

RSATWSPortType	
<b>Service Documentation</b>	Web services for the Regulatory Sequence Analysis Tools (RSAT). Tools developed by Jacques van Helden (jvanheld@scmbb.ulb.ac.be), SOAP/WSDL interface developed by Olivier Sand (oly@scmbb.ulb.ac.be).
<b>Server Address</b>	http://rsat.scmbb.ulb.ac.be/rsat/web_services/RSATWS.cgi
<b>Method</b>	<b>retrieve_seq</b>
<b>Description</b>	Returns upstream, downstream or coding DNA sequences for list of query genes.
<b>Parameters</b>	<b>Input Parameters</b> <p><b>output</b> Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String</p> <p><b>organism (required)</b> Organism. Words need to be underscore separated (example: Escherichia_coli_K12). Type = String</p> <p><b>query</b> A list of query genes. Type = String</p> <p><b>all</b> Return sequences for all the genes of the organism if value = 1. Incompatible with query. Type = Integer</p> <p><b>noorf</b> Prevent overlap with upstream open reading frames (ORF) if value = 1. Type = Integer</p> <p><b>from</b> Inferior limit of the region to retrieve. Default is organism dependant (example: Saccharomyces cerevisiae = -800). Type = Integer</p> <p><b>to</b> Superior limit of the region to retrieve. Default is '-1'. Type = Integer</p> <p><b>featype</b> Type of genome features to load. Supported: CDS, mRNA, tRNA, rRNA. Type = String</p> <p><b>type</b> Sequence type. Supported: upstream, downstream, ORF (unspliced open reading frame). Type = String</p> <p><b>format</b> Sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, FastA Type = String</p> <p><b>lw</b> Line width (0 for whole sequence on one line). Type = Integer</p> <p><b>label</b> Field(s) to be used in the sequence label. Multiple fields can be specified, separated by commas. Supported: id, name, organism_name, sequence_type, current_from, current_to, ctg, orf_strand, reg_left, reg_right. Default: name. Type = String</p> <p><b>label_sep</b> Separator between the label fields. Default:   (pipe character). Type = String</p> <p><b>nocom</b> No comments if value = 1. Only the identifier and the sequence are returned. By default, the comment indicates the ORF and upstream sequence coordinates. Type = Integer</p> <p><b>repeat</b> Use the repeat masked version of the genome if value = 1. Attention: repeated regions are annotated for some genomes only. Type = Integer</p> <p><b>imp_pos</b> Admit imprecise positions if value = 1. Type = Integer</p> <b>Output Parameters</b> <p><b>server</b> Location of the result file on the server. This can be used as input for a further request. Type = String</p> <p><b>command</b> The stand alone command executed on the server. Type = String</p> <p><b>client</b> The results. Type = String</p>
<b>Method</b>	<b>purge_seq</b>
<b>Description</b>	Mask repeated fragments of an input sequence.
<b>Parameters</b>	<b>Input Parameters</b> <p><b>output</b> Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String</p> <p><b>sequence</b> Sequence to purge. You need to supply either this parameter or the next one (tmp_infile). Type = String</p> <p><b>tmp_infile</b> Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence). Type = String</p>

<b>format</b>	Sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, FastA. Type = String
<b>match_length</b>	Minimal match length. Default is 40. Type = Integer
<b>mismatch</b>	Number of mismatches allowed. Default is 3. Type = Integer
<b>str</b>	Discard duplications on the direct strand only (1) or on the reverse complement as well (2). Default is 2. Type = Integer
<b>delete</b>	Delete repeats instead of masking them if value = 1. Type = Integer
<b>mask_short</b>	Mask (replace by N characters) sequences shorter than the specified length. Type = Integer

**Output Parameters**

<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
<b>command</b>	The stand alone command executed on the server. Type = String
<b>client</b>	The results. Type = String

**Method****oligo\_analysis****Description**

Analysis of the statistical significance of all the oligomers of a given size in a sequence. Commonly used to detect over-represented oligonucleotides in a set of promoter sequences.

