

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the date.

2016 年 2 月 26 日

2015 年度

多階層オミクス情報

RDF を利用した

SPARQL 検索の

高速化ならびに

Stanza の開発

操作説明書

Several thin, curved, light blue lines that sweep upwards from the bottom left towards the center of the page.

田中 聡

Trans-IT

目次

1. はじめに	1
2. 必要環境	1
3. 多階層オミクスデータの RDF 化プログラムの開発	1
3.1. プログラムの入手	1
3.2. プログラムのビルド	1
3.3. プログラムの実行	1
3.4. 入力ファイル	2
3.5. ds02 サーバー以下の実行環境	5
4. RDBに蓄積されている多階層オミクスデータの SPARQL 検索を可能にするためのマッピングファイルの作成	5
4.1. プログラムの入手	5
4.2. プログラムのビルド	5
4.3. プログラムの実行	6
4.4. 入力ファイル	7
4.5. ds02 サーバー以下の実行環境	10
5. ゲノム座標情報を簡便に検索するための SPARQL ライブラリの開発	11
5.1. 使用方法	11
5.2. 関数一覧	12
5.2.1. dbclsSparql.setEndpoint	12
5.2.2. dbclsSparql.exec	12
5.2.3. dbclsSparql.executeQuery	12
5.2.4. dbclsSparql.addTableFromObjects	13
5.2.5. dbclsFaldo.getRegion	13
5.2.6. dbclsFaldo.getInclusionRelation	14
5.2.7. dbclsFaldo.isOverlapping	14
5.2.8. dbclsFaldo.isIncluding	14
5.2.9. dbclsFaldo.getRegionsInRange	15
5.2.10. dbclsFaldo.getUpstreamRegions	15
5.3. サンプルプログラム	16
6. SPARQL 検索結果を表示する為の Stanza の開発	18
6.1. 機能一覧	19
6.2. 指定位置アレル表示 (allele)	19
6.3. 指定アレル存在判定 (beacon)	21
6.4. 指定区間 SNV 一覧表示 (snv)	22

1. はじめに

本文書は 2015 年度「多階層オミクス情報 RDF を利用した SPARQL 検索の高速化ならびに Stanza の開発」の操作の手順を説明したものである。

2. 必要環境

本プログラムを使用するにあたっての、必要な実行環境は以下の通りである。

- Java のビルドおよび実行環境 (Java 1.6.0 以降)
- JavaScript をサポートしたブラウザ (Google Chrome, Safari, Firefox, Internet Explorer など)

3. 多階層オミクスデータの RDF 化プログラムの開発

本章では VCF を RDF データに変換する為のプログラムについての操作説明を行なう。

3.1. プログラムの入手

本プログラムはソース管理管理サイト GitHub (<https://github.com/>) より取得できる。
ソースの取得はバージョン管理システムプログラムである Git を用いる。

```
% git clone https://github.com/DBCLS-human/multiomics15.git
```

GitHub のアカウントを既に持っていて、SSH 認証鍵を既に登録している場合は SSH を用いて取得する事もできる。

```
% git clone git@github.com:DBCLS-human/multiomics15.git
```

3.2. プログラムのビルド

変換プログラムは git より取得した multiomics15 リポジトリの src/java 以下に格納されている。変換プログラムは src/java フォルダに移動した後、javac コマンドでビルドする事ができる。

```
% javac jp/dbcls/rdf/vcf/Vcf2Rdf.java
```

3.3. プログラムの実行

プログラムの実行は java コマンドにより行なう。

```
% java -classpath [src/java フォルダパス] jp.dbcls.rdf.vcf.Vcf2Rdf [オプション]
```

Vcf2Rdf のオプションは以下のものが存在する。

表 1 Vcf2Rdf オプション

オプション名	説明
in	入力ファイルパス (必須)
out	出力ファイルパス (必須)
sample	サンプル名 (必須)
template	テンプレートファイルパス (任意) 指定しない場合はデフォルトのテンプレートファイルが指定される。(multiomics15 リポジトリフォルダ以下の src/java/jp/dbcls/rdf/vcf/template.ttl)

実行例

```
% java -classpath dev/multiomics25/src/java jp.dbcls.vcf.Vcf2Rdf -in data.vcf
-out data.ttl -sample TSE000086 -template template.ttl
```

3.4. 入力ファイル

VCF ファイルは以下の構造を持つ。

Chrom	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	s_18
c1.fa	10234	rs145599635	C	T	72.66	.	AC=1;AF=0.50;AN=2;BaseQRankSum=-0.777;DB=53;Dels=0.02;FS=1.862;HRun=1;HaplotypeScore=43.1232;MQ=26.10;MQ0=1;MQRankSum=3.530;QD=1.37;ReadPosRankSum=2.915;SB=-42.24	GT:AD:DP:GQ:PL	0/1:42,10:52:61.94:103,0,62
c1.fa	10235	.	T	A	106.33	.	AC=1;AF=0.50;AN=2;BaseQRankSum=0.773;DP=52;Dels=0.02;FS=1.922;HRun=2;HaplotypeScore=39.1992;MQ=26.04;MQ0=1;MQRankSum=3.142;QD=2.04;ReadPosRankSum=3.043;SB=-42.24	GT:AD:DP:GQ:PL	0/1:40,9:51:21.93:136,0,22
c1.fa	14907	rs79585140	A	G	101.53	.	AC=1;AF=0.50;AN=2;BaseQRankSum=0.724;DB=DP=82;Dels=0.00;FS=5.251;HRun=1;HaplotypeScore=0.0000;MQ=17.43;MQ0=47;MQRankSum=2.257;QD=1.24;ReadPosRankSum=1.406;SB=-64.97	GT:AD:DP:GQ:PL	0/1:74,8:82:99:132,0,125
...									

