

Beacon 2.0 Draft Première

Jordi Rambla *et al.*



Have you seen a...

Mutation

...like that?



YES!

Have you seen...

Calls, Variants,
Samples, Patients...

...like that?

But I'm only interested in the
ones related to a... still YES!

Disease, Gene, Tissue...

What can you tell me about it?

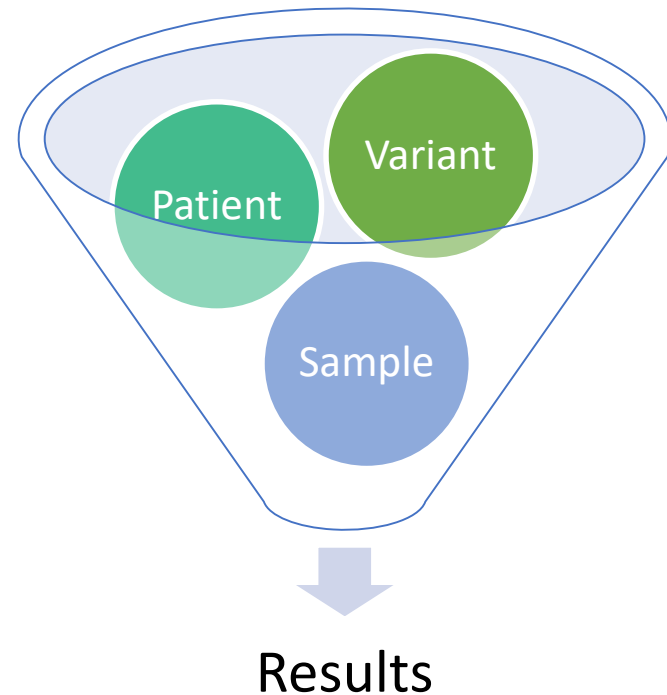
Well...

- This is the effect of the mutation
- These are the phenotypes I've seen
- This is how to get to the source data
- ...

*... but I can only provide some details
about patients if you are granted*

To consider...

- If we are having data on Variants, Patients, Samples, etc. we are not restricted to a single entry point for the query



Have you seen a... ?



- Allow for entity queries
 - Lists
 - .../**variants**
 - .../**samples**
 - .../**patients**
 - Individual entities
 - .../variants/{**id**}
 - .../samples/{**id**}
 - .../patients/{**id**}
 - Filtering
 - .../variants?chr=1&pos=1001&alt=T
 - .../patients?filter=... (*see later*)

Get linked entities ...



- Just one entity. REST approach
 - E.g. Variants of a Sample
 - `.../samples/{sampleid}/Variants`
- More complex: Variants & Patients of several samples
 - Parameter *includeDetails*
 - `.../samples?includeDetails:variants,patients`
 - `.../samples/{sampleid}?includeDetails:variants,patients`

Returned structure



- Sample1,
 - PatientA
 - Variant1, Variant2,...
 - Sample2
 - PatientE
 - Variant2, Variant3,...
 - (...)
- To consider: This distribution allows for independent schemas for each object (see later)

Returned structure



- Patient1,
 - Sample1, Sample2,...
 - Variant1, Variant2,...
 - Patient2
 - Sample3, Sample4,...
 - Variant2, Variant3,...
 - (...)
- Variant1,
 - Sample1, Sample2,...
 - Patient1
 - Variant2
 - Sample3, Sample4,...
 - Patient1, Patient2,...
 - (...)

Dealing with unstable schemas

- Declare requested schema at query time

```
"request": {  
  "Patient": "beacon-patient-v1.0",  
  "Variant": "beacon-variant-representation-v0.1",  
  "VariantAnnotation": "beacon-variant-annotation-v1.0"  
}
```

- State returned schema in the response

```
"results": [ {  
  "patient": {  
    "version": "beacon-patient-v1.0",  
    "value": {  
      "sex": "female",  
      (...)  
    }  
  }  
}
```

- Beacon to declare supported schemas
- Beacon to support a default schema for each entity

Variant schema

```
"variantsFound": [
{
  "variant": {
    "version": "beacon-variant-v1.0",
    "value": [
      {
        "variantDetails": {
          "chromosome": "",
          "referenceBases": "",
          "alternateBases": "",
          "variantType": "",
          "start": 0,
          "end": 0
        },
        "info": { }
      }
    ]
  },
  "variantAnnotations": {
```

```
"variantAnnotations": {
  "version": "beacon-variant-annotation-v1.0",
  "value": {
    "variantMetadata": {
      "version": "beacon-variant-metadata-v1.0",
      "value": [
        {
          "geneId": "",
          "HGVSId": "",
          "transcriptId": "",
          "alleleId": "",
          "variantClassification": "",
          "variantType": "",
          "disease": "",
          "proteinChange": "",
          "clinVarId": "",
          "pubmedId": "",
          "timestamp": "",
          "info": { }
        }
      ]
    },
    "info": {
      "cellBase": { },
      "dbSNP": { },
      "clinVar": { }
    }
  }
},
```

Initial schemas to consider



- Patient
 - *Beacon default
 - Phenopackets: Individual
 - SchemaBlocks: Individual
- Sample
 - *Beacon default
 - PhPck: Biosample
 - SB: Biosample
- Variant Representation
 - *Beacon default
 - Beacon “legacy”
 - GA4GH VR
 - SchemaBlocks
- Variant Annotation
 - Many...

But only if...



- Extension to *each* Beacon Query
- Leveraging ontologies
- Backend should know the mapping between ontology term (e.g. AgeOfOnset) and the “database column” (e.g. Age)
- Allowing for custom dictionaries (customFilters section)

```
// female
Filters: [
  "PATO:0000383"
]

// neoplastic sample (not study, nor individual)
customFilters: [
  "sample.provenance.id=EFO:0009656"
]

// female, age of onset, abnormal sample (~case)
Filters: [
  "PATO:0000383"
  , "HP:0011007=>49"
  , "EFO:0009656"
]
```

Transversal features



- Details per dataset
- handovers
- Authentication and /access_levels
- /filtering_terms
- Endpoint aliases to allow simpler queries