**Parameters****Input Parameters**

<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
<b>sequence</b>	Input sequence. You need to supply either this parameter or the next one (tmp_infile). Type = String
<b>tmp_infile</b>	Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence). Type = String
<b>format</b>	Input sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, fasta. Default is fasta. Type = String
<b>length</b> (required)	Oligomer length. Type = Integer
<b>organism</b> (required)	Organism. Words need to be underscore separated (example: Escherichia_coli_K12). Type = String
<b>background</b>	Background model: Type of sequences used as background model for estimating expected oligonucleotide frequencies. Supported: upstream, upstreamL, upstream-noorf, intergenic, input. Type = String
<b>stats</b>	List of statistics to return. Supported: occ, mseq, freq, proba, ratio, zscore, like, pos, rank. Type = String
<b>noov</b>	No overlapping of oligos allowed if value = 1. Disable the detection of overlapping matches for self-overlapping patterns (ex TATATA, GATAGA). Type = Integer
<b>str</b>	Oligonucleotide occurrences found on both stands are summed (2) or not (1). Default is 2. Type = Integer
<b>sort</b>	Sort oligomers according to overrepresentation if value = 1. Type = Integer
<b>lth</b>	Lower threshold on some parameter. Format='param value'. Type = String

**Output Parameters**

<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
<b>command</b>	The stand alone command executed on the server. Type = String
<b>client</b>	The results. Type = String

**Method****dyad\_analysis****Description**

Analysis of the statistical significance of all the spaced dyads of a given size in a sequence. Commonly used to detect over-represented spaced dyads in a set of promoter sequences.

Parameters	<b>Input Parameters</b>	
	<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
	<b>sequence</b>	Input sequence. You need to supply either this parameter or the next one (tmp_infile). Type = String
	<b>tmp_infile</b>	Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence). Type = String
	<b>format</b>	Input sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, fasta. Default is fasta. Type = String
	<b>length (required)</b>	Dyad length. Type = Integer
	<b>spacing (required)</b>	Spacing between elements of the dyads. Type = String
	<b>organism (required)</b>	Organism. Words need to be underscore separated (example: Escherichia_coli_K12). Type = String
	<b>background</b>	Background model: Type of sequences used as background model for estimating expected oligonucleotide frequencies. Supported: upstream, upstreamL, upstream-noorf, intergenic, input. Type = String
	<b>stats</b>	List of statistics to return. Supported: occ, mseq, freq, proba, ratio, zscore, like, pos, rank. Type = String
	<b>type</b>	dyad_type (dr   ir   rep   any) In order to fasten execution, the program can be asked to restrict its analysis to symmetric dyads. Four types are accepted: dr - direct repeats: the second element is the same as the first one; ir - inverted repeats: the second element is the reverse complement of the first one; rep - repeats: direct and inverted repeats are evaluated. any - (default) When selecting the option any, the analysis is performed on all non-symmetric dyads as well. Type = String
	<b>noov</b>	No overlapping of dyads allowed if value = 1. Disable the detection of overlapping matches for self-overlapping patterns (ex TATATA, GATAGA). Type = Integer
	<b>str</b>	Dyad occurrences found on both stands are summed (2) or not (1). Default is 2. Type = Integer
	<b>sort</b>	Sort dyads according to overrepresentation if value = 1. Type = Integer
	<b>under</b>	Detect under-represented instead of over-represented dyads (left tail significance test) if value = 1. Type = Integer
	<b>two_tails</b>	Detect under-represented and over-represented dyads (two-fail significance test) if value = 1. Type = Integer
	<b>zeroocc</b>	Report also dyads with zero occurrences (provided they fit the other thresholds) if value = 1. By default, the program reports only patterns present in the sequence. If the left tail or two-tail test is applied, patterns with zero occurrences are automatically taken into account. In some other cases, one would also like to detect patterns absent from the sequence. This is the function of the option -zeroocc. Type = Integer
	<b>lth</b>	Lower threshold on some parameter. Format='param value'. Type = String
	<b>uth</b>	Upper threshold on some parameter. Format='param value'. Type = String
	<b>Output Parameters</b>	
	<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
	<b>command</b>	The stand alone command executed on the server. Type = String
	<b>client</b>	The results. Type = String
<b>Method</b>	<b>dna_pattern</b>	
<b>Description</b>	Searches all occurrences of a pattern within DNA sequences.	
<b>Parameters</b>	<b>Input Parameters</b>	
	<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
	<b>sequence</b>	Input sequence. You need to supply either this parameter or the next one (tmp_infile). Type = String
	<b>tmp_infile</b>	Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence). Type = String