各々の行が SNV のデータに相当するが、これをテンプレートファイルに従って出力する。テンプレートファイルは以下の構造をしている。

@prefix rdf:	<http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:	<http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl:	<http://www.w3.org/2002/07/owl#> .
@prefix obo:	<http://purl.obolibrary.org/obo/> .
@prefix dc:	<http://purl.org/dc/terms/> .
@prefix kero:	<http://dbtss.hgc.jp/rdf/ontology/> .

```

@prefix faldo:          <http://biohackathon.org/resource/faldo#> .
@prefix dbsnp:          <http://info.identifiers.org/dbsnp/> .
@prefix ncbisnp:        <http://www.ncbi.nlm.nih.gov/projects/SNP/snp_ref.cgi?rs=> .
@prefix ensemblvariation: <https://github.com/simonjupp/ensembl-
rdf/blob/master/ontology/ensembl_variation_ontology.owl#> .

<http://dbtss.hgc.jp/rdf/experiment/{{Experiment}}>
  a kero:Experiment.

[vcf]
<http://dbtss.hgc.jp/rdf/experiment/{{Experiment}}> kero:has_SNV
<http://dbtss.hgc.jp/rdf/data/{{Experiment}}/{{Chrom}}:{{Pos}}> .

<http://dbtss.hgc.jp/rdf/data/{{Experiment}}/{{Chrom}}:{{Pos}}> a obo:SO_0001483;
  rdfs:label "variation on chr{{Chrom}}:{{Pos}} from {{Experiment}}";
  faldo:location <http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}:{{Pos}}-{{Pos}}:1>;
  kero:additionalInformation "{{Info}}";
  kero:alleleFrequency "{{Info|AF}}";
  kero:allelicDepths "{{Detail:AD}}";
  kero:dbsnpID "{{Id0}}";
  kero:dbsnpID "{{Id1}}";
  kero:dbsnpID "{{Id2}}";
  kero:dbsnpID "{{Id3}}";
  kero:dbsnpID "{{Id4}}";
  kero:dbsnpID "{{Id5}}";
  kero:dbsnpID "{{Id6}}";
  kero:dbsnpID "{{Id7}}";
  kero:dbsnpID "{{Id8}}";
  kero:dbsnpID "{{Id9}}";
  kero:genotype "{{Detail:GT}}";
  kero:genotypeData "{{Format}}|{{Detail}}";
  kero:genotypeQuality {{Detail:GQ}};
  kero:quality {{Qual}};
  dc:identifier "{{Chrom}}:{{Pos}}";
  rdfs:seeAlso dbsnp:{{Id0}}, ncbisnp:{{Id0}};
  rdfs:seeAlso dbsnp:{{Id1}}, ncbisnp:{{Id1}};
  rdfs:seeAlso dbsnp:{{Id2}}, ncbisnp:{{Id2}};
  rdfs:seeAlso dbsnp:{{Id3}}, ncbisnp:{{Id3}};
  rdfs:seeAlso dbsnp:{{Id4}}, ncbisnp:{{Id4}};
  rdfs:seeAlso dbsnp:{{Id5}}, ncbisnp:{{Id5}};
  rdfs:seeAlso dbsnp:{{Id6}}, ncbisnp:{{Id6}};
  rdfs:seeAlso dbsnp:{{Id7}}, ncbisnp:{{Id7}};
  rdfs:seeAlso dbsnp:{{Id8}}, ncbisnp:{{Id8}};
  rdfs:seeAlso dbsnp:{{Id9}}, ncbisnp:{{Id9}};
  ensemblvariation:has_allele
<http://dbtss.hgc.jp/rdf/data/{{Experiment}}/{{Chrom}}:{{Pos}}#{{Ref}}>,
  <http://dbtss.hgc.jp/rdf/data/{{Experiment}}/{{Chrom}}:{{Pos}}#{{Alt}}> .

<http://dbtss.hgc.jp/rdf/data/{{Experiment}}/{{Chrom}}:{{Pos}}#{{Ref}}>
  a ensemblvariation:reference_allele, ensemblvariation:ancestral_allele;
  rdfs:label "{{Experiment}} chr{{Chrom}}:{{Pos}} allele {{Ref}}";
  dc:identifier "{{Chrom}}:{{Pos}}#{{Ref}}".

<http://dbtss.hgc.jp/rdf/data/{{Experiment}}/{{Chrom}}:{{Pos}}#{{Alt}}>

```

```

a ensemblvariation:derived_allele;
rdfs:label "{{Experiment}} chr{{Chrom}}:{{Pos}} allele {{Alt}}";
dc:identifier "{{Chrom}}:{{Pos}}#{{Alt}}".

<http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}:{{Pos}}:1>
a faldo:Region;
rdfs:label "GRCh38 chr{{Chrom}}:{{Pos}} Forward";
faldo:begin <http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}:{{Pos}}:1>;
faldo:end <http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}:{{Pos}}:1>;
faldo:reference <http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}>;
dc:identifier "chromosome:GRCh38:{{Chrom}}:{{Pos}}:1".

<http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}:{{Pos}}:1>
a faldo:ExactPosition, faldo:ForwardStrandPosition;
rdfs:label "RCh38 chr{{Chrom}}:{{Pos}} Forward";
faldo:position {{Pos}};
faldo:reference <http://dbtss.hgc.jp/rdf/location/chromosome:GRCh38:{{Chrom}}>;
dc:identifier "chromosome:GRCh38:{{Chrom}}:{{Pos}}:1".

