Swiss Institute of
Bioinformatics

A very brief introduction to RDF and SPARQL

Marco Pagni *et al.*

6 June 2024

Online

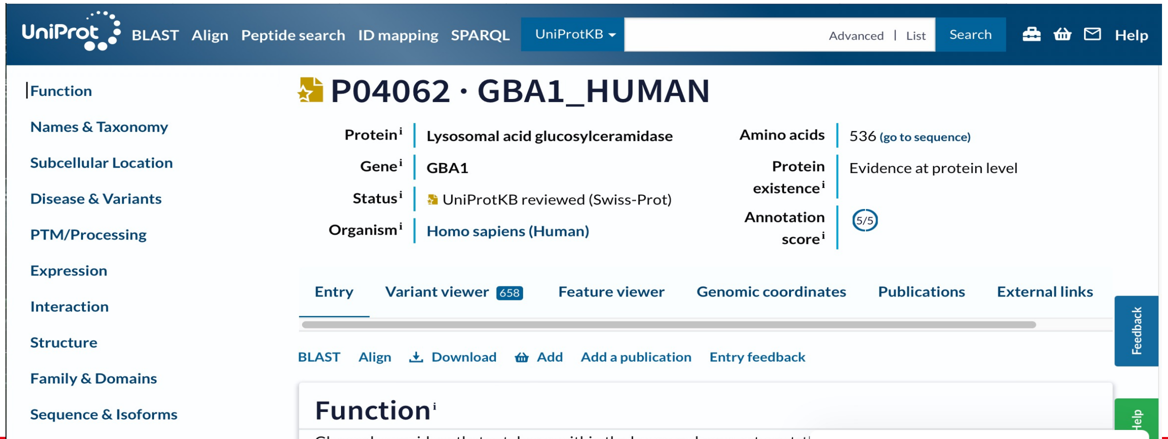
IRI - Internationalized Resource Identifier

- In the RDF world, IRIs are used as “names”, or an equivalent of “IDs”, for graph nodes.
- IRI often looks like URL, and indeed can often be used such as (this is convenient, but not mandatory)
- For example:

`http://purl.uniprot.org/uniprot/P04062`

is the legacy IRI of GBA1_HUMAN. When search in a browser, the UniprotProt server redirect it to

`https://www.uniprot.org/uniprotkb/P04062/entry`



P04062 · GBA1_HUMAN

Protein	Lysosomal acid glucosylceramidase	Amino acids	536 (go to sequence)
Gene	GBA1	Protein existence	Evidence at protein level
Status	UniProtKB reviewed (Swiss-Prot)	Annotation score	5/5
Organism	Homo sapiens (Human)		

Function

Long and short forms of IRIs

In the Turtle serialization of RDF, IRI must be "quoted" using **`<>`**:

```
<http://purl.uniprot.org/uniprot/P04062>
```

which is known as **long-form** syntax of IRIs.

By using a prefix definition, one can rewite IRI in a **short-form** notation:

```
@prefix up: <http://purl.uniprot.org/uniprot/> .
up:P04062
```

Very important for RDF:

- The long form is the reference one. It is the only form that matters for data exchanges.
- The short form is human friendly, but
  - the prefix declaration is local to the file or client software (*i.e.* it is not publicly defined).
  - different short-form identifiers may actually refer to the same long form identifier.

PREFIXES and vocabularies

Prefix definitions are local, but there existst some generally accepted conventions for widely use vocabularies

short	long
rdf:	`http://www.w3.org/1999/02/22-rdf-syntax-ns#`
rdfs:	`http://www.w3.org/2000/01/rdf-schema#`
owl	`http://www.w3.org/2002/07/owl#`
skos:	`http://www.w3.org/2004/02/skos/core#`
foaf:	`http://xmlns.com/foaf/0.1/`

RDF Triple

The simplest possible RDF graph is made of a single triple, for example in Turtle syntax:

```
<http://purl.uniprot.org/uniprot/P04062>
      <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
            <http://purl.uniprot.org/uniprot/Protein>
```

