primer3 release 1.1.3

INTRODUCTION

Primer3 picks primers for PCR reactions, considering as criteria:

o oligonucleotide melting temperature, size, GC content,
  and primer-dimer possibilities,

o PCR product size,

o positional constraints within the source (template) sequence, and

o possibilities for ectopic priming (amplifying the wrong sequence)

o many other constraints.

All of these criteria are user-specifiable as constraints, and
some are specifiable as terms in an objective function that
characterizes an optimal primer pair.

Whitehead Institute for Biomedical Research provides a web-based
front end to primer3 at
http://fokker.wi.mit.edu/cgi-bin/primer3/primer3_www.cgi

CITING PRIMER3

We request but do not require that use of this software be cited in
publications as

Steve Rozen and Helen J. Skaletsky (2000) Primer3 on the WWW for
general users and for biologist programmers.
In: Krawetz S, Misener S (eds) Bioinformatics Methods and
Protocols: Methods in Molecular Biology. Humana Press, Totowa,
NJ, pp 365-386
Source code available at http://sourceforge.net/projects/primer3/

The paper above is available at
http://jura.wi.mit.edu/rozen/papers/rozen-and-skaletsky-2000-primer3.pdf

REPORTING BUGS AND PROBLEMS AND SUGGESTING ENHANCEMENTS

For error reports or requests for enhancements, please send e-mail
to primer3-mail (at) lists.sourceforge.net after replacing (at)
with @.

INSTALLATION INSTRUCTIONS

Unzip and untar the distribution.

DO NOT do this on a PC -- primer3_core will not compile if pc

newlines get inserted into the source files. Instead, move the
distribution (primer3_<release>.tar.gz) to Unix, and then

```
$ unzip primer3_1.0.1.tar.gz
$ tar xvf primer3_1.0.1.tar
$ cd primer3_1.0.1/src

If you do not use gcc, modify the makefile to
  use your (ANSI) C compiler and appropriate
  compile and link flags.

$ make all

# Warnings about pr_release being unused are harmless.
# You should have created executables primer3_core, ntdpal,
#  olgotm, and long_seq_tm_test

$ make test

# You should not see 'FAILED' during the tests.
```

If your perl command is not called perl (for example, if it is
called perl5) you will have to modify the
Makefile in the test/ directory.

ntdpal (NucleoTide Dynamic Programming ALignment) is a
stand-alone program that provides primer3's alignment
functionality (local, a.k.a. Smith-Waterman, global,
a.k.a. Needleman-Wunsch, plus "half global"). It is provided
strictly as is; for further documentation please see the code.

BUILDING OSX UNIVERSAL BINARY

** To build a processor-native, non-universal binary of primer3,
the following is unneccesary**.

The instructions above should be sufficient.

A pre-compiled, universal binary download for OSX is available from
http://sourceforge.net/projects/primer3/ for the current release.

These instructions assume you want to build binaries compatible
with *both* of the current processor architectures used by the Apple
platform (i,e. the binaries will be run on both PPC and intel platforms).

Provided you have the OS X developer tools installed
(you can download from http://developer.apple.com after
registering for a free account), you can compile a universal
build (intel and PPC native) of primer3.

o you must be running OS X > 10.4 and should have the most
        recent version of XCode
o run `make -f Makefile.OSX.Leopard all` if you run OS X 10.5
o run `make -f Makefile.OSX.Tiger all` is you run OS X 10.4
o run the tests as directed above

Additional instructions for 'installing' the binaries may be found in
the README.OSX.txt.

You should be able to compile a 3-way binary which includes PPC64 support
(intel, PPC, PPC64) by adding the `-arch ppc64` flag to the
end of both the CFLAGS and LDFLAGS lines at the top of Makefile.OSX.
This has not been tested.

SYSTEM REQUIREMENTS

Please see http://sourceforge.net/projects/primer3/ for up-to-date
information. Primer3 should compile on any Linux/Unix system
including MacOS 10 and on other systems with POSIX C
(e.g. MSWindows). The Makefile may need to be modified for
compilation with C compilers other than gcc. Our hope is to
distribute binarie for SourceForge in the near future. Primer3
still uses many Kernighan-&-Richie-style function headers, so
you might have to force your compiler to accept them.

INVOKING primer3_core

By default, the executable program produced by the Makefile is
called primer3_core. This is the C program that does the heavy
lifting of primer picking. There is also a more user-friendly
web interface (distributed separately).

The command line for primer3 is:

primer3_core [-format_output] [-strict_tags] < input_file.txt

-format_output indicates that primer3_core should generate
   user-oriented (rather than program-oriented) output.

-strict_tags indicates that primer3_core should generate
   a fatal error if there is any tag in the input that
   it does not recognize (see INPUT AND OUTPUT CONVENTIONS).

WARNING: primer3_core only reads its input on stdin, so the usual
unix convention of

primer3_core input_file.txt

will not work. Primer3_core will just sit there forever
waiting for its input on stdin.

Note: The old flag -2x_compat is no longer supported.

INPUT AND OUTPUT CONVENTIONS

By default, primer3 accepts input in Boulder-io format, a
pre-XML, pre-RDF, text-based input/output format for
program-to-program data interchange. By default, primer3 also
produces output in the same format.

When run with the -format_output command-line flag, primer3
prints a more user-oriented report for each sequence.

Primer3 exits with 0 status if it operates correctly. See EXIT
STATUS CODES below for additional information.

The syntax of the version of Boulder-io recognized by primer3 is
as follows:

- Input consists of a sequence of RECORDs.

- A RECORD consists of a sequence of (TAG,VALUE) pairs, each terminated
  by a newline character (\n). A RECORD is terminated by '='
  appearing by itself on a line.

- A (TAG,VALUE) pair has the following requirements:

    - the TAG must be immediately (without spaces)
      followed by '='.
    - the pair must be terminated by a newline character.

An example of a legal (TAG,VALUE) pair is

```
PRIMER_SEQUENCE_ID=my_marker
```

and an example of a BOULDER-IO record is

```
PRIMER_SEQUENCE_ID=test1
SEQUENCE=GACTGATCGATGCTAGCTACGATCGATCGATGCATGCTAGCTAGCTAGCTGCTAGC
=
```

Many records can be sent, one after another. Below is an example of three different records which might be passed through a boulder-io stream:

```
PRIMER_SEQUENCE_ID=test1
SEQUENCE=GACTGATCGATGCTAGCTACGATCGATCGATGCATGCTAGCTAGCTAGCTGCTAGC
=
PRIMER_SEQUENCE_ID=test2
SEQUENCE=CATCATCATCATCGATGCTAGCATCNNACGTACGANCANATGCATCGATCGT
=
PRIMER_SEQUENCE_ID=test3
SEQUENCE=NACGTAGCTAGCATGCACNACTCGACNACGATGCACNACAGCTGCATCGATGC
=
```

Primer3 reads boulder-io on stdin and echos its input and returns results in boulder-io format on stdout. Primer3 indicates many user-correctable errors by a value in the PRIMER_ERROR tag (see below) and indicates other errors, including system configuration errors, resource errors (such out-of-memory errors), and detected programming errors by a message on stderr and a non-zero exit status.

Below is the list of input tags that primer3 recognizes. Primer3 echos and ignores any tags it does not recognize, unless the -strict_tags flag is set on the command line, in which case primer3 prints an error in the PRIMER_ERROR output tag (see below), and prints additional information on stdout; this option can be useful for debugging systems that incorporate primer.