```

各々の SNV について出力される部分については、テンプレートファイルの [snv] 以下に記述する。

また、各々の SNV や実行パラメータ毎に違う部分については {{パラメータ名}} で記述する。テンプレートファイルのパラメータには以下のものがある。

表 2 Vcf2Rdf テンプレートファイル パラメータ

パラメータ名	説明
Experiment	Vcf2Rdf の -sample で指定したサンプル名
Chrom	VCF ファイルの Chromo 列の数値部分もしくは X, Y, M
Pos	VCF ファイルの Pos 列の値
Id0 ~ Id1	VCF ファイルの ID 列の値。ここはセミコロン(;) 区切りで複数の値が入るので 1 個目は Id0, 2 個目は Id1, ... , 10 個目は Id9 に格納される。
Ref	VCF ファイルの REF 列の値。
Alt	VCF ファイルの ALT 列の値
Info	VCF ファイルの INFO 列の値
Format	VCF ファイルの FORMAT 列の値
Detail	VCF ファイルの s_18 列の値

値が存在しない部分については出力されない。

また特殊な記述方法として Info 列は (パラメータ名)=(値) の記述がセミコロン(;) 区切り

で、Detail については Format 列で記述されている名のパラメータがコロン(:) 区切りで記述されている。

Info 列の各々の値は {{Info|(パラメータ名)}}で、Detail 列の各々の値は {{Detail:(パラメータ名)}} で取得する事ができる。

例) {{Info|AF}}, {{Detail:GQ}}

このファイルは新たに作成して `-template` オプションで指定する方法と、`multiomics15` リポジトリフォルダ以下の `src/java/jp/dbcls/rdf/vcf/template.ttl` を編集する方法とがある。

3.5. ds02 サーバー以下の実行環境

DBCLS のサーバー ds02 には `/opt/services/RDF-SPARQL/dev/multiomics15` にリポジトリフォルダが存在する。

また、このプログラムを簡単に実行する為のシェルスクリプトが `/opt/services/RDF-SPARQL/sh/vcf2rdf.sh` に存在する。

実行例

```
% /opt/services/RDF-SPARQL/sh/vcf2rdf.sh -sample TSE000086 -in data.vcf  
-out data.ttl -template template.ttl
```

4. RDB に蓄積されている多階層オミクスデータの SPARQL 検索を可能にするためのマッピングファイルの作成

本章では `d2rq` および `ontop` で RDB に格納されている情報に対して SPARQL 検索を行なう為の `mapping` ファイルを作成する為のプログラムの使用法について記述する。

4.1. プログラムの入手

マッピングファイルの作成プログラムは `vcf2rdf` プログラムと同様 GitHub で公開されている。

入手方法については 3.1 を参照

4.2. プログラムのビルド

マッピングファイル作成プログラムは `vcf2rdf` プログラムと同様 `multiomics15` フォルダ

以下の `src/java` 以下に格納されている。ビルドは `src/java` フォルダに移動し `javac` プログラムで行なう。

```
% javac jp/dbcls/rdf/mapping/MapCreator.java
```

4.3. プログラムの実行

プログラムの実行は `java` コマンドにより行なう。

```
% java -classpath [src/java フォルダパス] jp.dbcls.rdf.mapping.MapCreator  
[オプション]
```

`MapCreator` のオプションは以下のものが存在する。

表 1 `MapCreator` オプション

オプション名	説明
props	データベース接続およびテーブル設定ファイル。 (任意) 指定しない場合は <code>multiomics15</code> リポジトリフォルダ以下の <code>src/java/jp/dbcls/rdf/mapping/tables.properties</code> を用いる
out	出力ファイルパス (必須)
template	テンプレートファイルパス (必須) 指定したファイルが <code>Current</code> フォルダから見て存在しない場合は <code>multiomics15</code> リポジトリフォルダ以下の <code>src/java/jp/dbcls/rdf/mapping</code> 以下を検索する。 上記のフォルダにデフォルトで用意されているテンプレートファイルとして以下のファイルがある。 <code>vcf-lc2.n3.template</code> : d2rq 用テンプレートファイル <code>vcf-lc2.owl.template</code> : ontop 様 owl テンプレートファイル <code>vcf-lc2.obda.template</code> : ontop 用 obda テンプレートファイル

実行例


```
% java -classpath dev/multiomics25/src/java jp.dbcls.mapping.MapCreator
-template vcf-lc2.obda.template -props tables.properties -out ontop.obda
```

