BioXSD
BioJSON BioYAML

Towards unified formats for sequences, alignments, features, and annotations

Matúš Kalaš[1], Sveinung Gundersen[2], László Kaján[3], Hervé Ménager[4], Jon Ison[5], Christophe Blanchet[6], Steve Pettifer[7], Rodrigo Lopez[8], Kristoffer Rapacki[5], Inge Jonassen[1], and open for contributions

[1]Computational Biology Unit, Department of Informatics, University of Bergen, Bergen, Norway; [2]Institute for Cancer Research, Oslo University Hospital and Department of Informatics, University of Oslo, Oslo, Norway; [3]unaffiliated, previously Bioinformatics and Computational Biology Department, Technische Universität München, Garching, Germany; [4]Institut Pasteur, Paris, France; [5]Department of Systems Biology, Technical University of Denmark, Kongens Lyngby, Denmark; [6]L'Institut Français de Bioinformatique, Gif-sur-Yvette, France; [7]School of Computer Science, University of Manchester, Manchester, UK; [8]European Bioinformatics Institute, EMBL, Wellcome Trust Genome Campus, Hinxton, Cambridge, UK.

2016

/bioxsd/bioxsd **@BioXSD** **http://groups.google.com/group/bioxsd** **http://bioxsd.org** **support@bioxsd.org**

