LEIDEN UNIVERSITY MEDICAL CENTER

Mutalyzer webservices

Jeroen F. J. Laros

Leiden Genome Technology Center

Department of Human Genetics

Center for Human and Clinical Genetics

Introduction

Mutalyzer: a curational tool for Locus Specific Mutation Databases (LsdBs)

Variant nomenclature checker applying *Human Genome Variation Society* (HGVS) guidelines.

Introduction

Mutalyzer: a curational tool for Locus Specific Mutation Databases (LsdBs)

Variant nomenclature checker applying *Human Genome Variation Society* (HGVS) guidelines.

- Is the syntax of the variant description valid?
- Does the reference sequence exist?
- Is the variant possible on this reference sequence?
- Is this variant description the recommended one?

Introduction

Mutalyzer: a curational tool for Locus Specific Mutation Databases (LsdBs)

Variant nomenclature checker applying *Human Genome Variation Society* (HGVS) guidelines.

- Is the syntax of the variant description valid?
- Does the reference sequence exist?
- Is the variant possible on this reference sequence?
- Is this variant description the recommended one?

Basic effect prediction.

- Is the description of the transcript product as expected?
- Is the predicted protein as expected?

Introduction

HGVS nomenclature

Genomic orientated positions:

AL449423.14:g.[65449_65463del;65564T>C]

Introduction

HGVS nomenclature

Genomic orientated positions:

AL449423.14:g.[65449_65463del;65564T>C]

Coding sequence orientated positions:

AL449423.14(CDKN2A_v001):c.[5A>G;106_120del]

Introduction

HGVS nomenclature

Genomic orientated positions:

AL449423.14:g.[65449_65463del;65564T>C]

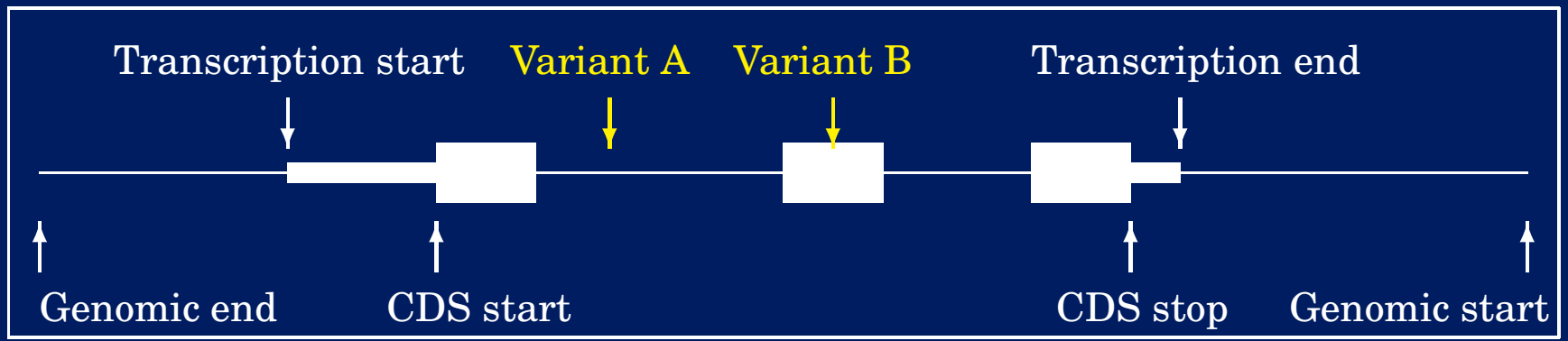
Coding sequence orientated positions:

AL449423.14(CDKN2A_v001):c.[5A>G;106_120del]

- **AL449423.14** – reference sequence.
- **CDKN2A_v001** – transcript variant **1** of gene CDKN2A.
- **c.[5A>G;106_120del]**
  - A *substitution* at position **5** counting from the start codon.
  - A *deletion* from position **106** to position **120**.