4.4. 入力ファイル

データベース接続、テーブル情報ファイルは以下の様に記述する。

```
SQLServer=<server>
Database=<database>
User=<user>
Password=<password>
BSDDataTable=bs_data_9696_LC2ad
ChipInfTable=chip_inf_9606_chr1_LC2ad_h3k4me3,¥
chip_inf_9606_chr2_LC2ad_h3k4me3,¥
chip_inf_9606_chr3_LC2ad_h3k4me3,¥
chip_inf_9606_chr4_LC2ad_h3k4me3,¥
chip_inf_9606_chr5_LC2ad_h3k4me3,¥
chip_inf_9606_chr6_LC2ad_h3k4me3,¥
chip_inf_9606_chr7_LC2ad_h3k4me3,¥
chip_inf_9606_chr8_LC2ad_h3k4me3,¥
chip_inf_9606_chr9_LC2ad_h3k4me3,¥
chip_inf_9606_chr10_LC2ad_h3k4me3,¥
chip_inf_9606_chr11_LC2ad_h3k4me3,¥
chip_inf_9606_chr12_LC2ad_h3k4me3,¥
chip_inf_9606_chr13_LC2ad_h3k4me3,¥
chip_inf_9606_chr14_LC2ad_h3k4me3,¥
chip_inf_9606_chr15_LC2ad_h3k4me3,¥
chip_inf_9606_chr16_LC2ad_h3k4me3,¥
chip_inf_9606_chr17_LC2ad_h3k4me3,¥
chip_inf_9606_chr18_LC2ad_h3k4me3,¥
chip_inf_9606_chr19_LC2ad_h3k4me3,¥
chip_inf_9606_chr20_LC2ad_h3k4me3,¥
chip_inf_9606_chr21_LC2ad_h3k4me3,¥
chip_inf_9606_chr22_LC2ad_h3k4me3,¥
chip_inf_9606_chrM_LC2ad_h3k4me3,¥
chip_inf_9606_chrX_LC2ad_h3k4me3,¥
chip_inf_9606_chrY_LC2ad_h3k4me3
ChromhmmInfTable=chromhmm_inf_9606_LC2ad
CpgBincountTable=cpg_bincount_9606_ucsc
RefgeneTable=refgene_9606
RnaseqQtyTable=maseq_qty_9606_LC2ad
TssBincountTable=tss_bincount_9606_chr1_LC2ad,¥
tss_bincount_9606_chr2_LC2ad,¥
tss_bincount_9606_chr3_LC2ad,¥
tss_bincount_9606_chr4_LC2ad,¥
tss_bincount_9606_chr5_LC2ad,¥
tss_bincount_9606_chr6_LC2ad,¥
tss_bincount_9606_chr7_LC2ad,¥
tss_bincount_9606_chr8_LC2ad,¥
tss_bincount_9606_chr9_LC2ad,¥
tss_bincount_9606_chr10_LC2ad,¥
```

```

tss_bincount_9606_chr11_LC2ad,¥
tss_bincount_9606_chr12_LC2ad,¥
tss_bincount_9606_chr13_LC2ad,¥
tss_bincount_9606_chr14_LC2ad,¥
tss_bincount_9606_chr15_LC2ad,¥
tss_bincount_9606_chr16_LC2ad,¥
tss_bincount_9606_chr17_LC2ad,¥
tss_bincount_9606_chr18_LC2ad,¥
tss_bincount_9606_chr19_LC2ad,¥
tss_bincount_9606_chr20_LC2ad,¥
tss_bincount_9606_chr21_LC2ad,¥
tss_bincount_9606_chr22_LC2ad,¥
tss_bincount_9606_chrM_LC2ad,¥
tss_bincount_9606_chrX_LC2ad,¥
tss_bincount_9606_chrY_LC2ad
VcfInfTable=vcf_inf_9606_chr1_LC2ad,¥
vcf_inf_9606_chr2_LC2ad,¥
vcf_inf_9606_chr3_LC2ad,¥
vcf_inf_9606_chr4_LC2ad,¥
vcf_inf_9606_chr5_LC2ad,¥
vcf_inf_9606_chr6_LC2ad,¥
vcf_inf_9606_chr7_LC2ad,¥
vcf_inf_9606_chr8_LC2ad,¥
vcf_inf_9606_chr9_LC2ad,¥
vcf_inf_9606_chr10_LC2ad,¥
vcf_inf_9606_chr11_LC2ad,¥
vcf_inf_9606_chr12_LC2ad,¥
vcf_inf_9606_chr13_LC2ad,¥
vcf_inf_9606_chr14_LC2ad,¥
vcf_inf_9606_chr15_LC2ad,¥
vcf_inf_9606_chr16_LC2ad,¥
vcf_inf_9606_chr17_LC2ad,¥
vcf_inf_9606_chr18_LC2ad,¥
vcf_inf_9606_chr19_LC2ad,¥
vcf_inf_9606_chr20_LC2ad,¥
vcf_inf_9606_chr21_LC2ad,¥
vcf_inf_9606_chr22_LC2ad,¥
vcf_inf_9606_chrM_LC2ad,¥
vcf_inf_9606_chrX_LC2ad,¥
vcf_inf_9606_chrY_LC2ad
WRnaseqTable=w_rnaseq_refgene