Except for tags with the type "interval list" each tag is allowed only ONCE in any given input record. This restriction is not systematically checked in this beta release: use care.

There are 2 major classes of input tags. "Sequence" input tags describe a particular input sequence to primer3, and are reset after every boulder record. "Global" input tags describe the general parameters that primer3 should use in its searches, and

the values of these tags persist between input boulder records until or unless they are explicitly reset. Errors in "Sequence" input tags invalidate the current record, but primer3 will continue to process additional records. Errors in "Global" input tags are fatal because they invalidate the basic conditions under which primers are being picked.

"Sequence" Input Tags

PRIMER_SEQUENCE_ID (string, optional)

(MARKER_NAME is a deprecated synonym maintained for v2 compatibility.)

An identifier that is reproduced in the output to enable users to identify the source of the chosen primers.

This tag must be present if PRIMER_FILE_FLAG is non-zero.

SEQUENCE (nucleotide sequence, REQUIRED)

The sequence from which to choose primers. The sequence must be presented 5' -> 3' (see the discussion of the PRIMER_SELF_END argument). The bases may be upper or lower case. No newlines should be inserted into the sequence, because the Boulder-IO parser will assume that a line ends at a newline.

INCLUDED_REGION (interval, optional)

A sub-region of the given sequence in which to pick primers. For example, often the first dozen or so bases of a sequence are vector, and should be excluded from consideration. The value for this parameter has the form

<start>,<length>

where <start> is the index of the first base to consider, and <length> is the number of subsequent bases in the primer-picking region.

TARGET (interval list, default empty)

If one or more Targets is specified then a legal primer pair must flank at least one of them. A Target might be a simple sequence

repeat site (for example a CA repeat) or a single-base-pair polymorphism. The value should be a space-separated list of

<start>,<length>

pairs where <start> is the index of the first base of a Target, and <length> is its length.

For backward compatibility primer3 accepts (but ignores) a trailing ,<description> for each element of this argument.

EXCLUDED_REGION (interval list, default empty)

Primer oligos may not overlap any region specified in this tag. The associated value must be a space-separated list of

<start>,<length>

pairs where <start> is the index of the first base of the excluded region, and <length> is its length. This tag is useful for tasks such as excluding regions of low sequence quality or for excluding regions containing repetitive elements such as ALUs or LINEs.

PRIMER_COMMENT (string, optional)

The value of this tag is ignored.

COMMENT (string, optional)

Deprecated synonym for PRIMER_COMMENT.

PRIMER_SEQUENCE_QUALITY (quality list, default empty)

A list of space separated integers. There must be exactly one integer for each base in SEQUENCE if this argument is non-empty. For example, for the sequence ANNTTCA... PRIMER_SEQUENCE_QUALITY might be 45 10 0 50 30 34 50 67 .... High numbers indicate high confidence in the base called at that position and low numbers indicate low confidence in the base call at that position. This parameter is only relevant if you are using a base calling program that provides quality information (for example phred).

PRIMER_LEFT_INPUT (nucleotide sequence, default empty)

The sequence of a left primer to check and around which to design right primers and optional internal oligos. Must be a substring of SEQUENCE.

PRIMER_RIGHT_INPUT (nucleotide sequence, default empty)

The sequence of a right primer to check and around which to design left primers and optional internal oligos. Must be a substring of the reverse strand of SEQUENCE.

PRIMER_START_CODON_POSITION (int, default -1000000)

This parameter should be considered EXPERIMENTAL at this point. Please check the output carefully; some erroneous inputs might cause an error in primer3.

Index of the first base of a start codon. This parameter allows primer3 to select primer pairs to create in-frame amplicons e.g. to create a template for a fusion protein. Primer3 will attempt to select an in-frame left primer, ideally starting at or to the left of the start codon, or to the right if necessary. Negative values of this parameter are legal if the actual start codon is to the left of available sequence. If this parameter is non-negative primer3 signals an error if the codon at the position specified by this parameter is not an ATG. A value less than or equal to -10^6 indicates that primer3 should ignore this parameter.

Primer3 selects the position of the right primer by scanning right from the left primer for a stop codon. Ideally the right primer will end at or after the stop codon.

"Global" Input Tags

PRIMER_PICK_ANYWAY (boolean, default 0)

If true pick a primer pair even if PRIMER_LEFT_INPUT, PRIMER_RIGHT_INPUT, or PRIMER_INTERNAL_OLIGO_INPUT violates specific constraints.

PRIMER_MISPRIMING_LIBRARY (string, optional)

The name of a file containing a nucleotide sequence library of sequences to avoid amplifying (for example repetitive sequences, or

possibly the sequences of genes in a gene family that should not be amplified.) The file must be in (a slightly restricted) FASTA format (W. B. Pearson and D.J. Lipman, PNAS 85:8 pp 2444-2448 [1988]); we briefly discuss the organization of this file below. If this parameter is specified then primer3 locally aligns each candidate primer against each library sequence and rejects those primers for which the local alignment score times a specified weight (see below) exceeds PRIMER_MAX_MISPRIMING. (The maximum value of the weight is arbitrarily set to 100.0.)

Each sequence entry in the FASTA-format file must begin with an "id line" that starts with '>'. The contents of the id line is "slightly restricted" in that primer3 parses everything after any optional asterisk ('*') as a floating point number to use as the weight mentioned above. If the id line contains no asterisk then the weight defaults to 1.0. The alignment scoring system used is the same as for calculating complementarity among oligos (e.g. PRIMER_SELF_ANY), except for the handling of IUB/IUPAC ambiguity codes (discussed below).

The remainder of an entry contains the sequence as lines following the id line up until a line starting with '>' or the end of the file. Whitespace and newlines are ignored. Characters 'A', 'T', 'G', 'C', 'a', 't', 'g', 'c' and IUB/IUPAC 'ambiguity' codes ('R, 'Y', 'K', 'M', 'S', 'W', 'N', including lower case) are retained. For technical reasons the length of the sequence must be >= 3. Of course, sequences of length < 10 or so are probably useless, but will be accepted without complaint.

WARNING: always set PRIMER_LIB_AMBIGUITY_CODES_CONSENSUS=0 if any sequence in the library contains strings of 'N's: NNNNNNNNNNNNNNNNNNN.
NOWWW
There are no restrictions on line length.

An empty value for this parameter indicates that no repeat library should be used and "turns off" the use of a previously specified library.

Repbase (J. Jurka, A.F.A. Smit, C. Pethiyagoda, and others, 1995-1996, ftp://ncbi.nlm.nih.gov/repository/repbase) is an excellent source of repeat sequences and pointers to the literature. (The Repbase files need to be converted to Fasta format before they can be used by primer3.)

PRIMER_LIB_AMBIGUITY_CODES_CONSENSUS (boolean, default 1)

If set to 1, treat ambiguity codes as if they were consensus codes when matching oligos to mispriming or mishyb libraries. For example, if this flag is set, then a C in an oligo will be scored as a perfect match to an S in a library sequence, as will a G in the oligo. More importantly, though, any base in an oligo will be scored as a perfect match to an N in the library. This is very bad if the library contains strings of Ns, as no oligo will be legal (and it will take a long time to find this out). So unless you know for sure that your library does not have runs of Ns (or Xs), then set this flag to 0.

PRIMER_MAX_MISPRIMING (decimal,9999.99, default 12.00)

The maximum allowed weighted similarity with any sequence in PRIMER_MISPRIMING_LIBRARY.