That can be rewritten using short-form notations

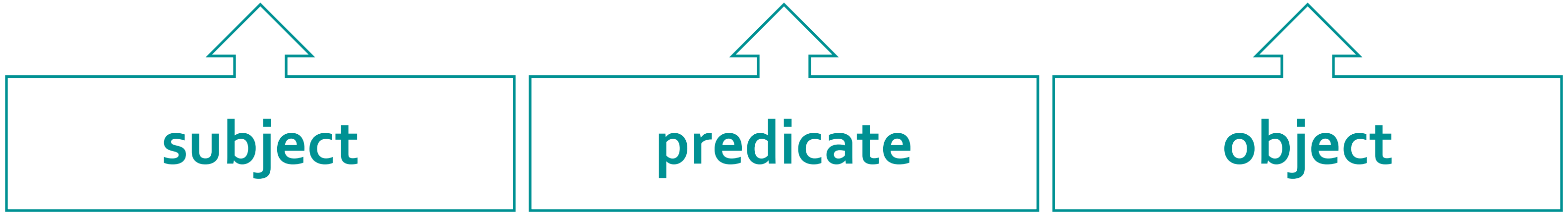
```
@prefix up:  <http://purl.uniprot.org/uniprot/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
up:P04062 rdf:type up:Protein
```

which can be further simplifies as Turtle support **a** as syntactic sugar for **`rdf:type`**

```
up:P04062 a up:Protein
```

RDF triple

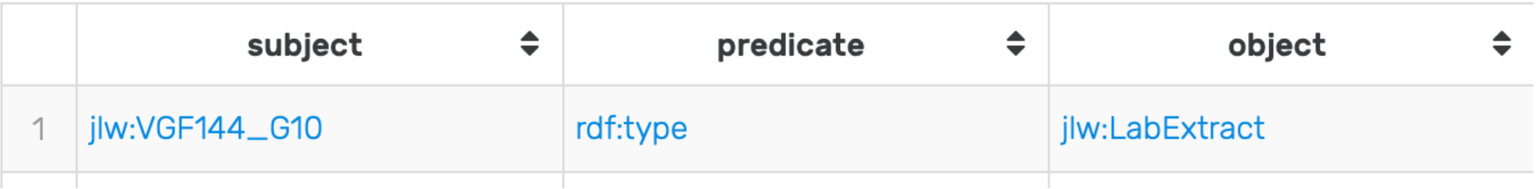
`up:P04062    rdf:type    up:Protein`



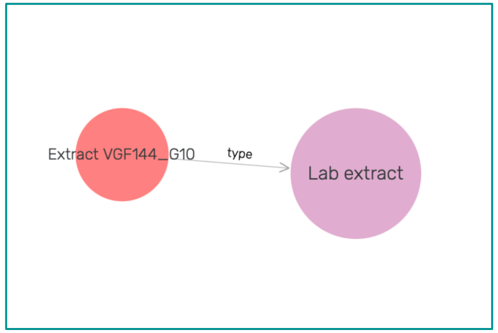
subject

predicate

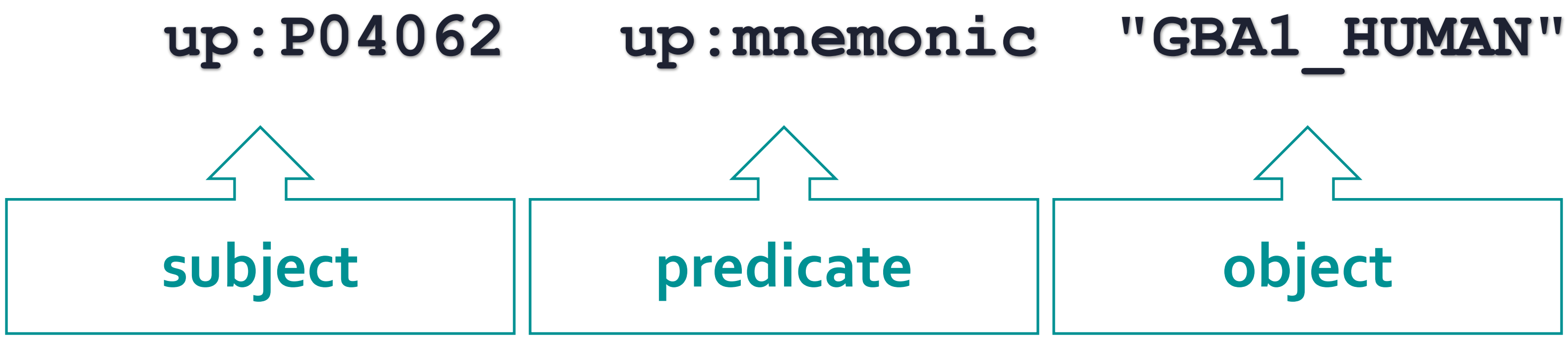
object



	subject	predicate	object
1	jlw:VGF144_G10	rdf:type	jlw:LabExtract



Literal



`up:P04062` `up:mnemonic` `"GBA1_HUMAN"`

subject

predicate

object

Literals are only permitted to occur as the object position

Literal can be optionnally typed:

`"GBA1_HUMAN"^^xsd:string` is the same as `"GBA1_HUMAN"`

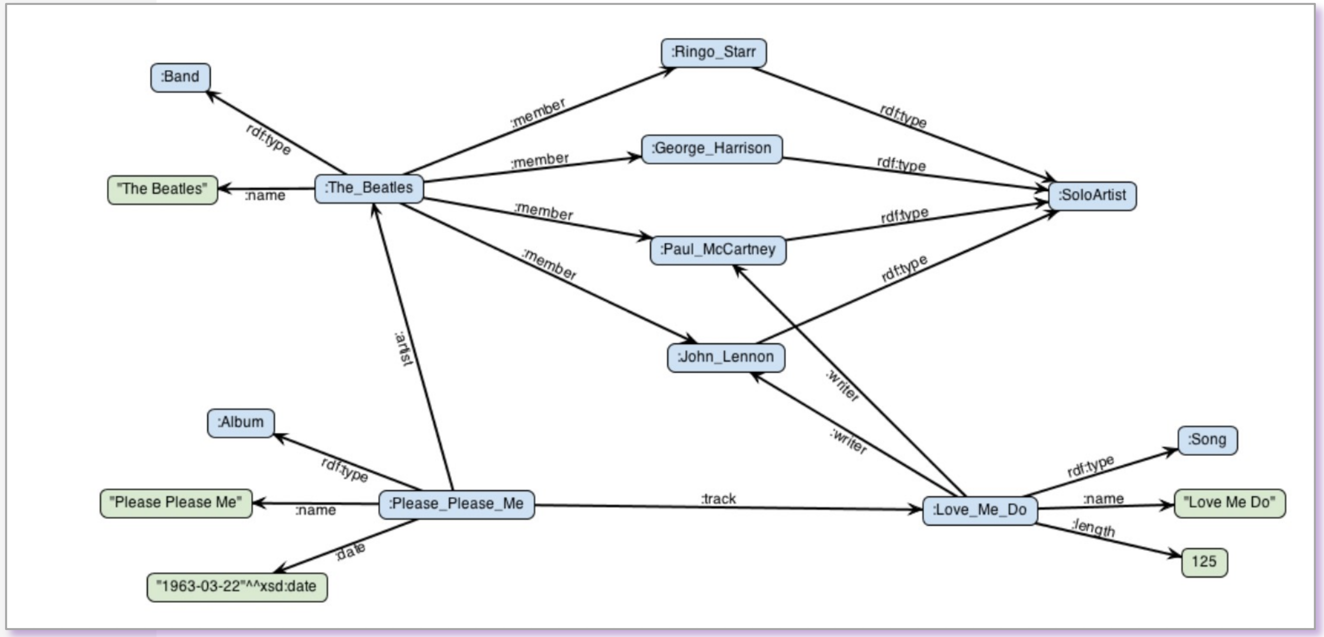
`"5"^^xsd:integer` is the same as `5`

`"2018-04-09T12:00:00"^^xsd:dateTime`

RDF graph

```
PREFIX : <http://contextualise.dev/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

:The_Beatles      rdf:type   :Band .
:The_Beatles      :name      "The Beatles" .
:The_Beatles      :member    :John_Lennon .
:The_Beatles      :member    :Paul_McCartney .
:The_Beatles      :member    :Ringo_Starr .
:The_Beatles      :member    :George_Harrison .
:John_Lennon      rdf:type   :SoloArtist .
:Paul_McCartney   rdf:type   :SoloArtist .
:Ringo_Starr      rdf:type   :SoloArtist .
:George_Harrison  rdf:type   :SoloArtist .
:Please_Please_Me rdf:type   :Album .
:Please_Please_Me :name      "Please Please Me" .
:Please_Please_Me :date      "1963-03-22"^^xsd:date .
:Please_Please_Me :artist    :The_Beatles .
:Please_Please_Me :track     :Love_Me_Do .
:Love_Me_Do       rdf:type   :Song .
:Love_Me_Do       :name      "Love Me Do" .
:Love_Me_Do       :length    125 .
:Love_Me_Do       :writer    :John_Lennon .
:Love_Me_Do       :writer    :Paul_McCartney .
```



Punctuation in Turtle syntax

```
ex:Anna a foaf:Person                .
ex:Anna foaf:knowns ex:Bob           .
ex:Bob a foaf:Person                 .
ex:Bob foaf:mBox mail:bob@gmail.com  .
ex:Bob foaf:mBox mail:bob@github.com
```