```

固定のパラメータは以下の通り

表 2 Mapping ファイル作成プログラム設定ファイルパラメータ

パラメータ名	説明
SQLServer	MySQL データベースサーバー
Database	データベース名
User	データベース接続ユーザー

Password	データベース接続パスワード
----------	---------------

テーブル名の定義については任意のパラメータ名で記述する事ができる。
 その際、同じスキーマ(列)を持つテーブルについては複数定義して、プログラム実行時に
 各々のテーブルについて一括で出力する事ができる。その際、複数テーブルはカンマ(,) 区
 切りで記述する。

また、テンプレートファイルについては以下の様な方式で記述する。

```

ap:database a d2rq:Database;

    d2rq:jdbcDriver "com.mysql.jdbc.Driver";
    d2rq:jdbcDSN "jdbc:mysql://{{SQLServer}}/{{Database}}";
    d2rq:username "{{User}}";
    d2rq:password "{{Password}}";
    jdbc:autoReconnect "true";
    jdbc:zeroDateTimeBehavior "convertToNull";
    .

# Experiment
map:tse000086 a d2rq:ClassMap;

    d2rq:dataStorage map:database;
    d2rq:uriPattern "http://dbtss.hgc.jp/rdf/experiment/TSE000086";
    d2rq:class kero:Experiment;
    .

# SNV
[VcfInfTable]
map:{{VcfInfTable}} a d2rq:PropertyBridge;

    d2rq:belongsToClassMap map:tse000086;
    d2rq:property kero:has_SNV;
    d2rq:uriSqlExpression "concat('http://dbtss.hgc.jp/rdf/data/TSE000086/',
substring({{VcfInfTable}}.chr, 4), ' ', {{VcfInfTable}}.pos)";
    .

map:{{VcfInfTable}}_class a d2rq:ClassMap;

    d2rq:dataStorage map:database;

```

```

d2rq:uriSqlExpression "concat('http://dbtss.hgc.jp/rdf/data/TSE000086/',
substring({{VcfInfTable}}.chr, 4), ':', {{VcfInfTable}}.pos)";

d2rq:class obo:SO_0001483;

.

map:{{VcfInfTable}}_label a d2rq:PropertyBridge;

d2rq:belongsToClassMap map:{{VcfInfTable}}_class;

d2rq:property rdfs:label;

d2rq:pattern "variation on @@{{VcfInfTable}}.chr@:@: @@{{VcfInfTable}}.pos@@ from
TSE000086";
[/VcfTable]

```

ゲノム座標情報を簡便に検索するための SPARQL ライブラリの開発
 SPARQL 検索結果を表示する為の Stanza の開発

設定ファイルに記述されているパラメータを用いる場合は {{パラメータ名}} で記述する。
 また、複数テーブルを定義している場合は

[テーブルグループ名] と [/テーブルグループ名] (例: [VcfTable] と [/VcfTable]) を記述した行で挟む事によって、各々のテーブルについて出力する事ができる。

また、その際の各々のテーブル名については {{テーブルグループ名}} で記述する事ができる。

例えば上記の設定ファイルの場合 VcfTable は vcf_inf_9606_chr1_LC2ad, vcf_inf_9606_chr2_LC2ad, ... , vcf_inf_9606_chrY_LC2ad のそれぞれを出力する。

4.5. ds02 サーバー以下の実行環境

mapping ファイル作成のプログラムも vcf2rdf と同様に /opt/services/RDF-SPARQL/dev/multiomics15 以下にリポジトリフォルダが存在する。

また、このプログラムを簡単に実行する為のシェルスクリプトが /opt/services/RDF-SPARQL/sh/create_map.sh に存在する。

実行例

```
% /opt/services/RDF-SPARQL/sh/create_map.sh --template vcf-lc2.obda.template  
-props tables.properties --out ontop.obda
```

5. ゲノム座標情報を簡便に検索するための SPARQL ライブラリの開発

ここでは、区間情報が格納されている SPARQL DB を検索する為の JavaScript ライブラリの使用方法を説明する。

SPARQL 関連の一般的な機能や、区間検索に関する機能を提供している。

5.1. 使用方法

本ライブラリを使用するにあたっては、まず以下のライブラリが必須である。

- Ajax
- jQuery

それらを使用する為に、まずはライブラリを使用する HTML ファイルの head タグに以下のコードを埋め込む。

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js"  
type="text/javascript"></script>  
  
<script src="http://code.jquery.com/jquery-1.11.0.min.js"></script>
```

そして、本ライブラリの SPARQL 関連機能を使用する場合には 以下のコードを埋め込む。

```
<script src="http://humanrdf.dbcls.jp/js/sparql.js" type="text/javascript"></script>
```

ゲノム座標検索関連の機能を使用する場合には さらに以下のコードを埋め込む。

```
<script src="http://humanrdf.dbcls.jp/js/faldo.js" type="text/javascript"></script>
```

これら全てを埋め込んだコードは以下の様になる。

```
<!DOCTYPE html>  
<html lang="ja">  
  <head>  
    <title>本文タイトル</title>  
    <meta charset="utf-8">
```