PRIMER_MAX_TEMPLATE_MISPRIMING (decimal,9999.99, default -1.00)

The maximum allowed similarity to ectopic sites in the template. A negative value means do not check. The scoring system is the same as used for PRIMER_MAX_MISPRIMING, except that an ambiguity code in the template is never treated as a consensus (see PRIMER_LIB_AMBIGUITY_CODES_CONSENSUS).

PRIMER_PAIR_MAX_MISPRIMING (decimal,9999.99, default 24.00)

The maximum allowed sum of similarities of a primer pair (one similarity for each primer) with any single sequence in PRIMER_MISPRIMING_LIBRARY.
Library sequence weights are not used in computing the sum of similarities.

PRIMER_PAIR_MAX_TEMPLATE_MISPRIMING (decimal,9999.99, default -1.00)

The maximum allowed summed similarity of both primers to ectopic sites in the template. A negative value means do not check. The scoring system is the same as used for PRIMER_PAIR_MAX_MISPRIMING, except that an ambiguity code in the template is never treated as a consensus (see PRIMER_LIB_AMBIGUITY_CODES_CONSENSUS). Primer3 does not check the similarity of hybridization oligos (internal oligos) to locations outside of the amplicon.

PRIMER_PRODUCT_MAX_TM (float, default 1000000.0)

The maximum allowed melting temperature of the amplicon. Primer3 calculates product Tm calculated using the formula from Bolton and McCarthy, PNAS 84:1390 (1962) as presented in Sambrook, Fritsch and Maniatis, Molecular Cloning, p 11.46 (1989, CSHL Press).

```
   Tm = 81.5 + 16.6(log10([Na+])) + .41*(%GC) - 600/length
```

Where [Na+] is the molar sodium concentration, (%GC) is the percent of Gs and Cs in the sequence, and length is the length of the sequence.

A similar formula is used by the prime primer selection program in GCG (http://www.gcg.com), which instead uses 675.0 / length in the last term (after F. Baldino, Jr, M.-F. Chesselet, and M.E. Lewis, Methods in Enzymology 168:766 (1989) eqn (1) on page 766 without the mismatch and formamide terms). The formulas here and in Baldino et al. assume Na+ rather than K+. According to J.G. Wetmur, Critical Reviews in BioChem. and Mol. Bio. 26:227 (1991) 50 mM K+ should be equivalent in these formulae to .2 M Na+. Primer3 uses the same salt concentration value for calculating both the primer melting temperature and the oligo melting temperature. If you are planning to use the PCR product for hybridization later this behavior will not give you the Tm under hybridization conditions.

PRIMER_PRODUCT_MIN_TM (float, default -1000000.0)

The minimum allowed melting temperature of the amplicon. Please see the documentation on the maximum melting temperature of the product for details.

PRIMER_EXPLAIN_FLAG (boolean, default 0)

If this flag is non-0, produce PRIMER_LEFT_EXPLAIN, PRIMER_RIGHT_EXPLAIN, and PRIMER_INTERNAL_OLIGO_EXPLAIN output tags, which are intended to provide information on the number of oligos and primer pairs that primer3 examined, and statistics on the number discarded for various reasons. If -format_output is set similar information is produced in the user-oriented output.

PRIMER_PRODUCT_SIZE_RANGE (size range list, default 100-300)

The associated values specify the lengths of the product that the user wants the primers to create, and is a space separated list of elements of the form

<x>-<y>

where an <x>-<y> pair is a legal range of lengths for the product. For example, if one wants PCR products to be between 100 to 150 bases (inclusive) then one would set this parameter to 100-150. If one desires PCR products in either the range from 100 to 150 bases or in the range from 200 to 250 bases then one would set this parameter to 100-150 200-250.

Primer3 favors ranges to the left side of the parameter string. Primer3 will return legal primers pairs in the first range regardless the value of the objective function for these pairs. Only if there are an insufficient number of primers in the first range will primer3 return primers in a subsequent range.

For those with primarily a computational background, the PCR product size is size (in base pairs) of the DNA fragment that would be produced by the PCR reaction on the given sequence template. This would, of course, include the primers themselves.

PRIMER_PICK_INTERNAL_OLIGO (boolean, default 0)

If the associated value is non-0, then primer3 will attempt to pick an internal oligo (hybridization probe to detect the PCR product). This tag is maintained for backward compatibility. Use PRIMER_TASK.

PRIMER_GC_CLAMP (int, default 0)

Require the specified number of consecutive Gs and Cs at the 3' end of both the left and right primer. (This parameter has no effect on the internal oligo if one is requested.)

PRIMER_OPT_SIZE (int, default 20)

Optimum length (in bases) of a primer oligo. Primer3 will attempt to pick primers close to this length.

PRIMER_DEFAULT_SIZE (int, default 20)

A deprecated synonym for PRIMER_OPT_SIZE, maintained for v2

compatibility.

PRIMER_MIN_SIZE (int, default 18)

Minimum acceptable length of a primer. Must be greater than 0 and less than or equal to PRIMER_MAX_SIZE.

PRIMER_MAX_SIZE (int, default 27)

Maximum acceptable length (in bases) of a primer. Currently this parameter cannot be larger than 35. This limit is governed by maximum oligo size for which primer3's melting-temperature is valid.

PRIMER_OPT_TM (float, default 60.0C)

Optimum melting temperature(Celsius) for a primer oligo. Primer3 will try to pick primers with melting temperatures are close to this temperature. The oligo melting temperature formula used can be specified by user. Please see PRIMER_TM_SANTALUCIA for more information.

PRIMER_MIN_TM (float, default 57.0C)

Minimum acceptable melting temperature(Celsius) for a primer oligo.

PRIMER_MAX_TM (float, default 63.0C)

Maximum acceptable melting temperature(Celsius) for a primer oligo.

PRIMER_MAX_DIFF_TM (float, default 100.0C)

Maximum acceptable (unsigned) difference between the melting temperatures of the left and right primers.

PRIMER_TM_SANTALUCIA (int, default 0)

Specifies details of melting temperature calculation. (New in v. 1.1.0, added by Maido Remm and Triinu Koressaar.)

A value of 1 (*RECOMMENDED*) directs primer3 to use the table of thermodynamic values and the method for melting temperature calculation suggested in the paper [SantaLucia JR (1998) "A unified view of polymer, dumbbell and oligonucleotide DNA nearest-neighbor thermodynamics", Proc Natl Acad Sci 95:1460-65 http://dx.doi.org/10.1073/pnas.95.4.1460].

A value of 0 directs primer3 to a backward compatible calculation (in other words, the only calculation availble in previous version of primer3).

This backward compatible calculation uses the table of thermodynamic parameters in the paper [Breslauer KJ, Frank R, Blˆcker H and Marky LA (1986) "Predicting DNA duplex stability from the base sequence" Proc Natl Acad Sci 83:4746-50 http://dx.doi.org/10.1073/pnas.83.11.3746], and the method in the paper [Rychlik W, Spencer WJ and Rhoads RE (1990) "Optimization of the annealing temperature for DNA amplification in vitro", Nucleic Acids Res 18:6409-12 http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pubmed&pubmedid=2243783].

Use tag PRIMER_SALT_CORRECTIONS, to specify the salt correction method for melting temperature calculation.