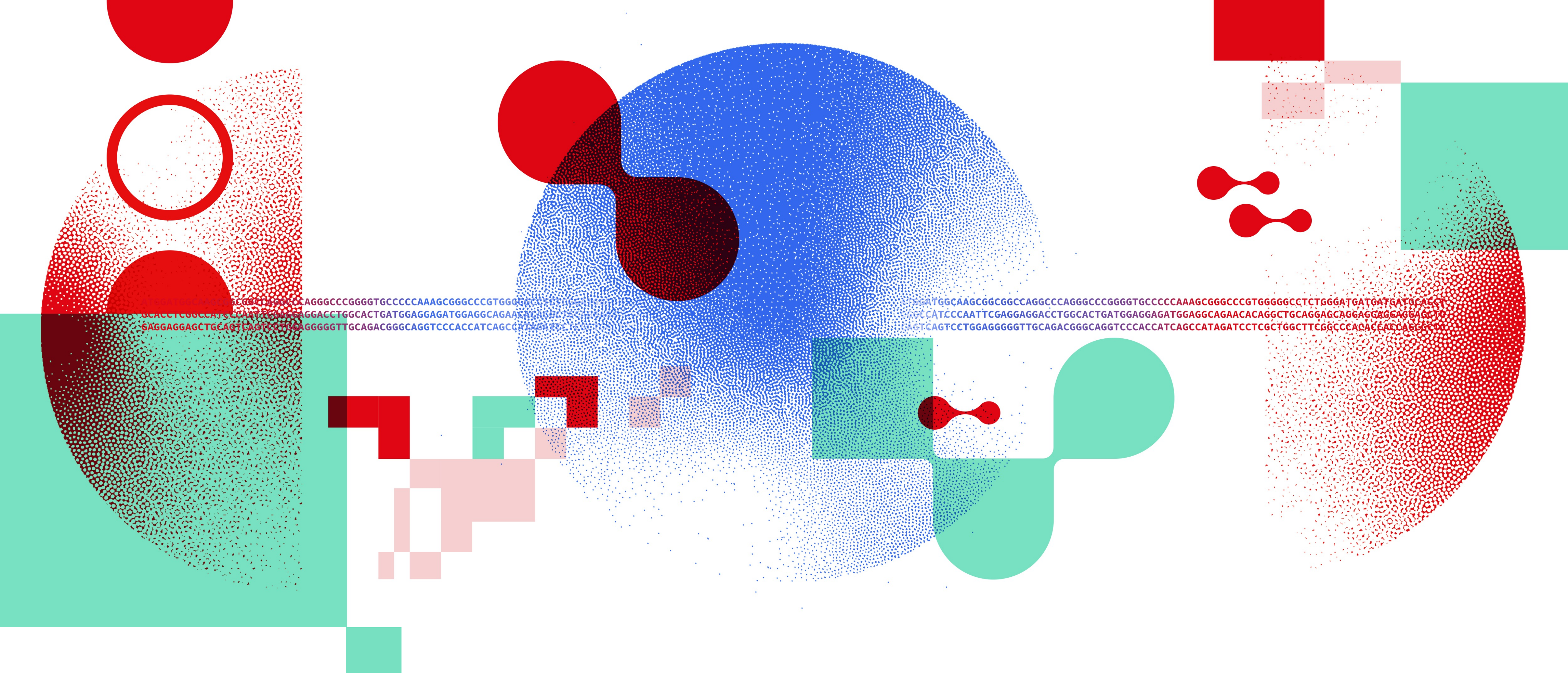
dot is the triple separator

```
ex:Anna a foaf:Person                ;
        foaf:knowns ex:Bob           .
ex:Bob a foaf:Person                 ;
       foaf:mBox mail:bob@gmail.com  ;
       foaf:mBox mail:bob@github.com
```

semicolon is a triple separator, with implicit subject

```
ex:Anna a foaf:Person                ;
        foaf:knowns ex:Bob           .
ex:Bob a foaf:Person                 ;
       foaf:mBox mail:bob@gmail.com  ,
                 mail:bob@github.com
```

comma is a triple separator, with implicit subject and object



ATGGATGGCAAGCGGCGGCCAGGCCCAGGGCCCGGGGTGCCCCCAAAGCGGGCCCGTGGGGGCCTCTGGGATGATGATGATGCACCT
GGCCATCCCAATTCGAGGAGGACCTGGCACTGATGGAGGAGATGGAGGCAGAACACAGGCTGCAGGAGCAGGAGGAGGAGCTG
CAGTCAGTCCTGGAGGGGGTTGCAGACGGGCAGGTCCCACCATCAGCCATAGATCCTCGCTGGCTTCGGCCCACACCACCAGCGCTG

elixir SWITZERLAND

SIB