```

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js" type="text/javascript"
></script>
<script src="http://code.jquery.com/jquery-1.11.0.min.js"></script>
<script src="http://humanrdf.dbcls.jp/js/sparql.js"></script>
<script src="http://humanrdf.dbcls.jp/js/faldo.js"></script>
<script>
    自分が行ないたい JavaScript の処理
</script>
</head>
<body>
    本文
</body>
</html>

```

5.2. 関数一覧

本ライブラリは以下の関数を提供する。

5.2.1. dbclsSparql.setEndpoint

ファイル: sparql.js

説明: Endpoint の URL をセットする。

引数:

endpoint

Endpoint URL。何もセットしない時には

http://humanrdf.dbcls.jp/sparql を使用する。

戻り値: なし

5.2.2. dbclsSparql.exec

ファイル: sparql.js

説明: SPARQL を実行する。

引数:

sparql

SPARQL 文字列。

戻り値:

SPARQL 実行結果 (Ajax ライブラリにより得られるオブジェクト)

5.2.3. dbclsSparql.executeQuery

ファイル: sparql.js

説明: SPARQL クエリーを実行し、オブジェクト配列を得る。

引数:

sparql

SPARQL 文字列。

戻り値:

オブジェクト配列。

オブジェクトが持つプロパティは SPARQL 文に依存する。

例えば "select ?name ?age ?birthday where { ..." の様な SPARQL 文の場合には
オブジェクトはプロパティとして name, age, birthday を持つ。

5.2.4. dbclsSparql.addTableFromObjects

ファイル: sparql.js

説明: オブジェクト配列からテーブルを追加する。

引数:

box

テーブルを加える要素のオブジェクト。

jQuery の \$(セレクタ) により取得する。

classPrefix

テーブルの要素に追加するクラス名の接頭辞。

例えば classPrefix に 'dbcls' を要素を指定すると、

追加するテーブルやその行、セルにクラス dbcls_table, dbcls_row, dbcls_cell,
dbcls_header_row, dbcls_header_cell, dbcls_content_row, dbcls_content_cell,
dbcls_content_odd_row, dbcls_content_even_row が付与される。

objects

オブジェクト配列

戻り値:

なし

5.2.5. dbclsFaldo.getRegion

ファイル: faldo.js

説明: 区間オブジェクトを取得する。

引数:

name

区間オブジェクト名 (e.g., A549H3K27ac_peak32)

classPrefix

戻り値:

指定した名前を持つ区間オブジェクト。

区間オブジェクトはプロパティとして name (オブジェクト名), label (ラベル),
chromosome (染色体), type (forward か reverse か), begin (区間開始位置), end (区

間終了位置)

5.2.6. dbclsFaldo.getInclusionRelation

ファイル: faldo.js

説明: 2つの区間オブジェクトの包含関係を取得する。

引数:

name1

1 目の区間オブジェクト名 (e.g., A549H3K27ac_peak32)

name2

2 目の区間オブジェクト名 (e.g., A549H3K27ac_peak32)

戻り値:

2つの区間オブジェクトの包含関係。

戻り値は、dbclsFaldo.InclusionRelation.NOT_EXIST (存在しない),

dbclsFaldo.InclusionRelation.DIFFERENT_REFERENCE (違う染色体),

dbclsFaldo.InclusionRelation.SAME_REFERENCE (染色体は同じだが重ならない),

dbclsFaldo.InclusionRelation.OVERLAPPING (重なっている),

dbclsFaldo.InclusionRelation.INCLUDING (どちらかが、もう片方に完全に含まれている)

5.2.7. dbclsFaldo.isOverlapping

ファイル: faldo.js

説明: 2つの区間オブジェクトが重なっているか否かを判定する。

引数:

name1

1 目の区間オブジェクト名 (e.g., A549H3K27ac_peak32)

name2

2 目の区間オブジェクト名 (e.g., A549H3K27ac_peak32)

戻り値:

2つの区間オブジェクトが重なっている、もしくは どちらかがもう片方に完全に含まれているときは **true**。 重なっていない時には **false**。

5.2.8. dbclsFaldo.isIncluding

ファイル: faldo.js

説明:

2つの区間オブジェクトにおいて、片方の区間がもう片方の区間に完全に含まれているか否かを判定する。

引数:

name1

1 目の区間オブジェクト名 (e.g., A549H3K27ac_peak32)

name2

2 目の区間オブジェクト名 (e.g., A549H3K27ac_peak32)

戻り値:

片方の区間がもう片方の区間に完全に含まれていれば **true**。

そうでなければ **false**。

5.2.9. dbclsFaldo.getRegionsInRange

ファイル: faldo.js

説明:

指定範囲に存在する区間オブジェクトを取得する。

引数:

chrom

検索対象染色体。(e.g., 'chr1')

start

指定範囲、開始位置

end

指定範囲、終了位置

include

区間オブジェクトが指定範囲に完全に 含まれている必要がある場合には **true**。

そうでない場合は **false**。

戻り値:

指定範囲に存在する区間オブジェクトの配列。。

5.2.10. dbclsFaldo.getUpstreamRegions

ファイル: faldo.js

説明:

指定区間の上流に含まれる区間オブジェクトを取得する。

引数:

region

検索の基準となる区間オブジェクト名。(e.g., A549H3K27ac_peak32)

length

検索する基準からの範囲の長さ。

include

区間オブジェクトが指定範囲に完全に 含まれている必要がある場合には **true**。
そうでない場合は **false**。

戻り値:

指定範囲に存在する区間オブジェクトの配列。。

5.3. サンプルプログラム

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <title>JS Demo</title>
    <meta charset="utf-8">

    <!-- テーブルのデザイン -->
    <style>
      .demo_table {
        border: solid 1px #000000;
      }

      .demo_cell {
        border: solid 1px #000000;
      }

      .demo_header_cell {
        background-color: #000000;
        color: #ffffff;
        font-weight: bold;
      }

      .demo_content_odd_row {
        background-color: #e0e0e0;
      }

      .demo_content_even_row {
        background-color: #ffffff;
      }
    </style>

    <!-- 必要ライブラリ -->
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js" type="text/javascript"></script>
    <script src="http://code.jquery.com/jquery-1.11.0.min.js"></script>
    <script src="http://humanrdf.dbcls.jp/js/sparql.js"></script>
    <script src="http://humanrdf.dbcls.jp/js/faldo.js"></script>
    <script>
      // 読み込み時間関数定義
      function ready() {
        // Endpoint セット
        var endpoint = 'http://humanrdf.dbcls.jp/sparql';
        dbclsSparql.setEndpoint( endpoint );
```

```
// Chromosome1, 範囲 1200000 - 1300000 にある情報を取得。(重なれば OK)
var objects = dbclsFaldo.getRegionsInRange( 'chr1', 1200000, 1300000, false );

// テーブル追加
dbclsSparql.addTableFromObjects( $( '#result_box' ), 'demo', objects );
}

// 読み込み時関数呼び出し
$( function() {
    ready();
})
</script>
</head>
<body>
<h1>JS ライブラリプログラム例</h1>

<!-- テーブル表示領域 -->
<div id="result_box"></div>
</body>
</html>
```