Example of calculating the melting temperature of an oligo if PRIMER_TM_SANTALUCIA=1 and PRIMER_SALT_CORRECTIONS=1 recommended values):

primer=CGTGACGTGACGGACT

Using default salt and DNA concentrations we have

Tm = deltaH/(deltaS + R*ln(C/4)),

where R is the gas constant (1.987 cal/K mol)
and C is the DNA concentration.

deltaH(predicted) =

  = dH(CG) + dH(GT) + dH(TG) + .. + dH(CT) +
     + dH(init.w.term.GC) + dH(init.w.term.AT) =

  = -10.6 + (-8.4) + (-8.5) + .. + (-7.8) + 0.1 + 2.3 =

  = -128.8 kcal/mol

where 'init.w.term GC' and 'init.w.term AT' are two
initiation parameters for duplex formation: 'initiation with
terminal GC' and 'initiation with terminal AT'

deltaS(predicted) =

  = dS(CG) + dS(GT) + dS(TG) + .. + dS(CT) +
    + dS(init.w.term.GC) + dS(init.w.term.AT) =

  = -27.2 + (-22.4) + (-22.7) + .. + (-21.0) + (-2.8) + 4.1 =

  = -345.2 cal/k*mol

deltaS(salt corrected) =
  = deltaS(predicted) + 0.368*15(NN pairs)*ln(0.05M monovalent cations) =
  = -361.736

Tm = -128.800/(-361.736+1.987*ln((5*10^(-8))/4)) =
   = 323.704 K

Tm(C) = 323.704 - 273.15 = 50.554 C

PRIMER_SALT_CONC (float, default 50.0 mM)

The millimolar concentration of monovalent salt cations (usually KCl) in
the PCR.
Primer3 uses this argument to calculate oligo and primer melting
temperatures. Use tag PRIMER_DIVALENT_CONC to specify the concentration
of divalent cations (in this case you also should use tag
PRIMER_DNTP_CONC).

PRIMER_DIVALENT_CONC (float, default 0.0 mM)

The millimolar concentration of divalent salt cations (usually MgCl^(2+))
in
the PCR. (New in v. 1.1.0, added by Maido Remm and Triinu Koressaar)

Primer3 converts concentration of divalent cations to concentration
of monovalent cations using formula suggested in the paper [Ahsen von N,
Wittwer CT, Schutz E (2001) "Oligonucleotide Melting Temperatures under PCR
Conditions: Nearest-Neighbor Corrections for Mg^(2+), Deoxynucleotide
Triphosphate,
and Dimethyl Sulfoxide Concentrations with Comparision to Alternative
Empirical

Formulas", Clinical Chemistry 47:1956-61 http://www.clinchem.org/cgi/content/full/47/11/1956].

[Monovalent cations] = [Monovalent cations] + 120*(([divalent cations] - [dNTP])^0.5)

According to the formula concentration of desoxynucleotide triphosphate [dNTP] must be smaller than concentration of divalent cations. If the specified concentration of dNTPs is larger than specified concentration of divalent cations then the effect of divalent cations is not considered. The concentration of dNTPs is included to the formula beacause of some magnesium is bound by the
dNTP. Attained concentration of monovalent cations is used to calculate oligo/primer
melting temperature. Use tag PRIMER_DNTP_CONC to specify the concentration of dNTPs.

PRIMER_DNTP_CONC (float, default 0.0 mM)

The millimolar concentration of deoxyribonucleotide triphosphate. This argument is considered only if PRIMER_DIVALENT_CONC is specified. See PRIMER_DIVALENT_CONC.

PRIMER_SALT_CORRECTIONS (int, default 0)

Specifies the salt correction formula for the melting temperature calculation. (New in v. 1.1.0, added by Maido Remm and Triinu Koressaar)

A value of 1 (*RECOMMENDED*) directs primer3 to use the salt correction formula in the paper [SantaLucia JR (1998) "A unified view of polymer, dumbbell and oligonucleotide DNA nearest-neighbor thermodynamics", Proc Natl Acad Sci 95:1460-65
http://dx.doi.org/10.1073/pnas.95.4.1460]

A value of 0 directs primer3 to use the the salt correction formula in the paper [Schildkraut, C, and Lifson, S (1965) "Dependence of the melting temperature of DNA on salt concentration", Biopolymers 3:195-208 (not available on-line)]. This was the formula used in previous version of primer3.

A value of 2 directs primer3 to use the salt correction formula in the paper [Owczarzy R, You Y, Moreira BG, Manthey JA, Huang L, Behlke MA and Walder JA (2004) "Effects of sodium ions on DNA duplex oligomers: Improved predictions of melting temperatures", Biochemistry 43:3537-54 http://dx.doi.org/10.1021/bi034621r].

PRIMER_LOWERCASE_MASKING (int, default 0)

This option allows for intelligent design of primers in sequence in which masked regions (for example repeat-masked regions) are lower-cased. (New in v. 1.1.0, added by Maido Remm and Triinu Koressaar)

A value of 1 directs primer3 to reject primers overlapping lowercase a base exactly at the 3' end.

This property relies on the assumption that masked features (e.g. repeats) can partly overlap primer, but they cannot overlap the 3'-end of the primer. In other words, lowercase bases at other positions in the primer are accepted, assuming that the masked features do not influence the primer performance if they do not overlap the 3'-end of primer.

PRIMER_MIN_GC (float, default 20.0%)

Minimum allowable percentage of Gs and Cs in any primer.

PRIMER_OPT_GC_PERCENT (float, default 50.0%)

Optimum GC percent. This parameter influences primer selection only if PRIMER_WT_GC_PERCENT_GT or PRIMER_WT_GC_PERCENT_LT are non-0.

PRIMER_MAX_GC (float, default 80.0%)

Maximum allowable percentage of Gs and Cs in any primer generated by Primer.

PRIMER_DNA_CONC (float, default 50.0 nM)

The nanomolar concentration of annealing oligos in the PCR. Primer3 uses this argument to calculate oligo melting temperatures. The default (50nM) works well with the standard protocol used at the Whitehead/MIT Center for Genome Research--0.5 microliters of 20 micromolar concentration for each primer oligo in a 20 microliter reaction with 10 nanograms template, 0.025 units/microliter Taq polymerase in 0.1 mM each

dNTP, 1.5mM MgCl2, 50mM KCl, 10mM Tris-HCL (pH 9.3) using 35 cycles with an annealing temperature of 56 degrees Celsius. This parameter corresponds to 'c' in equation (ii) of the paper [Rychlik W, Spencer WJ and Rhoads RE (1990) "Optimization of the annealing temperature for DNA amplification in vitro", Nucleic Acids Res 18:6409-12 http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pubmed&pubmedid=2243783], where a suitable value (for a lower initial concentration of template) is "empirically determined". The value of this parameter is less than the actual concentration of oligos in the reaction because it is the concentration of annealing oligos, which in turn depends on the amount of template (including PCR product) in a given cycle. This concentration increases a great deal during a PCR; fortunately PCR seems quite robust for a variety of oligo melting temperatures.

See ADVICE FOR PICKING PRIMERS.

PRIMER_NUM_NS_ACCEPTED (int, default 0)

Maximum number of unknown bases (N) allowable in any primer.

PRIMER_SELF_ANY (decimal,9999.99, default 8.00)

The maximum allowable local alignment score when testing a single primer for (local) self-complementarity and the maximum allowable local alignment score when testing for complementarity between left and right primers. Local self-complementarity is taken to predict the tendency of primers to anneal to each other without necessarily causing self-priming in the PCR. The scoring system gives 1.00 for complementary bases, -0.25 for a match of any base (or N) with an N, -1.00 for a mismatch, and -2.00 for a gap. Only single-base-pair gaps are allowed. For example, the alignment

```
5' ATCGNA 3'
   || | |
3' TA-CGT 5'
```

is allowed (and yields a score of 1.75), but the alignment

```
5' ATCCGNA 3'
   ||  | |
3' TA--CGT 5'
```

is not considered. Scores are non-negative, and a score of 0.00 indicates that there is no reasonable local alignment between two oligos.