Introduction

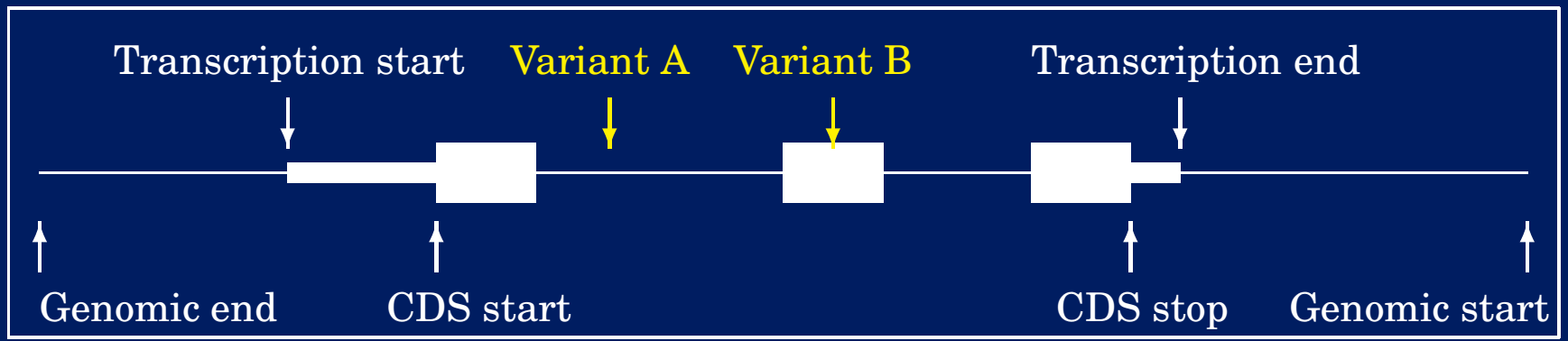
Coordinate systems



Name	g.	n.	c.
Genomic start	1	100+d70	*10+d70
Genomic end	300	1-u50	-30-u50
Transcription start	250	1	-30
Transcription end	70	100	*10
CDS start	220	30	1
CDS stop	80	90	60

Introduction

Coordinate systems

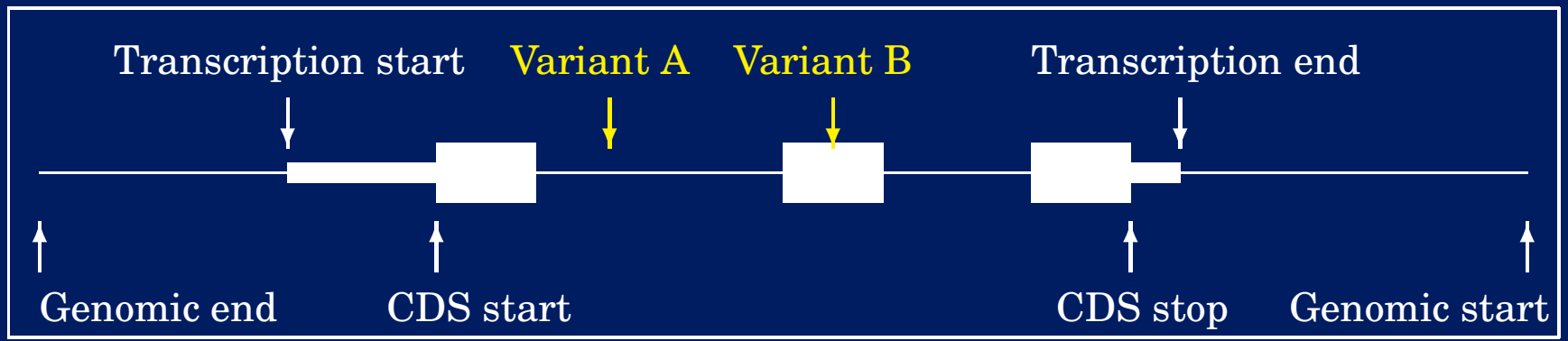


c. positions:

- Positions in introns are relative to the nearest exonic position.
- Positions before the CDS are indicated with a **-** sign.
- Positions after the CDS are indicated with a ***** sign.

Introduction

Coordinate systems



`c.` positions:

- Positions in introns are relative to the nearest exonic position.
- Positions before the CDS are indicated with a **`-`** sign.
- Positions after the CDS are indicated with a **`*`** sign.
- Position **`-1`** and **`1`** are adjacent.
- If **`60`** is the last position of the CDS, then **`60`** and **`*1`** are adjacent.

Interfaces

User friendly interfaces

Name checker	-	Full nomenclature / semantic check.
Syntax checker	-	Only nomenclature check.
Position converter	-	Mapping, lifting over (build / transcripts).
SNP converter	-	DbSNP rsId to HGVS.
Name generator	-	Point and click to make a description.
GenBank Uploader	-	Custom reference sequences.

Interfaces

User friendly interfaces

Name checker	-	Full nomenclature / semantic check.
Syntax checker	-	Only nomenclature check.
Position converter	-	Mapping, lifting over (build / transcripts).
SNP converter	-	DbSNP rsId to HGVS.
Name generator	-	Point and click to make a description.
GenBank Uploader	-	Custom reference sequences.