<b>format</b>	Input sequence format. Supported: IG (Intelligenetics), WC (wconsensus), raw, fasta. Default is fasta. Type = String
<b>subst</b>	Number of substitutions allowed. Type = Integer
<b>pattern</b> (required)	Pattern to match. Type = String
<b>id</b>	Pattern identifier. Type = String
<b>origin</b>	Origin for the calculation of positions (0 for end of sequence). Type = String
<b>noov</b>	No overlapping of oligos allowed if value = 1. Disable the detection of overlapping matches for self-overlapping patterns (ex TATATA, GATAGA). Type = Integer
<b>str</b>	Oligonucleotide occurrences found on both stands are summed (2) or not (1). Default is 2. Type = Integer
<b>sort</b>	Sort oligomers according to overrepresentation if value = 1. Type = Integer
<b>th</b>	Threshold on match count. Type = Integer
<b>Output Parameters</b>	
<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
<b>command</b>	The stand alone command executed on the server. Type = String
<b>client</b>	The results. Type = String

**Method****gene\_info****Description**

Get information about genes.

**Parameters****Input Parameters**

<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
<b>organism</b> (required)	Organism. Words need to be underscore separated (example: Escherichia_coli_K12). Type = String
<b>query</b> (required)	List of gene(s) for which you want info on or list of keywords to search for (can be regular expressions). Type = String
<b>full</b>	Full match only (no substring matching) if value = 1. Type = Integer
<b>noquery</b>	Do not print the query at the begining of each line if value = 1. Type = Integer
<b>descr</b>	Match query against the description, too, not just against gene ID and name if value = 1. Type = Integer
<b>featype</b>	Feature type (CDS, mRNA, tRNA, rRNA, scRNA). Type = String

**Output Parameters**

<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
<b>command</b>	The stand alone command executed on the server. Type = String
<b>client</b>	The results. Type = String

**Method****supported\_organisms****Description**

Get a list of supported organisms.

**Parameters****Input Parameters**

<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
<b>format</b>	Output format (supported: html_list,html_table,array,text,keys,names,sizes,full,tree,html_tree). Type = String
<b>taxon</b>	Root taxon. Type = String

**Output Parameters**

<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
<b>command</b>	The stand alone command executed on the server. Type = String
<b>client</b>	The results. Type = String

**Method** **convert\_seq****Description** Converts a sequence between two formats (e.g. fasta -> raw).**Parameters****Input Parameters**

<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
<b>sequence</b>	Sequence to convert. Type = String
<b>tmp_infile</b>	Name of the file with input sequence on the server. You need to supply either this parameter or the previous one (sequence). Type = String
<b>from</b> (required)	Format of input sequence (embl, fasta, filetype, gcg, genbank, ig, maf, multi, ncbi, raw, tab, wc, wconsensus). Type = String
<b>to</b> (required)	Format of output sequence (fasta, filetype, ft, ig, multi, raw, tab, wc, wconsensus). Type = String

**Output Parameters**

<b>server</b>	Location of the result file on the server. This can be used as input for a further request. Type = String
<b>command</b>	The stand alone command executed on the server. Type = String
<b>client</b>	The results. Type = String

**Method** **compare\_classes****Description** Compare two class files(the query file and the reference file).Each class of the query file is compared to each class of the reference file.The number of common elements is reported, as well as the probability to observe at least this number of common elements by chance alone.**Parameters** **Input Parameters**

<b>output</b>	Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String
<b>ref_classes</b>	A tab-delimited text file containing the description of reference classes Type = String
<b>query_classes</b>	A tab-delimited text file containing the description of query classes Type = String
<b>return_fields</b>	List of fields to return. Supported field: dotprod,entropy,freq,jac_sim,members,occ,proba,rank Type = Integer
<b>score_column</b>	Specify a column of the input file containing a score associated to each member Type = Integer
<b>input_classes</b>	This file will be used as both reference and query.This is equivalent to -q input_file -r input_file. Type = String
<b>upper_threshold_field</b>	Supported fields : E(QR),E_val,F(!Q!R),F(Q!R),F(Q),F(QR),F(R!Q),F(R),H(Q),H(Q,R),H(Q!R),H(R),H(R!Q),I(Q,R),IC,P(QR),P(Q!R),P(R!Q),P_val,Q,QR,QvR,R,U(Q!R),U(R!Q),dH(Q,R),dotprod,jac_sim,rank,sig Type = String
<b>upper_threshold_value</b>	upper threshold value for a supported field Type = Float
<b>lower_threshold_field</b>	Supported fields: same fields as upper_threshold_field. Type = String
<b>lower_threshold_value</b>	lower threshold value for a given fields. Type = Float
<b>sort</b>	sort on the basis of the specified key. Type = String
<b>distinct</b>	Prevent to compare each class with itself (when the reference and query files contain the same classes) Type = Integer
<b>triangle</b>	(only valid if query file and reference file are the same) Do not perform the reciprocal comparisons. Type = Integer