PRIMER_SELF_END (decimal 9999.99, default 3.00)

The maximum allowable 3'-anchored global alignment score when testing a single primer for self-complementarity, and the maximum allowable 3'-anchored global alignment score when testing for complementarity between left and right primers. The 3'-anchored global alignment score is taken to predict the likelihood of PCR-priming primer-dimers, for example

```
5' ATGCCCTAGCTTCCGGATG 3'
             ||| |||||
          3' AAGTCCTACATTTAGCCTAGT 5'
```

or

```
5` AGGCTATGGGCCTCGCGA 3'
               ||||||
            3' AGCGCTCCGGGTATCGGA 5'
```

The scoring system is as for the Maximum Complementarity argument. In the examples above the scores are 7.00 and 6.00 respectively. Scores are non-negative, and a score of 0.00 indicates that there is no reasonable 3'-anchored global alignment between two oligos. In order to estimate 3'-anchored global alignments for candidate primers and primer pairs, Primer assumes that the sequence from which to choose primers is presented 5'->3'. It is nonsensical to provide a larger value for this parameter than for the Maximum (local) Complementarity parameter because the score of a local alignment will always be at least as great as the score of a global alignment.

PRIMER_DEFAULT_PRODUCT (size range list, default 100-300)

A deprecated synonym for PRIMER_PRODUCT_SIZE_RANGE, maintained for v2 compatibility.

PRIMER_FILE_FLAG (boolean, default 0)

If the associated value is non-0, then primer3 creates two output files for each input SEQUENCE. File <sequence_id>.for lists all acceptable left primers for <sequence_id>, and <sequence_id>.rev

lists all acceptable right primers for <sequence_id>, where
<sequence_id> is the value of the PRIMER_SEQUENCE_ID tag (which
must be supplied). In addition, if the input tag
PRIMER_PICK_INTERNAL_OLIGO is non-0, primer3 produces a file
<sequence_id>.int, which lists all acceptable internal oligos.

PRIMER_MAX_POLY_X (int, default 5)

The maximum allowable length of a mononucleotide repeat,
for example AAAAAA.

PRIMER_LIBERAL_BASE (boolean, default 0)

This parameter provides a quick-and-dirty way to get primer3 to
accept IUB / IUPAC codes for ambiguous bases (i.e. by changing
all unrecognized bases to N). If you wish to include an
ambiguous
base in an oligo, you must set PRIMER_NUM_NS_ACCEPTED to a
non-0 value.

Perhaps '-' and '* ' should be squeezed out rather than changed
to 'N', but currently they simply get converted to N's. The authors
invite user comments.

PRIMER_NUM_RETURN (int, default 5)

The maximum number of primer pairs to return. Primer pairs
returned are sorted by their "quality", in other words by the
value of the objective function (where a lower number indicates a
better primer pair). Caution: setting this parameter to a large
value will increase running time.

PRIMER_FIRST_BASE_INDEX (int, default 0)

This parameter is the index of the first base in the input
sequence. For input and output using 1-based indexing (such as
that used in GenBank and to which many users are accustomed) set
this parameter to 1. For input and output using 0-based indexing
set this parameter to 0. (This parameter also affects the
indexes in the contents of the files produced when the primer
file flag is set.)

PRIMER_MIN_QUALITY (int, default 0)

The minimum sequence quality (as specified by
PRIMER_SEQUENCE_QUALITY) allowed within a primer.

PRIMER_MIN_END_QUALITY (int, default 0)

The minimum sequence quality (as specified by
PRIMER_SEQUENCE_QUALITY) allowed within the 5' pentamer of a
primer.

PRIMER_QUALITY_RANGE_MIN (int, default 0)

The minimum legal sequence quality (used for error checking
of PRIMER_MIN_QUALITY and PRIMER_MIN_END_QUALITY).

PRIMER_QUALITY_RANGE_MAX (int, default 100)

The maximum legal sequence quality (used for error checking
of PRIMER_MIN_QUALITY and PRIMER_MIN_END_QUALITY).

PRIMER_INSIDE_PENALTY (float, default -1.0)

Non-default values are valid only for sequences with 0 or 1
target regions. If the primer is part of a pair that spans a
target and overlaps the target, then multiply this value times
the number of nucleotide positions by which the primer overlaps
the (unique) target to get the 'position penalty'. The effect of
this parameter is to allow primer3 to include overlap with the
target as a term in the objective function.

PRIMER_OUTSIDE_PENALTY (float, default 0.0)

Non-default values are valid only for sequences with 0 or 1
target regions. If the primer is part of a pair that spans a
target and does not overlap the target, then multiply this value
times the number of nucleotide positions from the 3' end to the
(unique) target to get the 'position penalty'. The effect of
this parameter is to allow primer3 to include nearness to the
target as a term in the objective function.

PRIMER_MAX_END_STABILITY (float 999.9999, default 100.0)

The maximum stability for the last five 3' bases of a left or
right primer. Bigger numbers mean more stable 3' ends. The
value is the maximum delta G (kcal/mol) for duplex disruption for
the five 3' bases as calculated using the nearest-neighbor
parameter values specified by PRIMER_TM_SANTALUCIA.

If PRIMER_TM_SANTALUCIA=1, then delta G for the most stable 5-mer

duplex (GCGCG) is 6.86 kcal/mol, and delta G for the most labile
5-mer (TATAT) is 0.86 kcal/mol.

If PRIMER_TM_SANTALUCIA=0, then delta G for the most stable 5-mer
duplex (GCGCG) is 13.4 kcal/mol, and delta G for the most labile
5-mer duplex (TATAC) is 4.6 kcal/mol.

PRIMER_PRODUCT_OPT_TM (float, default 0.0)

The optimum melting temperature for the PCR product. 0 indicates
that there is no optimum temperature.

PRIMER_PRODUCT_OPT_SIZE (int, default 0)

The optimum size for the PCR product. 0 indicates that there is
no optimum product size. This parameter influences primer
pair selection only
if PRIMER_PAIR_WT_PRODUCT_SIZE_GT or
PRIMER_PAIR_WT_PRODUCT_SIZE_LT is non-0.

PRIMER_TASK (string, default pick_pcr_primers)

Tell primer3 what task to perform. Legal values are pick_pcr_primers,
pick_pcr_primers_and_hyb_probe, pick_left_only, pick_right_only,
pick_hyb_probe_only. The tasks should be self explanatory, except
that we note that pick_pcr_primers_and_hyb_probe is
equivalent to the setting PRIMER_PICK_INTERNAL_OLIGO to a non-zero
value and setting PRIMER_TASK to pick_pcr_primers.

PRIMER_WT_TM_GT (float, default 1.0)

Penalty weight for primers with Tm over PRIMER_OPT_TM.

PRIMER_WT_TM_LT (float, default 1.0)

Penalty weight for primers with Tm under PRIMER_OPT_TM.

PRIMER_WT_SIZE_LT (float, default 1.0)

Penalty weight for primers shorter than PRIMER_OPT_SIZE.

PRIMER_WT_SIZE_GT (float, default 1.0)

Penalty weight for primers longer than PRIMER_OPT_SIZE.

