAACTTG
CCGCCGCGTGAAATTTCTATGAAGGATGTTTTCCGTTCTGGTGATTCGTCTAAGAAGT
TTAAGATTGCTGAGGGTCAGTGGTATCGTTATGCGCCTTCGTATGTTTCTCCTGCTTA
TCACCTTCTTGAAGGCTTCCCATTCATTCAGGAACCGCCTTCTGGTGATTTGCAAGAA
CGCGTACTTATTCGCCACCATGATTATGACCAGTGTTTCCAGTCCGTTCAGTTGTTGC
AGTGGAATAGTCAGGTTAAATTTAATGTGACCGTTTATCGCAATCTGCCGACCACTCG
CGATTCAATCATGACTTCGTGATAAAAGATTGAGTGTGAGGTTATAACGCCGAAGCGG
TAAAAATTTTAATTTTTGCCGCTGAGGGGTTGACCAAGCGAAGCGCGGTAGGTTTTCT
GCTTAGGAGTTTAATCATGTTTCAGACTTTTATTTCTCGCCATAATTCAAACTTTTTT
TCTGATAAGCTGGTTCTCACTTCTGTTACTCCAGCTTCTTCGGCACCTGTTTTACAGA
CACCTAAAGCTACATCGTCAACGTTATATTTTGATAGTTTGACGGTTAATGCTGGTAA
TGGTGGTTTTCTTCATTGCATTCAGATGGATACATCTGTCAACGCCGCTAATCAGGTT
GTTTCTGTTGGTGCTGATATTGCTTTTGATGCCGACCCTAAATTTTTTGCCTGTTTGG
TTCGCTTTGAGTCTTCTTCGGTTCCGACTACCCTCCCGACTGCCTATGATGTTTATCC
TTTGAATGGTCGCCATGATGGTGGTTATTATACCGTCAAGGACTGTGTGACTATTGAC
GTCCTTCCCCGTACGCCGGGCAATAACGTTTATGTTGGTTTCATGGTTTGGTCTAACT
TTACCGCTACTAAATGCCGCGGATTGGTTTCGCTGAATCAGGTTATTAAAGAGATTAT
```

```
for (i = 1; i <= L; i++)
  {
    rsc   = om->rf[dsq[i]];
    tsc   = om->tf;
    dcv   = infv;
    xEv   = infv;
    Dmaxv = infv;
    xBv   = _mm_set1_ps(xB);

    /* Right shifts by 4 bytes. 4,8,12,x becomes x,4,8,12.
     */
    mpv = MMX(Q-1);  mpv = _mm_shuffle_ps(mpv, mpv, _MM_SHUFFLE(2, 1, 0, 0));   mpv = _mm_move_ss(mpv, infv);
    dpv = DMX(Q-1);  dpv = _mm_shuffle_ps(dpv, dpv, _MM_SHUFFLE(2, 1, 0, 0));   dpv = _mm_move_ss(dpv, infv);
    ipv = IMX(Q-1);  ipv = _mm_shuffle_ps(ipv, ipv, _MM_SHUFFLE(2, 1, 0, 0));   ipv = _mm_move_ss(ipv, infv);

    for (q = 0; q < Q; q++)
      {
        /* Calculate new MMX(i,q); don't store it yet, hold it in sv. */
        sv   =                _mm_add_ps(xBv, *tsc);  tsc++;
        sv   = _mm_max_ps(sv, _mm_add_ps(mpv, *tsc)); tsc++;
        sv   = _mm_max_ps(sv, _mm_add_ps(ipv, *tsc)); tsc++;
        sv   = _mm_max_ps(sv, _mm_add_ps(dpv, *tsc)); tsc++;
        sv   = _mm_add_ps(sv, *rsc);                  rsc++;
        xEv  = _mm_max_ps(xEv, sv);

        /* Load {MDI}(i-1,q) into mpv, dpv, ipv;
         * {MDI}MX(q) is then the current, not the prev row
         */
        mpv = MMX(q);
        dpv = DMX(q);
        ipv = IMX(q);

        /* Do the delayed stores of {MD}(i,q) now that memory is usable */
        MMX(q) = sv;
        DMX(q) = dcv;

        /* Calculate the next D(i,q+1) partially: M->D only;
         * delay storage, holding it in dcv
         */
        dcv   = _mm_add_ps(sv, *tsc); tsc++;
        Dmaxv = _mm_max_ps(dcv, Dmaxv);