Latest stable release: http://bioxsd.org/BioXSD-1.1.xsd

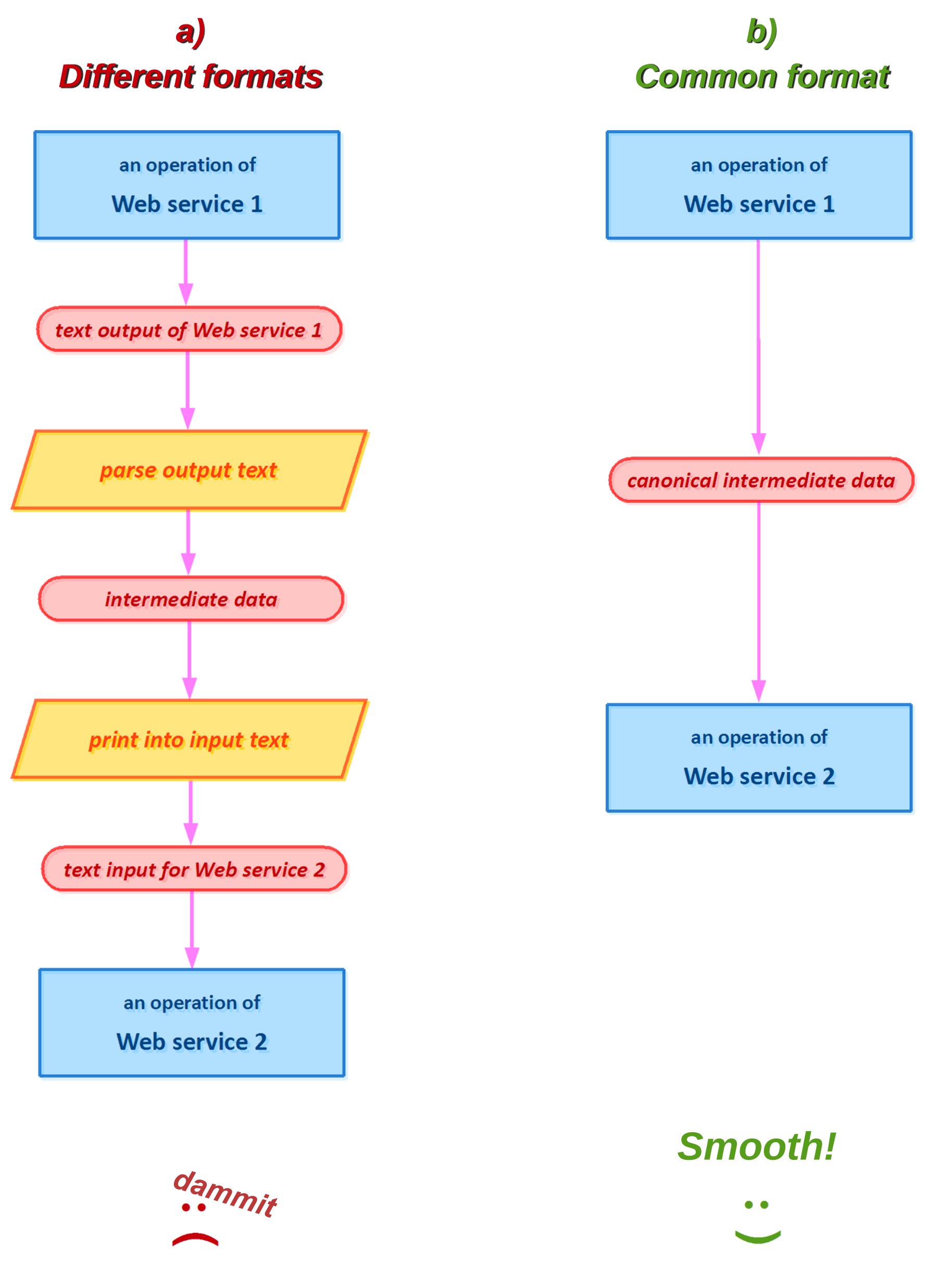
MOTIVATION

Without a common format, using diverse tools in a workflow demands conversions, "shims", or do-it-yourself parsing. And worst of all, **maintaining** these in the future.

The 2 scenarios show demands for connecting 2 tools (e.g. Web services) that use:

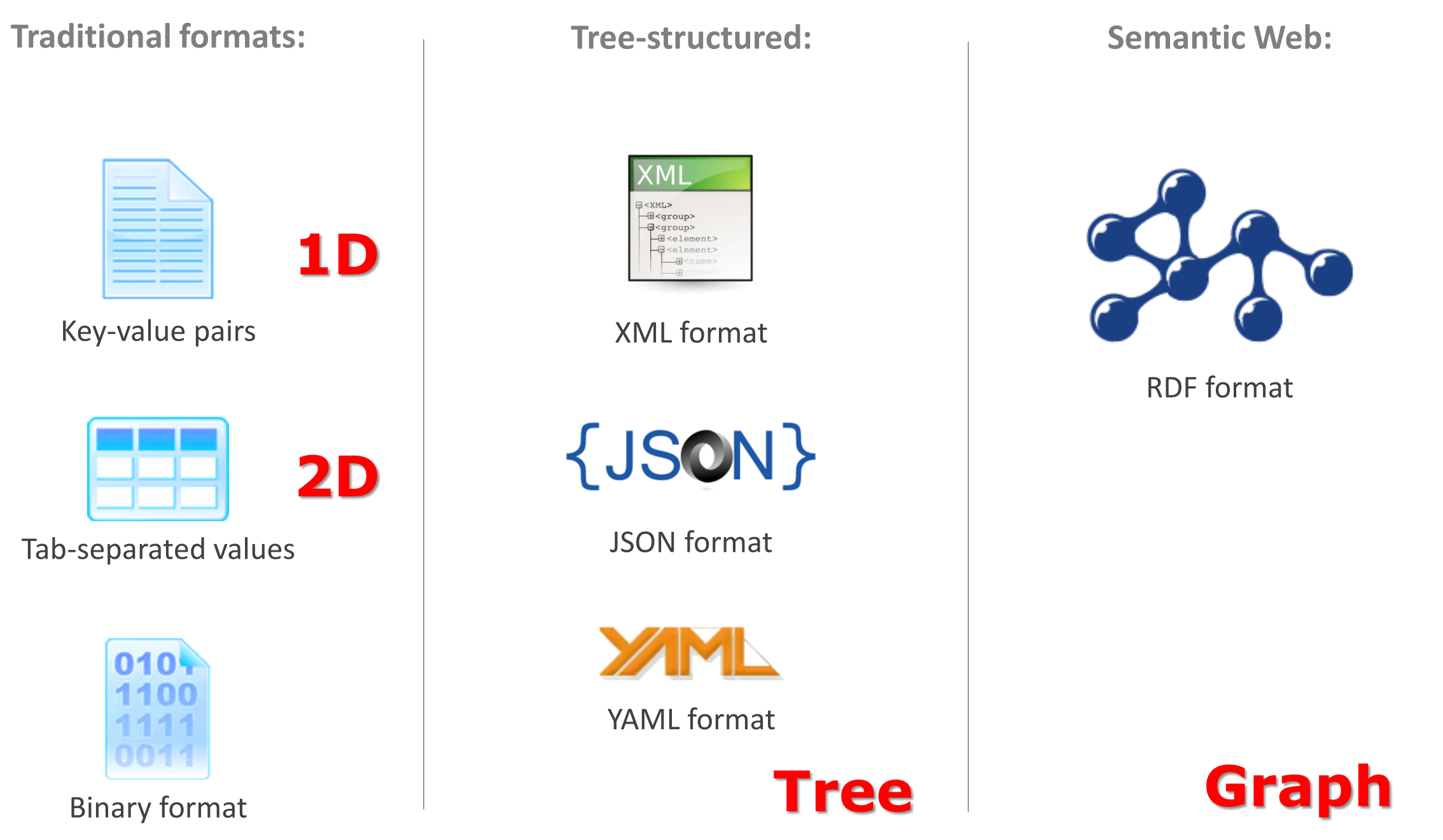
a) Different formats

b) A common format


a)
Different formats
an operation of Web service 1
text output of Web service 1
parse output text
intermediate data
print into input text
text input for Web service 2
an operation of Web service 2
dammit
b)
Common format
an operation of Web service 1
canonical intermediate data
an operation of Web service 2
Smooth!

TECHNOLOGY CHOICES

Different paradigms of data formatting represent data differently.



A machine-understandable definition of a specific format (a data model, a schema) is highly beneficial for validation and maintainability.

GTrack format TSV with column definitions http://gtrack.no	**XML Schema (XSD) 1.0** **XML Schema 1.1** **Relax NG** **JSON Schema** **...**	**OWL** **...**

SIMPLE EXAMPLE: BioXSD sequence record

Example data instance, BioXSD in XML:

```
<mySequenceRecord
        xsi:type="bx:GeneralAminoacidSequenceRecord"
        xmlns:bx="http://bioxsd.org/BioXSD-1.1"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://bioxsd.org/BioXSD-1.1 http://bioxsd.org/BioXSD-1.1.xsd"
>
    <bx:sequence>MDPLGDTLRRLREAFHAGRTRPAEFRAAQLQGLGRFLQENKQLLHDAL</bx:sequence>
    <bx:species
        dbName="NCBI Taxonomy"
        accession="9606"
        entryUri="http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=9606"
        speciesName="Human"
    />
    <bx:reference
        dbName="UniProt"
        accession="P43353"
        entryUri="http://www.uniprot.org/uniprot/P43353"
        sequenceVersion="1"
        variantAccession="P43353-1"
    >
        <bx:subsequencePosition>
            <bx:segment min="1" max="48"/>
        </bx:subsequencePosition>
    </bx:reference>
    <bx:name>Aldehyde dehydrogenase family 3 member B1 (ALDH3B1), N-terminus</bx:name>
</mySequenceRecord>
```

In BioJSON:

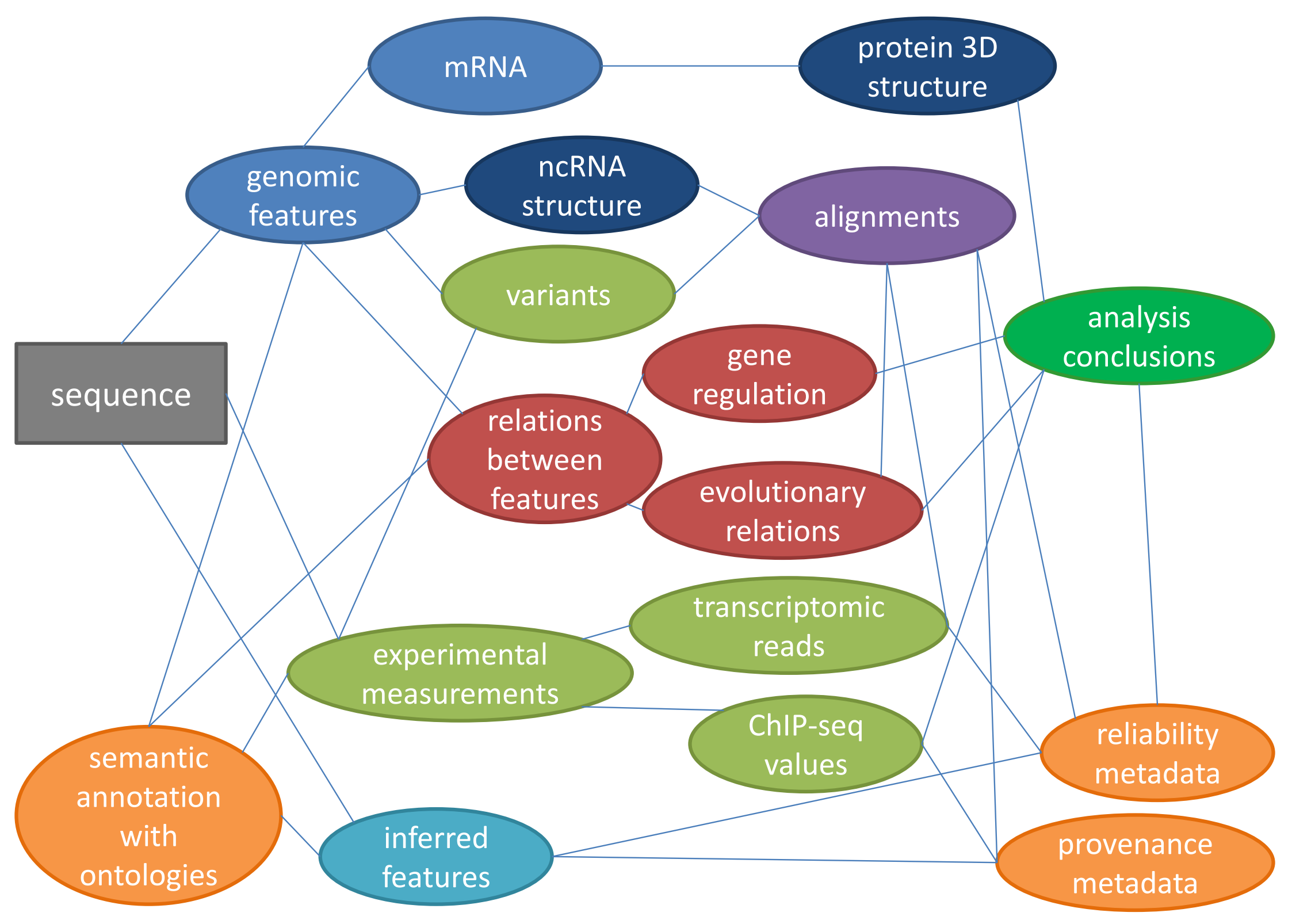
```
{
    "sequence": "MDPLGDTLRRLREAFHAGRTRPAEFRAAQLQGLGRFLQENKQLLHDAL",
    "species": {
        "dbName": "NCBI Taxonomy",
        "accession": "9606",
        "entryUri": "http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=9606",
        "speciesName": "Human"
    },
    "reference": {
        "dbName": "UniProt",
        "accession": "P43353",
        "entryUri": "http://www.uniprot.org/uniprot/P43353",
        "sequenceVersion": "1",
        "variantAccession": "P43353-1",
        "subsequencePosition": {
            "segment": {
                "min": 1,
                "max": 48
            }
        }
    },
    "name": "Aldehyde dehydrogenase family 3 member B1 (ALDH3B1), N-terminus"
}
```

In BioYAML:

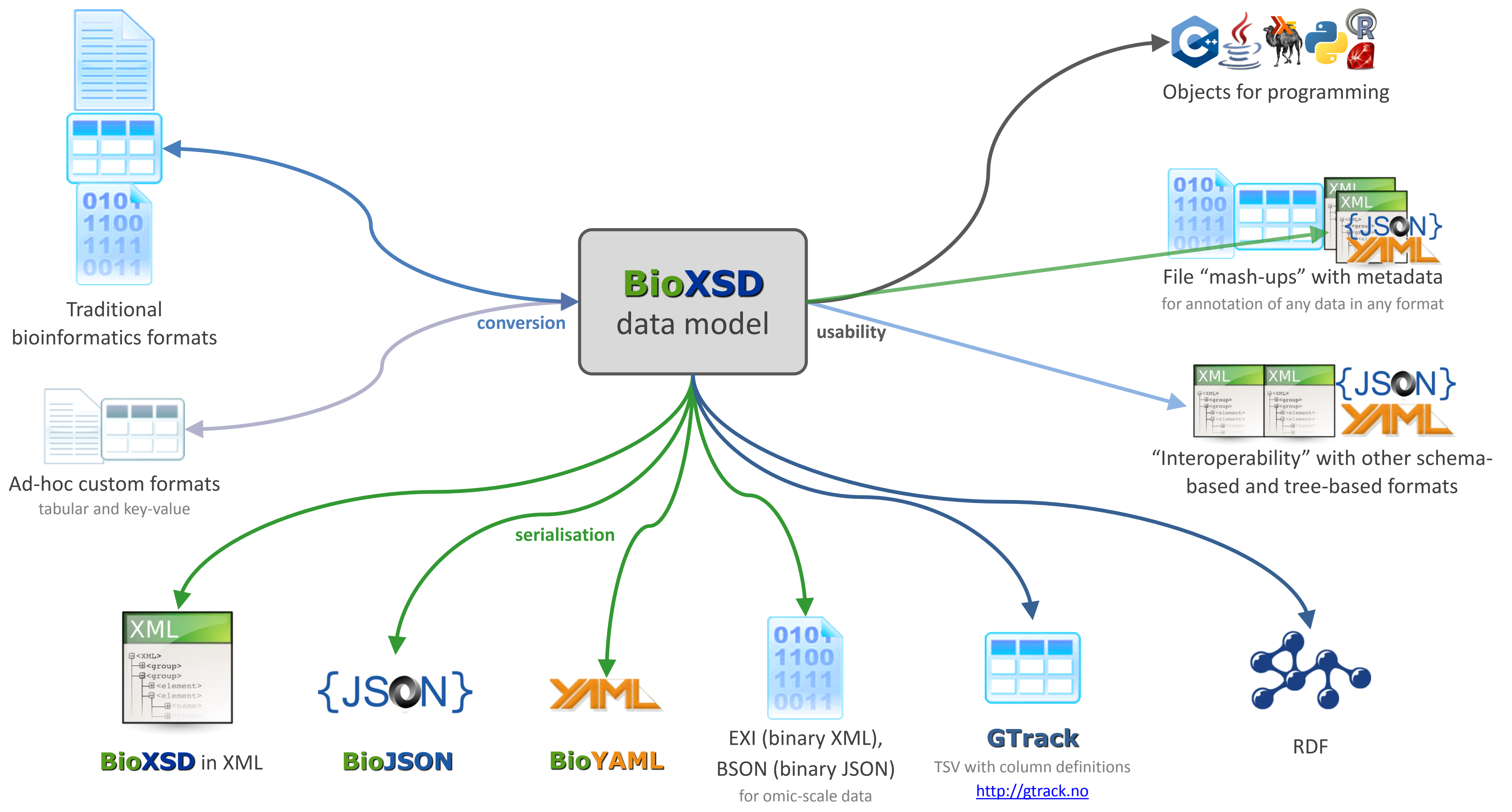
```
---
sequence: MDPLGDTLRRLREAFHAGRTRPAEFRAAQLQGLGRFLQENKQLLHDAL
species:
  dbName: NCBI Taxonomy
  accession: "9606"
  entryUri: http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=9606
  speciesName: Human
reference:
  dbName: UniProt
  accession: P43353
  entryUri: http://www.uniprot.org/uniprot/P43353
  sequenceVersion: "1"
  variantAccession: P43353-1
  subsequencePosition:
    segment:
      min: 1
      max: 48
name: Aldehyde dehydrogenase family 3 member B1 (ALDH3B1), N-terminus
```

COMPLEX EXAMPLE: BioXSD feature record

BioXSD can represent diverse interconnected features of a sequence, together with related data and metadata, in an integrated data record.



ONGOING DEVELOPMENTS: single data model with multiple choices of exchange formats and conversions


Traditional bioinformatics formats
Ad-hoc custom formats
tabular and key-value
conversion
BioXSD data model
usability
Objects for programming
File "mash-ups" with metadata
for annotation of any data in any format
"Interoperability" with other schema-based and tree-based formats
serialisation
BioXSD in XML
BioJSON
BioYAML
EXI (binary XML), BSON (binary JSON)
for omic-scale data
GTrack
TSV with column definitions
http://gtrack.no
RDF

BioXSD has been developed as a tree-based data model and an exchange format for basic bioinformatics data, centred around a bio-polymer sequence. BioXSD allows integration of diverse features, information, measurements, and inferred values about a biological molecule or its part, annotated with provenance and reliability metadata, ontology concepts, scientific remarks, and conclusions.

BioJSON and BioYAML are the ongoing developments. These exchange formats are based on the same data model as BioXSD, but providing serialisations in JSON and YAML respectively. BioJSON and BioYAML thus enrich the BioXSD family with alternatives to the original XML.

As tree-based data formats, BioXSD, BioJSON, and BioYAML are particularly suitable for programming in object-oriented languages, and for use with web applications and web APIs (Web services), while at the same time allowing a reasonable level of human readability.

BioXSD|BioJSON|BioYAML are developed together with GTrack (the universal tabular format for sequence features), by ELIXIR Norway and an international community of collaborators (http://bioxsd.org/#Contact). The BioXSD|GTrack family is going to support smooth interoperability between these alternative, universal formats, and between the tools that consume or provide them as inputs or outputs.