JSON-LD 1.1

January 16, 2020

Disclaimer:

This slide deck (© 2020 Johns Hopkins University) is an early draft of what will become two or more tutorials on the use of JSON-LD 1.1 in FHIR and other contexts. It represents work that is actively underway and, at the moment, is a partially complete checkpoint.

RDF - the “universal solvent” of data

Any data structure can be converted into its RDF equivalent

There *is* an RDF in your future…

… you can either use the tools, query languages, representations, etc. that exist today or…

… you can build anew.

We (strongly) represent considering the former…

Links

https://www.w3.org/TR/rdf11-primer/ - Into to RDF

https://json-ld.org/ - All things json-ld

https://json-ld.org/playground/ - Javascript JSON-LD Implementation

JSON and RDF

Javascript Object Notation (JSON)

https://www.json.org/json-en.html

JSON is built on two structures:

A collection of name/value pairs. In various languages, this is realized as an *object*, record, struct, dictionary, hash table, keyed list, or associative array.
An ordered list of values. In most languages, this is realized as an *array*, vector, list, or sequence.

Outline

Part 1: JSON-LD as an RDF Syntax

- Getting from JSON to RDF (aka. JSON-LD)
- Getting from RDF (turtle, xml, n3, …) into JSON-LD

Part 2: Transforming JSON-LD in to something usable

- JSON-LD context

Outline

Part 1: JSON-LD as an RDF Syntax

- **Getting from JSON to RDF (aka. JSON-LD)**
- Getting from RDF (turtle, xml, n3, …) into JSON-LD

Part 2: Transforming JSON-LD in to something usable

- JSON-LD context

JSON to RDF

Start with arbitrary JSON structure (link):

```
{ “id”: “1173418”,
  “active” : true,
  “name” : {
      “first” : “Seymour”,
      “middle” : “P”,
      “last”: “Snodgrass”
   },
   “birthdate” : “2002-07-21”,
   “children” : [“Tom”, “Dick”, “Harry”]
}
```

JSON and RDF

An arbitrary JSON structure (link):

Subjects

```
{ “id”: “1173418”,
  “active” : true,
  “name” : {
      “first” : “Seymour”,
      “middle” : “P”,
      “last”: “Snodgrass”
   },
   “birthdate” : “2002-07-21”,
   “children” : [“Tom”, “Dick”, “Harry”]
}
```

JSON and RDF

An arbitrary JSON structure (link):

```
{ “id”: “1173418”,
  “active” : true,
  “name” : {
      “first” : “Seymour”,
      “middle” : “P”,
      “last”: “Snodgrass”
  },
  “birthdate” : “2002-07-21”,
  “children” : [“Tom”, “Dick”, “Harry”]
}
```

Predicates

...

JSON and RDF

An arbitrary JSON structure (link):

```
{ “id”: “1173418”,
  “active” : true,
  “name” : {
      “first” : “Seymour”,
      “middle” : “P”,
      “last”: “Snodgrass”
   },
   “birthdate” : “2002-07-21”,
   “children” : [“Tom”, “Dick”, “Harry”]
}
```

Objects

...

JSON to RDF Conversion Steps

1. Convert predicates to URI’s - identify or, as a last resort, create the URI that represents the semantics of the corresponding JSON key

Predicates to URI's (This <u>is</u> the *hard* step)

This is the core of Linked Open Data — what do I (we) mean when I (we) say:

- "id"
- "active"
- "name"
- "name.first"
- "name.middle"
- "name.last"
- "birthdate"
- "children"

in *this* context?

Predicates to URIs (cheating for the moment)

```
{ "http://example.org/person/id": "1173418",
  "http://example.org/person/active" : true,
  "http://example.org/person/name" : {
      "http://example.org/person/first" : "Seymour",
      "http://example.org/person/middle" : "P",
      "http://example.org/person/last": "Snodgrass"
   },
   "http://example.org/person/birthdate" : "2002-07-21",
   "http://example.org/person/children" : ["Tom", "Dick",
"Harry"]
}
```

***Compacted* Format**

JSON-LD as RDF Triples

JSON Objects identified by Blank Nodes

JSON-LD Input | Options

_:b0

_:b1

```
{
  "http://example.org/person/id": "1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/name": {
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/middle": "P",
    "http://example.org/person/last": "Snodgrass"
  },
  "http://example.org/person/birthdate": "2002-07-21",
  "http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visuali

```
_:b0 <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
_:b0 <http://example.org/person/birthdate> "2002-07-21" .
_:b0 <http://example.org/person/children> "Dick" .
_:b0 <http://example.org/person/children> "Harry" .
_:b0 <http://example.org/person/children> "Tom" .
_:b0 <http://example.org/person/id> "1173418" .
_:b0 <http://example.org/person/name> _:b1 .
_:b1 <http://example.org/person/first> "Seymour" .
_:b1 <http://example.org/person/last> "Snodgrass" .
_:b1 <http://example.org/person/middle> "P" .
```

http://tinyurl.com/ygrvyzxg

JSON-LD as RDF Triples

Built-in JSON type

JSON-LD Input | Options

```
{
  "http://example.org/person/id": "1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/name": {
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/middle": "P",
    "http://example.org/person/last": "Snodgrass"
  },
  "http://example.org/person/birthdate": "2002-07-21",
  "http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visuali

```
_:b0 <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
_:b0 <http://example.org/person/birthdate> "2002-07-21" .
_:b0 <http://example.org/person/children> "Dick" .
_:b0 <http://example.org/person/children> "Harry" .
_:b0 <http://example.org/person/children> "Tom" .
_:b0 <http://example.org/person/id> "1173418" .
_:b0 <http://example.org/person/name> _:b1 .
_:b1 <http://example.org/person/first> "Seymour" .
_:b1 <http://example.org/person/last> "Snodgrass" .
_:b1 <http://example.org/person/middle> "P" .
```

http://tinyurl.com/ygrvyzxg

JSON-LD as RDF Triples

List order *not* preserved (!)