PRIMER_WT_GC_PERCENT_LT (float, default 1.0)

Penalty weight for primers with GC percent greater than
PRIMER_OPT_GC_PERCENT.

PRIMER_WT_GC_PERCENT_GT (float, default 1.0)

Penalty weight for primers with GC percent greater than
PRIMER_OPT_GC_PERCENT.

PRIMER_WT_COMPL_ANY (float, default 0.0)
PRIMER_WT_COMPL_END (float, default 0.0)
PRIMER_WT_NUM_NS (float, default 0.0)
PRIMER_WT_REP_SIM (float, default 0.0)
PRIMER_WT_SEQ_QUAL (float, default 0.0)
PRIMER_WT_END_QUAL (float, default 0.0)
PRIMER_WT_POS_PENALTY (float, default 0.0)
PRIMER_WT_END_STABILITY (float, default 0.0)
PRIMER_WT_TEMPLATE_MISPRIMING (float, default 0.0)
PRIMER_PAIR_WT_PR_PENALTY (float, default 1.0)
PRIMER_PAIR_WT_IO_PENALTY (float, default 0.0)
PRIMER_PAIR_WT_DIFF_TM (float, default 0.0)
PRIMER_PAIR_WT_COMPL_ANY (float, default 0.0)
PRIMER_PAIR_WT_COMPL_END (float, default 0.0)
PRIMER_PAIR_WT_PRODUCT_TM_LT (float, default 0.0)
PRIMER_PAIR_WT_PRODUCT_TM_GT (float, default 0.0)
PRIMER_PAIR_WT_PRODUCT_SIZE_GT (float, default 0.0)
PRIMER_PAIR_WT_PRODUCT_SIZE_LT (float, default 0.0)
PRIMER_PAIR_WT_REP_SIM (float, default 0.0)
PRIMER_PAIR_WT_TEMPLATE_MISPRIMING (float, default 0.0)

Like the arguments governing PCR primer selection, the input tags
governing internal oligo selection are divided into sequence
input tags and global input tags, with for former being
automatically reset after each input record, and the latter
persisting until explicitly reset.

Because the laboratory detection step using internal oligos
is independent of the PCR amplification procedure,
internal oligo tags have defaults that are independent
of the parameters that govern the selection of PCR primers.
For example, the melting temperature of an oligo
used for hybridization might be considerably lower
than that used as a PCR primer.

Internal Oligo "Sequence" Input Tags

PRIMER_INTERNAL_OLIGO_EXCLUDED_REGION (interval list, default empty)

Middle oligos may not overlap any region specified by this tag.
The associated value must be a space-separated list of

<start>,<length>

pairs, where <start> is the index of the first base of
an excluded region, and <length> is its length. Often one would
make Target regions excluded regions for internal oligos.

PRIMER_INTERNAL_OLIGO_INPUT (nucleotide sequence, default empty)

The sequence of an internal oligo to check and around which to
design left and right primers. Must be a substring of SEQUENCE.

Internal Oligo "Global" Input Tags

These tags are analogous to the global input tags (those
governing primer oligos) discussed above. The exception is
PRIMER_INTERNAL_OLIGO_SELF_END which is meaningless when applied
to internal oligos used for hybridization-based detection, since
primer-dimer will not occur. We recommend that
PRIMER_INTERNAL_OLIGO_SELF_END be set at least as high as
PRIMER_INTERNAL_OLIGO_SELF_ANY.

PRIMER_INTERNAL_OLIGO_OPT_SIZE (int, default 20)
PRIMER_INTERNAL_OLIGO_MIN_SIZE (int, default 18)
PRIMER_INTERNAL_OLIGO_MAX_SIZE (int, default 27)
PRIMER_INTERNAL_OLIGO_OPT_TM (float, default 60.0 degrees C)
PRIMER_INTERNAL_OLIGO_OPT_GC_PERCENT (float, default 50.0%)
PRIMER_INTERNAL_OLIGO_MIN_TM (float, default 57.0 degrees C)
PRIMER_INTERNAL_OLIGO_MAX_TM (float, default 63.0 degrees C)
PRIMER_INTERNAL_OLIGO_MIN_GC (float, default 20.0%)
PRIMER_INTERNAL_OLIGO_MAX_GC (float, default 80.0%)
PRIMER_INTERNAL_OLIGO_SALT_CONC (float, default 50.0 mM)
PRIMER_INTERNAL_OLIGO_DIVALENT_CONC (float, default 0.0 mM)
PRIMER_INTERNAL_OLIGO_DNTP_CONC (float, default 0.0 mM)
PRIMER_INTERNAL_OLIGO_DNA_CONC (float, default 50.0 nM)
PRIMER_INTERNAL_OLIGO_SELF_ANY (decimal 9999.99, default 12.00)
PRIMER_INTERNAL_OLIGO_MAX_POLY_X (int, default 5)
PRIMER_INTERNAL_OLIGO_SELF_END (decimal 9999.99, default 12.00)

PRIMER_INTERNAL_OLIGO_MISHYB_LIBRARY (string, optional)

Similar to PRIMER_MISPRIMING_LIBRARY, except that the event we
seek to avoid is hybridization of the internal oligo to sequences
in this library rather than priming from them.

PRIMER_INTERNAL_OLIGO_MAX_MISHYB (decimal,9999.99, default 12.00)

Similar to PRIMER_MAX_MISPRIMING except that this parameter applies
to the similarity of candidate internal oligos to the library
specified in PRIMER_INTERNAL_OLIGO_MISHYB_LIBRARY.

PRIMER_INTERNAL_OLIGO_MAX_TEMPLATE_MISHYB (decimal,9999.99, default 12.00)

Not implemented.

PRIMER_INTERNAL_OLIGO_MIN_QUALITY (int, default 0)

(Note that there is no PRIMER_INTERNAL_OLIGO_MIN_END_QUALITY.)

PRIMER_IO_WT_TM_GT (float, default 1.0)
PRIMER_IO_WT_TM_LT (float, default 1.0)
PRIMER_IO_WT_GC_PERCENT_GT (float, default 1.0)
PRIMER_IO_WT_GC_PERCENT_LT (float, default 1.0)
PRIMER_IO_WT_SIZE_LT (float, default 1.0)
PRIMER_IO_WT_SIZE_GT (float, default 1.0)
PRIMER_IO_WT_COMPL_ANY (float, default 0.0)
PRIMER_IO_WT_COMPL_END (float, default 0.0)
PRIMER_IO_WT_NUM_NS (float, default 0.0)
PRIMER_IO_WT_REP_SIM (float, default 0.0)
PRIMER_IO_WT_SEQ_QUAL (float, default 0.0)
PRIMER_IO_WT_END_QUAL (float, default 0.0)

AN EXAMPLE

One might be interested in performing PCR on an STS with a CA
repeat in the middle of it. Primers need to be chosen based on
the criteria of the experiment.

We need to come up with a boulder-io record to send to primer3 via
stdin. There are lots of ways to accomplish this. We could save
the record into a text file called 'input', and then type the
UNIX command 'primer3 < input'.