        /* Calculate and store I(i,q) */
        sv   =                _mm_add_ps(mpv, *tsc);  tsc++;
        sv   = _mm_max_ps(sv, _mm_add_ps(ipv, *tsc)); tsc++;
        IMX(q) = _mm_add_ps(sv, *rsc);                rsc++;
      }

    /* Now the "special" states, which start from Mk->E (->C, ->J->B) */
    /* The following incantation takes the max of xEv's elements  */
    xEv = _mm_max_ps(xEv, _mm_shuffle_ps(xEv, xEv, _MM_SHUFFLE(0, 3, 2, 1)));
    xEv = _mm_max_ps(xEv, _mm_shuffle_ps(xEv, xEv, _MM_SHUFFLE(1, 0, 3, 2)));
    _mm_store_ss(&xE, xEv);

    xN = xN +  om->xf[p7O_N][p7O_LOOP];
    xC = ESL_MAX(xC + om->xf[p7O_C][p7O_LOOP],  xE + om->xf[p7O_E][p7O_MOVE]);
    xJ = ESL_MAX(xJ + om->xf[p7O_J][p7O_LOOP],  xE + om->xf[p7O_E][p7O_LOOP]);
    xB = ESL_MAX(xJ + om->xf[p7O_J][p7O_MOVE],  xN + om->xf[p7O_N][p7O_MOVE]);
    /* and now xB will carry over into next i, and xC carries over after i=L */


    /* Finally the "lazy F" loop (sensu [Farrar07]). We can often
     * prove that we don't need to evaluate any D->D paths at all.
     *
     * The observation is that if we can show that on the next row,
     * B->M(i+1,k) paths always dominate M->D->...->D->M(i+1,k) paths
     * for all k, then we don't need any D->D calculations.
     *
     * The test condition is:
     *      max_k D(i,k) + max_k ( TDD(k-2) + TDM(k-1) - TBM(k) ) < xB(i)
     * So:
     *   max_k (TDD(k-2) + TDM(k-1) - TBM(k)) is precalc'ed in om->dd_bound;
     *   max_k D(i,k) is why we tracked Dmaxv;
     *   xB(i) was just calculated above.
     */
    Dmaxv = _mm_max_ps(Dmaxv, _mm_shuffle_ps(Dmaxv, Dmaxv, _MM_SHUFFLE(0, 3, 2, 1)));
    Dmaxv = _mm_max_ps(Dmaxv, _mm_shuffle_ps(Dmaxv, Dmaxv, _MM_SHUFFLE(1, 0, 3, 2)));
    _mm_store_ss(&Dmax, Dmaxv);
    if (Dmax + om->ddbound_f > xB)
      {
        /* Now we're obligated to do at least one complete DD path to be sure. */
        /* dcv has carried through from end of q loop above */
        dcv = _mm_shuffle_ps(dcv, dcv, _MM_SHUFFLE(2, 1, 0, 0));
        dcv = _mm_move_ss(dcv, infv);
        tsc = om->tf + 7*Q;   /* set tsc to start of the DD's */
        for (q = 0; q < Q; q++)
          {
            DMX(q) = _mm_max_ps(dcv, DMX(q));
            dcv    = _mm_add_ps(DMX(q), *tsc); tsc++;
          }

        /* We may have to do up to three more passes; the check
         * is for whether crossing a segment boundary can improve
         * our score.
         */
        do {
          dcv = _mm_shuffle_ps(dcv, dcv, _MM_SHUFFLE(2, 1, 0, 0));
          dcv = _mm_move_ss(dcv, infv);
          tsc = om->tf + 7*Q; /* set tsc to start of the DD's */
          for (q = 0; q < Q; q++)
            {
              if (! sse_any_gt_ps(dcv, DMX(q))) break;
              DMX(q) = _mm_max_ps(dcv, DMX(q));
              dcv    = _mm_add_ps(DMX(q), *tsc);   tsc++;
            }
        } while (q == Q);
      }
    else
```

The Eddy/Rivas laboratory

Elena Rivas
Tom Jones
Eric Nawrocki
Lee Henry
Fred Davis
Travis Wheeler
Pat Dennis
Seolkyoung Jung

HHMI Janelia Farm
http://selab.janelia.org

The Eddy/Rivas laboratory

Elena Rivas
Tom Jones
Eric Nawrocki
Lee Henry
Fred Davis
Travis Wheeler
Pat Dennis
Seolkyoung Jung

HHMI Janelia Farm
http://selab.janelia.org

"MatrixNucleus" image from Angus Lamond's lab, Univ. of Dundee, http://www.lamondlab.com