Bulk / RPC interfaces:

- Upload a table (CSV, Excel, Open Office Spreadsheet):
  - Name checker.
  - Syntax checker.
  - Position converter.
  - SNP converter.
- Webservices (SOAP).
  - 22 functions available.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Name checker.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Syntax checker.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Position converter.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

SNP converter.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Name generator.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

GenBank Uploader.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Added when using a batch interface.

Components

Core components and their usage

Config	-	Parsing config file.
Crossmap	-	Position conversions.
Db	-	Mapping, linking, queues, caching info.
File	-	CSV, Excel, OpenOffice tables.
GenRecord	-	Abstraction of annotated reference sequences.
GBparser	-	Instance of GenRecord (GenBank files).
LRGparser	-	Instance of GenRecord (LRG files).
Misc	-	
Mutator	-	Modify the reference sequence and annotation.
Output	-	Communication with the interfaces.
Parser	-	HGVS nomenclature parser.
Retriever	-	Retrieve / cache reference sequences.
Scheduler	-	Batch jobs scheduler.
Serializers	-	SOAP definitions of complex objects.

Added when using webservices.

Webservices

Life without webservices

Webservices

Life without webservices

Example: Get the first hit in google:

- Figure out what the server expects.
- `http://www.google.com/#q=test`
- Parse the resulting HTML file.

Webservices

Life without webservices

Example: Get the first hit in google:

- Figure out what the server expects.
- **`http://www.google.com/#q=test`**
- Parse the resulting HTML file.

Disadvantages:

- The communication variables can change (**`q`** changes to **`query`**).
- The resulting HTML file can change.

Conclusions:

- Requires quite some expertise to set up.
- Requires a lot of maintenance.

Webservices

SOAP webservices

Characteristics of the SOAP webservice:

- Communication XML/RPC over HTTP (not necessarily over port 80).
- Description of the interface is machine readable.

Communication over HTTP is essential for us (firewall etc.).

Webservices

SOAP webservices

Characteristics of the SOAP webservice:

- Communication XML/RPC over HTTP (not necessarily over port 80).
- Description of the interface is machine readable.

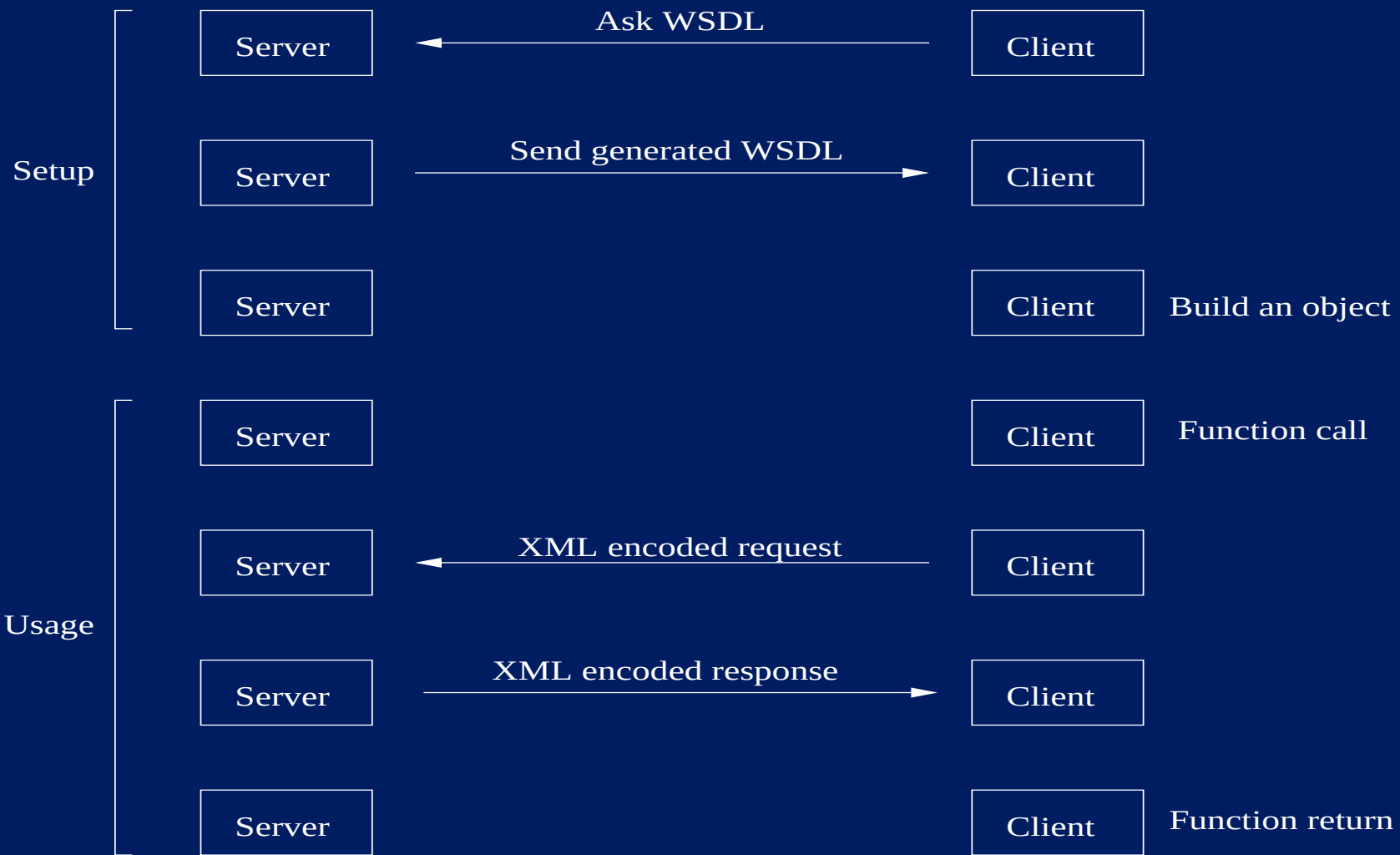
Communication over HTTP is essential for us (firewall etc.).