Let's look at the input record itself:

```
PRIMER_SEQUENCE_ID=example
SEQUENCE=GTAGTCAGTAGACNATGACNACTGACGATGCAGACNACACACACACACACAGCACACAGGTATTAG
TGGGCCATTCGATCCCGACCCAAATCGATAGCTACGATGACG
TARGET=37,21
PRIMER_OPT_SIZE=18
PRIMER_MIN_SIZE=15
PRIMER_MAX_SIZE=21
PRIMER_NUM_NS_ACCEPTED=1
PRIMER_PRODUCT_SIZE_RANGE=75-100
PRIMER_FILE_FLAG=1
PRIMER_PICK_INTERNAL_OLIGO=1
PRIMER_INTERNAL_OLIGO_EXCLUDED_REGION=37,21
PRIMER_EXPLAIN_FLAG=1
=
```

A breakdown of the reasoning behind each of the TAG=VALUE pairs
is below:

```
PRIMER_SEQUENCE_ID=example
```

The main intent of this tag is to provide an identifier for the
sequence that is meaningful to the user, for example when primer3
processes multiple records, and by default this tag is optional.
However, this tag is _required_ when PRIMER_FILE_FLAG is non-0
Because it provides the names of the files that contain lists
of oligos that primer3 considered.

```
SEQUENCE=GTAGTCAGTAGACNATGACNACTGACGATGCAGACNACACACACACACACAGCACACAGGTATTAG
TGGGCCATTCGATCCCGACCCAAATCGATAGCTACGATGACG
```

The SEQUENCE tag is of ultimate importance. Without it, primer3
has no idea what to do. This sequence is 92 bases long. Note that
there is no newline until the sequence terminates completely.

```
TARGET=37,21
```

There is a simple sequence repeat in our sequence, which starts
at base 37, and has a length of 21 bases. We want primer3 to
choose primers which flank the repeat site, so we let primer3 know
that we consider this site to be important.

```
PRIMER_OPT_SIZE=18
```

Since our sequence length is rather small (only 92 bases
long), we lower the PRIMER_OPT_SIZE from 20 to 18. It's

more likely that primer3 will succeed if it shoots for smaller primers with such a small sequence.

PRIMER_MIN_SIZE=15
PRIMER_MAX_SIZE=21

With the lowering of optimal primer size, it's good to lower the minimum and maximum sizes as well.

PRIMER_NUM_NS_ACCEPTED=1

Again, since we've got such a small sequence with a non-negligible amount of unknown bases (N's) in it, let's make primer3's job easier by allowing it to pick primers that have at most 1 unknown base.

PRIMER_PRODUCT_SIZE_RANGE=75-100

We reduce the product size range from the default of 100-300 because our source sequence is only 108 base pairs long. If we insisted on a product size of 100 base pairs primer3 would have few possibilities to choose from.

PRIMER_FILE_FLAG=1

Since we've got such a small sequence, Primer might fail to pick primers. We want to get the list of primers it considered, then, so that we might manually pick primers ourselves if Primer fails to do so. Setting this flag to 1 will force Primer to output the primers it considered to a forward_primer and a reverse_primer output file.

PRIMER_PICK_INTERNAL_OLIGO=1

We want to see if Primer v2.3 can pick an internal oligo for the sequence, so we set this flag to 1 (true).

PRIMER_INTERNAL_OLIGO_EXCLUDED_REGION=37,21

Normally CA-repeats make poor hybridization probes (because they not specific enough). Therefor we exclude the CA repeat (which is the TARGET) from consideration for the middle oligo.

PRIMER_EXPLAIN_FLAG=1

We want to see statistics about the oligos and oligo triples

(left primer, internal oligo, right primer) that primer3 examined.

=

The '=' character terminates the record.

Tere are some boulderio tags that we never even specified. (INCLUDED_REGION, EXCLUDED_REGION, et al.), which is perfectly legal. For the tags with default values, those defaults will be used in the analysis. For the tags with NO default values (like TARGET, for instance), the functionality requested by the those tags will simply be absent. It's not the case that we need to surround a simple sequence repeat every time we want to pick primers!

OUTPUT TAGS

For each boulderio record passed into primer3 via stdin, exactly one boulderio record comes out of primer3 on stdout. These output records contain everything that the input record contains, plus a subset of the following tag/value pairs. Unless noted by (*), each tag appears for each primer pair returned. The first version is PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO,PAIR}_<tag_name>. Tags of additional primers chosen are of the form PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO,PAIR}_<j>_<tag_name>. where <j> is an integer from 1 to n, where n is at most the value of PRIMER_NUM_RETURN.

In the descriptions below, 'i,n' represents a start/length pair, 's' represents a string, x represents an arbitrary integer, and f represents a float.

PRIMER_ERROR=s (*)

s describes user-correctible errors detected in the input (separated by semicolons). This tag is absent if there are no errors.

PRIMER_LEFT=i,n

The selected left primer (the primer to the left in the input sequence). i is the 0-based index of the start base of the primer, and n is t its length.

PRIMER_RIGHT=i,n

The selected right primer (the primer to the right in the input sequence). i is the 0-based index of the last base of the primer, and n is its length.

PRIMER_INTERNAL_OLIGO=i,n

The selected internal oligo. Primer3 outputs this tag if PRIMER_PICK_INTERNAL_OLIGO was non-0. If primer3 fails to pick a middle oligo upon request, this tag will not be output. i is the 0-based index of start base of the internal oligo, and n is its length.

PRIMER_PRODUCT_SIZE=x

x is the product size of the PCR product.

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_EXPLAIN=s (*)

s is a (more or less) self-documenting string containing statistics on the possiblities that primer3 considered in selecting a single oligo. For example

PRIMER_LEFT_EXPLAIN=considered 62, too many Ns 53, ok 9
PRIMER_RIGHT_EXPLAIN=considered 62, too many Ns 53, ok 9
PRIMER_INTERNAL_OLIGO_EXPLAIN=considered 87, too many Ns 39, overlap excluded region 40, ok 8

All the categories are exclusive, except the 'considered' category.

PRIMER_PAIR_EXPLAIN=s (*)

s is a self-documenting string containing statistics on picking a primer pair (plus internal oligo if requested). For exaple

PRIMER_PAIR_EXPLAIN=considered 81, unacceptable product size 49, no internal oligo 32, ok 0

All the categories are exclusive, except the 'considered' category.

In some cases primer3 will examine a primer pair before it discovers that one of the primers in the pair violates specified constraints. In this case PRIMER_PAIR_EXPLAIN might have a non-0 number 'considered', even though one or more of PRIMER_LEFT_EXPLAIN, PRIMER_RIGHT_EXPLAIN, or

PRIMER_INTERNAL_OLIGO_EXPLAIN has 'ok 0'.

PRIMER_PAIR_PENALTY=f

The value of the objective function for this pair (lower is better).

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_PENALTY=f

The contribution of this individual primer or oligo to the
objective function.

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_SEQUENCE=s

The actual sequence of the oligo. The sequence of left primer and
internal oligo is presented 5' -> 3' on the same strand as the
input SEQUENCE (which must be presented 5' -> 3'). The sequence
of the right primer is presented 5' -> 3' on the opposite strand
from the input SEQUENCE.

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_TM=f

The melting TM for the selected oligo.

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_GC_PERCENT=f

The percent GC for the selected oligo (denominator is the number
of non-ambiguous bases).

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_SELF_ANY=f
PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_SELF_END=f

The self-complementarity measures for the selected oligo.

PRIMER_PAIR_COMPL_ANY=f
PRIMER_PAIR_COMPL_END=f

The inter-pair complementarity measures for the selected left and
right primer

PRIMER_WARNING=s (*)

s lists warnings generated by primer (separated by semicolons);
this tag is absent if there are no warnings

PRIMER_{LEFT,RIGHT,PAIR}_MISPRIMING_SCORE=f, s

f is the maximum mispriming score for the right primer
against any sequence in the given PRIMER_MISPRIMING_LIBRARY;
s is the id of corresponding library sequence.
PRIMER_PAIR_MISPRIMING_SCORE is the maximum sum of
mispriming scores in any single library sequence (perhaps a
more reasonable estimator of the likelihood of mispriming).