上記の HTML ファイルにより下記の様な画面が表示される。

JS ライブラリプログラム例

name	label	chromosome	type	begin	end
A549H3K27ac_peak_30	A549H3K27ac_peak_30	chr1	forward	1208248	1208835
A549H3K27ac_peak_31	A549H3K27ac_peak_31	chr1	forward	1209396	1210028
A549H3K27ac_peak_32	A549H3K27ac_peak_32	chr1	forward	1240248	1241893
A549H3K27ac_peak_33	A549H3K27ac_peak_33	chr1	forward	1243450	1243858
A549H3K27ac_peak_34	A549H3K27ac_peak_34	chr1	forward	1244065	1244605
A549H3K27ac_peak_35	A549H3K27ac_peak_35	chr1	forward	1259636	1260027
A549H3K27ac_peak_36	A549H3K27ac_peak_36	chr1	forward	1260251	1260600
A549H3K27ac_peak_37	A549H3K27ac_peak_37	chr1	forward	1279459	1280220
A549H3K27ac_peak_38	A549H3K27ac_peak_38	chr1	forward	1280440	1281683
A549H3K27ac_peak_39	A549H3K27ac_peak_39	chr1	forward	1282944	1284060
A549H3K27ac_peak_40	A549H3K27ac_peak_40	chr1	forward	1284223	1284490
A549H3K27ac_peak_41	A549H3K27ac_peak_41	chr1	forward	1285157	1285357
A549H3K4me1_peak_28	A549H3K4me1_peak_28	chr1	forward	1207819	1208033
A549H3K4me1_peak_29	A549H3K4me1_peak_29	chr1	forward	1217339	1217654
A549H3K4me1_peak_30	A549H3K4me1_peak_30	chr1	forward	1240033	1240606
A549H3K4me1_peak_31	A549H3K4me1_peak_31	chr1	forward	1240710	1241124
A549H3K4me1_peak_32	A549H3K4me1_peak_32	chr1	forward	1241361	1242237
A549H3K4me1_peak_33	A549H3K4me1_peak_33	chr1	forward	1244925	1245111
A549H3K4me1_peak_34	A549H3K4me1_peak_34	chr1	forward	1253357	1253755
A549H3K4me1_peak_35	A549H3K4me1_peak_35	chr1	forward	1253875	1254121
A549H3K4me1_peak_36	A549H3K4me1_peak_36	chr1	forward	1258492	1258678
A549H3K4me1_peak_37	A549H3K4me1_peak_37	chr1	forward	1278889	1279177
A549H3K4me1_peak_38	A549H3K4me1_peak_38	chr1	forward	1279243	1280129
A549H3K4me1_peak_39	A549H3K4me1_peak_39	chr1	forward	1280247	1280900
A549H3K4me1_peak_40	A549H3K4me1_peak_40	chr1	forward	1280998	1282045
A549H3K4me1_peak_41	A549H3K4me1_peak_41	chr1	forward	1282702	1282930
A549H3K4me1_peak_42	A549H3K4me1_peak_42	chr1	forward	1288624	1288930

図 1 JavaScript ライブラリ使用ページ画面例

6. SPARQL 検索結果を表示する為の Stanza の開発

ここでは、DBCLS が提供する TogoStanza を用いて開発した、SPARQL 検索機能について記述する。

6.1. 機能一覧

ここで提供している機能には以下のものが存在する。

表 3 Stanza を用いた SPARQL 機能一覧

名前	説明
allele	指定位置の Allele を表示する。
beacon	指定 Allele が存在するかを判定する。
snv	指定区間の SNV 一覧を取得する。

6.2. 指定位置アレル表示 (allele)