**matrix** Return a pairwise matrix, where each row corresponds to a reference class, each column to a query class, and each cell contains a comparison between the two classes. The next argument indicates which statistics has to be return in the matrix (default=sig)Supported: E(QR),E\_val,F(!Q!R),F(Q!R),F(Q),F(QR),F(R!Q),F(R),H(Q),H(Q,R),H(Q!R),H(R),H(R!Q),I(Q,R),IC,P(QR),P(Q!R),P(R!Q),P\_val,Q,QR,QvR,R,U(Q!R),U(R!Q),dH(Q,R),dotprod,jac\_sim,rank,sig  
Type = String

#### Output Parameters

**server** Location of the result file on the server. This can be used as input for a further request.  
Type = String

**command** The stand alone command executed on the server.  
Type = String

**client** The results.  
Type = String

<b>Method</b>	<b>matrix_scan</b>
<b>Description</b>	Scan sequences with one or several position-specific scoring matrices (PSSM) to identify instances of the corresponding motifs(putative sites). This program supports a variety of background models (Bernoulli, Markov chains of any order).
<b>Parameters</b>	<p><b>Input Parameters</b></p> <p><b>output</b> Return type. Accepted values: 'server' (result is stored on a file on the server), 'client' (result is directly transferred to the client), or 'both'. Default is 'both'. Type = String</p> <p><b>sequence_file</b> Sequence file - all the formats supported in RSAT can be used as input (default: fasta) Type = String</p> <p><b>matrix_file</b> The matrix file is specified with the option "matrix_format" (see below) Default format: tab . Type = String</p> <p><b>matrix_format</b> Supported field: tab, cb, consensus, gibbs, meme, assembly. Type = String</p> <p><b>matrix_list</b> Indicate a file containing a list of matrices to be used for scanning the region. This facilitates the scanning of a sequence with a library of matrices (e.g. all the matrices from RegulonDB or TRANSFAC) Format: the matrix list file is a text file. The first word of each row is suppose to indicate a file name. Any further information on the same row is ignored. Type = String</p> <p><b>top_matrices</b> Only scan with the top # matrices per matrix file. This option is valid for some file formats containing multiple matrices where top matrices are generally more informative. Type = Integer</p> <p><b>background</b> Background model file is a tab-delimited file containing the specification of oligonucleotide frequencies. Type = String</p> <p><b>background_input</b> Calculate background model from the input sequence set. This option requires to specify the order of the background model with the option markov . Type = Integer</p> <p><b>background_window</b> Size of the sliding window for the background model calculation.This option requires to specify the order of the background model with the option markov (suitable for short order model only markov 0 or 1) Type = Integer</p> <p><b>markov</b> Order of the markov chain for the background model. This option is incompatible with the option background . Type = Integer</p> <p><b>background_pseudo</b> Pseudo frequency for the background models. Value must be a real between 0 and 1. If this option is not specified, the pseudo-frequency value depends on the background calculation. For -bginput and -window, the pseudo frequency is automatically calculated with the length (L) of the sequence following this formula : square-root of L divided by L+square-root of L. For -bgfile, default value is 0.01. If the training sequence length (L) is known, the value can be set by -bg_pseudo option to square-root of L divided by L+square-root of L. Type = Float</p> <p><b>return_fields</b> List of fields to return. Supported fields: sites, rank, limits, normw, bg_model, matrix, freq_matrix, weight_matrix, distrib . Type = String</p> <p><b>upper_threshold_field</b> Type = String</p> <p><b>upper_threshold_value</b> Type = Float</p> <p><b>lower_threshold_field</b> Type = String</p> <p><b>lower_threshold_value</b> Type = Float</p> <p><b>both_strand</b> Type = Integer</p>

**single\_strand**

Type = Integer

**Output Parameters**

**server** Location of the result file on the server. This can be used as input for a further request.  
Type = String

**command** The stand alone command executed on the server.  
Type = String

**client** The results.  
Type = String