The description of the interface is machine readable:

- The communication protocol can be abstracted.
  - The actual communication can change without the client being aware of it.
  - Functions can be added without a need for the client to update.

Webservices

SOAP webservices



Webservices

SOAP webservices

- The transport/communications layer is completely hidden from both the client as well as the server.
- Exported functions are normal local functions on the server side (makes testing easy).
- The client sees the functions as local functions (not different from functions included from a library).

Webservices

An example

```
1  @soapmethod(String, Integer, _returns = String)
2  def sayHello(name, times):
3      return ("Hello " + name + ' ') * times
```

Listing 1: Server side

```
1  from SOAPpy import WSDL
2  service = WSDL.Proxy("http://path_to_wsdl.wsdl")
3  print service.sayHello("MyName", 10)
```

Listing 2: Client side

Webservices

An example

```
1  @soapmethod(String, Integer, _returns = String)
2  def sayHello(name, times):
3      return ("Hello " + name + ' ') * times
```

Listing 1: Server side

```
1  from SOAPpy import WSDL
2  service = WSDL.Proxy("http://path_to_wsdl.wsdl")
3  print service.sayHello("MyName", 10)
```

Listing 2: Client side

```
1  from Bio import pairwise2
2  print pairwise2.align("AAAATT", "AATAA")
```

Listing 3: Local function (for comparison)

Webservices

Discovery

- The client object has a standard function that gives a list of function names and a description of the parameters.
- The WSDL file also contains full documentation (defined on the server).
- We also generate documentation from the source code on the website.
- Tools for viewing the WSDL are also available.

Webservices

Discovery

- The client object has a standard function that gives a list of function names and a description of the parameters.
- The WSDL file also contains full documentation (defined on the server).
- We also generate documentation from the source code on the website.
- Tools for viewing the WSDL are also available.

```
1  from SOAPpy import WSDL
2  service = WSDL.Proxy("http://path_to_wsdl.wsdl")
3  print service.show_methods()
```

Listing 4: WSDL

Examples

Small tools

Function	Description	Application
checkSyntax	Check the validity of the HGVS description.	Textmining
getdbSNPDescriptions	Get all HGVS descriptions from an rs number.	
getTranscriptsAndInfo	Get the transcripts of all genes and their info.	Gene locations Gene info
numberConversion	Convert from **c.** to **g.** or vice versa.	Mapping

Examples

Simulated reads

Idea:

- Apply variations to a chromosome.
- Generate paired-end reads from the mutated sequence.
- Map the reads.
- See how much variants are detected.

Examples

Simulated reads

Idea:

- Apply variations to a chromosome.
- Generate paired-end reads from the mutated sequence.
- Map the reads.
- See how much variants are detected.

Input:

- List of variants for a chromosome slice.
- Coordinates for the genomic slice.

Examples

Simulated reads

Idea:

- Apply variations to a chromosome.
- Generate paired-end reads from the mutated sequence.
- Map the reads.
- See how much variants are detected.

Input:

- List of variants for a chromosome slice.
- Coordinates for the genomic slice.

Workflow:

- sliceChromosome — Select the slice.
- runMutalyzer — Apply the variants, receive the mutated sequence.

`https://humgenprojects.lumc.nl/svn/sim-reads`

Questions?

Acknowledgements:

Martijn Vermaat
Gerben Stouten
Gerard Schaafsma
Ivo Fokkema
Jacopo Celli
Peter Taschner
Johan den Dunnen

http://www.mutalyzer.nl/