本機能はユーザーが指定した位置のアレルを表示する。

入出力は以下の通り。

入力

表 4 allele – Stanza パラメーター一覧

パラメータ名	説明
sample	サンプル名。(e.g., LC2/ad, TSE000086)
chromosome	染色体番号。(e.g., 1, 2, X, Y)
position	位置。(e.g., 10234)

出力

指定位置の Reference Allele および Alternative Allele。

使用方法

まずは、Stanza を利用可能にする為の設定として <head> タグの中に以下のコードを埋め込む。

```
<link rel="import" href="http://humanrdf.dbcls.jp/togostanza2/allele/">
```

そして <body>タグの結果を表示したい部分に 以下のコードを埋め込む。

```
<togostanza-allele sample="(サンプル名)" chromosome="(染色体番号)" position="(位置)"></togostanza-allele>
```

簡単な例としては以下の様なコードになる。

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <title>Stanza (Allele) Demo</title>
    <meta charset="utf-8">

    <!-- インポート -->
    <link rel="import" href="http://humanrdf.dbcls.jp/togostanza2/allele/">
  </head>
  <body>
    <h1>Stanza (Allele) 埋め込み例</h1>

    <!-- Stanza -->
    <togostanza-allele sample="LC2/ad" chromosome="1" position="10234"></togostanza-allele>
  </body>
</html>
```

上記のコードでは以下の様な画面が表示される。

Stanza (Allele) 埋め込み例

Sample=LC2/ad, Chromosome=1, Position=10234	
Reference Allele	Alternative Allele
C	T

図 2 Stanza (Allele) 表示例

6.3. 指定アレル存在判定 (beacon)

本機能はユーザーが指定したアレル(Alternative Allele)が 存在するか否かを判定する。
入出力は以下の通り。

入力

表 4 allele – Stanza パラメーター一覧

パラメータ名	説明
sample	サンプル名。(e.g., LC2/ad, TSE000086)
chromosome	染色体番号。(e.g., 1, 2, X, Y)
position	位置。(e.g., 10234)
allele	Alternative Allele

出力

Existing or Not Existing

指定した Alternative Allele が存在すれば true

使用方法

まずは、Stanza を利用可能にする為の設定として <head> タグの中に以下のコードを埋め込む。

```
<link rel="import" href="http://humanrdf.dbcls.jp/togostanza2/beacon/">
```

そして <body>タグの結果を表示したい部分に 以下のコードを埋め込む。

```
<togostanza-beacon sample="(サンプル名)" chromosome="(染色体番号)" position="(位置)"  
allele="(Alternative Allele)"></togostanza-beacon>
```

簡単な例としては以下の様なコードになる。

```
<!DOCTYPE html>  
<html lang="ja">  
  <head>  
    <title>Stanza (Beacon) Demo</title>  
    <meta charset="utf-8">  
  
    <!-- インポート -->  
    <link rel="import" href="http://humanrdf.dbcls.jp/togostanza2/beacon/">  
  </head>  
  <body>  
    <h1>Stanza (Beacon) 埋め込み例</h1>
```

```

<!-- Stanza -->
<togostanza-beacon sample="LC2/ad" chromosome="1" position="10234" allele="T"></togostanza-
beacon>
</body>
</html>

```

上記のコードでは以下の様な画面が表示される。

Stanza (Beacon) 埋め込み例

Sample=LC2/ad, Chromosome=1, Position=10234, Alternative Allele=T

Result

Existing

図 3 Stanza (Beacon) 表示例

6.4. 指定区間 SNV 一覧表示 (snv)

本機能はユーザーが指定した範囲の SNV 一覧を表示する。

入出力は以下の通り。

入力

表 4 allele – Stanza パラメーター一覧

パラメータ名	説明
sample	サンプル名。(e.g., LC2/ad, TSE000086)
chromosome	染色体番号。(e.g., 1, 2, X, Y)
range_start	指定範囲、開始位置。(e.g., 10000)
range_end	指定範囲、終了位置。(e.g., 11000)

出力

指定範囲に存在する SNV の個数、
および各々の SNV の位置、Reference Allele, Alternative Allele

使用方法

まずは、Stanza を利用可能にする為の設定として <head> タグの中に以下のコードを埋め込む。

```
<link rel="import" href="http://humanrdf.dbcls.jp/togostanza2/snv/">
```

そして <body>タグの結果を表示したい部分に 以下のコードを埋め込む。

```
<togostanza-snv sample="(サンプル名)" chromosome="(染色体番号)" range_start="(指定範囲開始位置)" range_end="(指定範囲終了位置)"></togostanza-snv>
```

簡単な例としては以下の様なコードになる。

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <title>Stanza (SNV) Demo</title>
    <meta charset="utf-8">

    <!-- インポート -->
    <link rel="import" href="http://humanrdf.dbcls.jp/togostanza2/snv/">
  </head>
  <body>
    <h1>Stanza (SNV) 埋め込み例</h1>

    <!-- Stanza -->
    <togostanza-snv sample="LC2/ad" chromosome="1" range_start="10000"
range_end="11000"></togostanza-snv>
  </body>
</html>
```

上記のコードでは以下の様な画面が表示される。

Stanza (SNV) 埋め込み例

Sample=LC2/ad, Chromosome=1, Range=10000-11000

2

Position	Reference Allele	Alternative Allele
10234	C	T
10235	T	A

図 4 Stanza (SNV) 表示例