PRIMER_{LEFT,RIGHT,PAIR}_TEMPLATE_MISPRIMING=f

Analogous to PRIMER_{LEFT,RIGHT,PAIR}_MISPRIMING_SCORE, except that
these output tags apply to mispriming within the template sequence.
This often arises, for example, in genes with repeated exons. For
backward compatibility, these tags only appear if the corresponding
input tags have defined values.

PRIMER_PRODUCT_TM=f

f is the melting temperature of the product. Calculated using
equation (iii) from the paper [Rychlik W, Spencer WJ and Rhoads
RE (1990) "Optimization of the annealing temperature for DNA
amplification in vitro", Nucleic Acids Res 18:6409-12
http://www.pubmedcentral.nih.gov/articlerender.fcgi?
tool=pubmed&pubmedid=2243783].
Printed only if a non-default value of PRIMER_MAX_PRODUCT_TM or
PRIMER_MIN_PRODUCT_TM is specified.

PRIMER_PRODUCT_TM_OLIGO_TM_DIFF=f

f is the difference between the melting temperature of the
product and the melting temperature of the less stable primer.
Printed only if PRIMER_MAX_PRODUCT_TM or PRIMER_MIN_PRODUCT_TM is
specified.

PRIMER_PAIR_T_OPT_A=f

f is T sub a super OPT from equation (i) in [Rychlik W, Spencer
WJ and Rhoads RE (1990) "Optimization of the annealing
temperature for DNA amplification in vitro", Nucleic Acids Res
18:6409-12
http://www.pubmedcentral.nih.gov/articlerender.fcgi?
tool=pubmed&pubmedid=2243783].
Printed only if PRIMER_MAX_PRODUCT_TM or PRIMER_MIN_PRODUCT_TM is
specified.

PRIMER_INTERNAL_OLIGO_MISHYB_SCORE=f, s

f is the maximum mishybridization score for the right primer against any sequence in the given PRIMER_INTERNAL_OLIGO_MISHYB_LIBRARY; s is the id of corresponding library sequence.

PRIMER_{LEFT,RIGHT,INTERNAL_OLIGO}_MIN_SEQ_QUALITY=i

i is the minimum _sequence_ quality within the primer or oligo (not to be confused with the PRIMER_PAIR_QUALITY output tag, which is really the value of the objective function.)

PRIMER_{LEFT,RIGHT}_END_STABILITY=f

f is the delta G of disruption of the five 3' bases of the primer.

PRIMER_STOP_CODON_POSITION=i

i is the position of the first base of the stop codon, if primer3 found one, or -1 if primer3 did not. Printed only if the input tag PRIMER_START_CODON_POSITION with a non-default value is supplied.

EXAMPLE OUTPUT

You should run it youself. Use the file 'example' in this directory as input.

ADVICE FOR PICKING PRIMERS

We suggest consulting: Wojciech Rychlik (1993) "Selection of Primers for Polymerase Chain Reaction" in BA White, Ed., "Methods in Molecular Biology, Vol. 15: PCR Protocols: Current Methods and Applications", pp 31-40, Humana Press, Totowa NJ.

CAUTIONS

Some of the most important issues in primer picking can be addressed only before using primer3. These are sequence quality (including making sure the sequence is not vector and not chimeric) and avoiding repetitive elements.

Techniques for avoiding problems include a thorough understanding of possible vector contaminants and cloning artifacts coupled with database searches using blast, fasta, or other similarity searching program to screen for vector contaminants and possible repeats. Repbase (J. Jurka, A.F.A. Smit, C. Pethiyagoda, and others, 1995-1996, ftp://ncbi.nlm.nih.gov/repository/repbase) is an excellent source of repeat sequences and pointers to the literature. (The Repbase files need to be converted to Fasta format before they can be used by primer3.) Primer3 now allows you to screen candidate oligos against a Mispriming Library (or a Mishyb Library in the case of internal oligos).

Sequence quality can be controlled by manual trace viewing and quality clipping or automatic quality clipping programs. Low-quality bases should be changed to N's or can be made part of Excluded Regions. The beginning of a sequencing read is often problematic because of primer peaks, and the end of the read often contains many low-quality or even meaningless called bases. Therefore when picking primers from single-pass sequence it is often best to use the INCLUDED_REGION parameter to ensure that primer3 chooses primers in the high quality region of the read.

In addition, primer3 takes as input a Sequence Quality list for use with those base calling programs

(e.g. Phred, Bass/Grace, Trout) that output this information.

WHAT TO DO IF PRIMER3 CANNOT FIND ANY PRIMERS?

Try relaxing various parameters, including the self-complementarity parameters and max and min oligo melting temperatures. For example, for very A-T-rich regions you might have to increase maximum primer size or decrease minimum melting temperature. It is usually unwise to reduce the minimum primer size if your template is complex (e.g. a mammalian genome), since small primers are more likely to be non-specific. Make sure that there are adequate stretches of non-Ns in the regions in which you wish to pick primers. If necessary you can also allow an N in your primer and use an oligo mixture containing all four bases at that position.

Try setting the PRIMER_EXPLAIN_FLAG input tag.

DIFFERENCES FROM EARLIER VERSIONS

See the file release_notes.txt in this directory.

EXIT STATUS CODES

```
 0 on normal operation
-1 under the following conditions:
   illegal command-line arguments.
   unable to fflush stdout.
   unable to open (for writing and creating) a .for, .rev
     or .int file (probably due to a protection problem).
-2 on out-of-memory
-3 empty input
-4 error in a "Global" input tag (message in PRIMER_ERROR).
```

Primer3 calls abort() and dumps core (if possible) if a programming error is detected by an assertion violation.

SIGINT and SIGTERM are handled essentially as empty input, except the signal received is returned as the exit status and printed to stderr.

In all of the error cases above Primer3 prints a message to stderr.

THE PRIMER3 WWW INTERFACE

This distribution does not contain the Primer3 WWW interface. Web interface code is likely available at (or linked to from) http://sourceforge.net/projects/primer3/.

ACKNOWLEDGMENTS

Initial development of Primer3 was funded by Howard Hughes Medical Institute and by the National Institutes of Health, National Human Genome Research Institute under grants R01-HG00257 (to David C. Page) and P50-HG00098 (to Eric S. Lander).

Primer3 was originally written by Helen J. Skaletsky (Howard Hughes Medical Institute, Whitehead Institute) and Steve Rozen (Whitehead

Institute/MIT Center for Genome Research), based on the design of earlier versions: Primer 0.5 (Steve Lincoln, Mark Daly, and Eric S. Lander) and Primer v2 (Richard Resnick). This initial version of this documentation was written by Richard Resnick and Steve Rozen, and the original web interface was designed by Richard Resnick. Lincoln Stein championed the use of the Boulder-IO format and the idea of making primer3 a software component. In addition, among others, Ernst Molitor, Carl Foeller, and James Bonfield contributed to the early design of primer3. We also thank Centerline Software, Inc., for uses of its TestCenter memory-error, -leak, and test-coverage checker, which helped us discover and correct a number of otherwise latent errors in Primer3.

Primer3 is now operating as open software development project hosted on SourceForge, and we are working out how to acknowledge all who have contributed to its enahancement. Current active developers can be found at http://sourceforge.net/projects/primer3/.