JSON-LD Input | Options

```
{
  "http://example.org/person/id": "1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/name": {
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/middle": "P",
    "http://example.org/person/last": "Snodgrass"
  },
  "http://example.org/person/birthdate": "2002-07-21",
  "http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visuali

```
_:b0 <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
_:b0 <http://example.org/person/birthdate> "2002-07-21" .
_:b0 <http://example.org/person/children> "Dick" .
_:b0 <http://example.org/person/children> "Harry" .
_:b0 <http://example.org/person/children> "Tom" .
_:b0 <http://example.org/person/id> "1173418" .
_:b0 <http://example.org/person/name> _:b1 .
_:b1 <http://example.org/person/first> "Seymour" .
_:b1 <http://example.org/person/last> "Snodgrass" .
_:b1 <http://example.org/person/middle> "P" .
```

http://tinyurl.com/ygrvyzxg

JSON to RDF Conversion Steps

1. Convert predicates to URI’s

2. Add subject URI’s where appropriate

Subject URIs

JSON-LD Input | Options

```
"@id": "http://hl7.org/patient/1173418",
"http://example.org/person/id": "1173418",
```

```
{
  "@id": "http://hl7.org/patient/1173418",
  "http://example.org/person/id": "1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/name": {
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/middle": "P",
    "http://example.org/person/last": "Snodgrass"
  },
  "http://example.org/person/birthdate": "2002-07-21",
  "http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA

```
<http://hl7.org/patient/1173418> <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
<http://hl7.org/patient/1173418> <http://example.org/person/birthdate> "2002-07-21" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Dick" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Harry" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Tom" .
<http://hl7.org/patient/1173418> <http://example.org/person/id> "1173418" .
<http://hl7.org/patient/1173418> <http://example.org/person/name> _:b0 .
_:b0 <http://example.org/person/first> "Seymour" .
_:b0 <http://example.org/person/last> "Snodgrass" .
_:b0 <http://example.org/person/middle> "P" .
```

http://tinyurl.com/yjascwhn

Subject URIs

Can be added to *any* JSON object

JSON-LD Input | Options

```
"@id": "http://hl7.org/patient/1173418/name",
```

```
{
  "@id": "http://hl7.org/patient/1173418",
  "http://example.org/person/id": "1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/name": {
    "@id": "http://hl7.org/patient/1173418/name",
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/middle": "P",
    "http://example.org/person/last": "Snodgrass"
  },
  "http://example.org/person/birthdate": "2002-07-21",
  "http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA

```
<http://hl7.org/patient/1173418/name> <http://example.org/person/first> "Seymour" .
<http://hl7.org/patient/1173418/name> <http://example.org/person/last> "Snodgrass" .
<http://hl7.org/patient/1173418/name> <http://example.org/person/middle> "P" .
<http://hl7.org/patient/1173418> <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
<http://hl7.org/patient/1173418> <http://example.org/person/birthdate> "2002-07-21" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Dick" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Harry" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Tom" .
<http://hl7.org/patient/1173418> <http://example.org/person/id> "1173418" .
<http://hl7.org/patient/1173418> <http://example.org/person/name> <http://hl7.org/patient/1173418/name> .
```

JSON to RDF Conversion Steps

1. Convert predicates to URI’s
2. Add subject URI’s where appropriate
3. Add data types, language tags and/or direction

Data types

JSON-LD Input | Options | Documen

```
{
  "@id": "http://hl7.org/patient/1173418",
  "http://example.org/person/id": "1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/name": {
    "@id": "http://hl7.org/patient/1173418/name",
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/middle": "P",
    "http://example.org/person/last": "Snodgrass"
  },
  "http://example.org/person/birthdate": {
    "@type": "http://www.w3.org/2001/XMLSchema#date",
    "@value": "2002-07-21"
  },
  "http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

```
},
"http://example.org/person/birthdate": {
  "@type": "http://www.w3.org/2001/XMLSchema#date",
  "@value": "2002-07-21"
},
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA | Signed w

```
<http://hl7.org/patient/1173418/name> <http://example.org/person/first> "Seymour" .
<http://hl7.org/patient/1173418/name> <http://example.org/person/last> "Snodgrass" .
<http://hl7.org/patient/1173418/name> <http://example.org/person/middle> "P" .
<http://hl7.org/patient/1173418> <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
<http://hl7.org/patient/1173418> <http://example.org/person/birthdate> "2002-07-21"^^<http://www.w3.org/2001/XMLSchema#date> .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Dick" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Harry" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Tom" .
<http://hl7.org/patient/1173418> <http://example.org/person/id> "1173418" .
<http://hl7.org/patient/1173418> <http://example.org/person/name> <http://hl7.org/patient/1173418/name> .
```

http://tinyurl.com/yz89vtqk

What just happened?

```
"http://example.org/person/birthdate": "2002-07-21",
```

↓

```
"http://example.org/person/birthdate": {
  "@type": "http://www.w3.org/2001/XMLSchema#date",
  "@value": "2002-07-21"
},
```

Expanded Format
All Equivalent (in JSON-LD)

```
"http://example.org/person/id": "1173418",
```

```
"http://example.org/person/id": {
  "@value": "1173418"
},
```

```
"http://example.org/person/id": [{
  "@value": "1173418"
}],
```

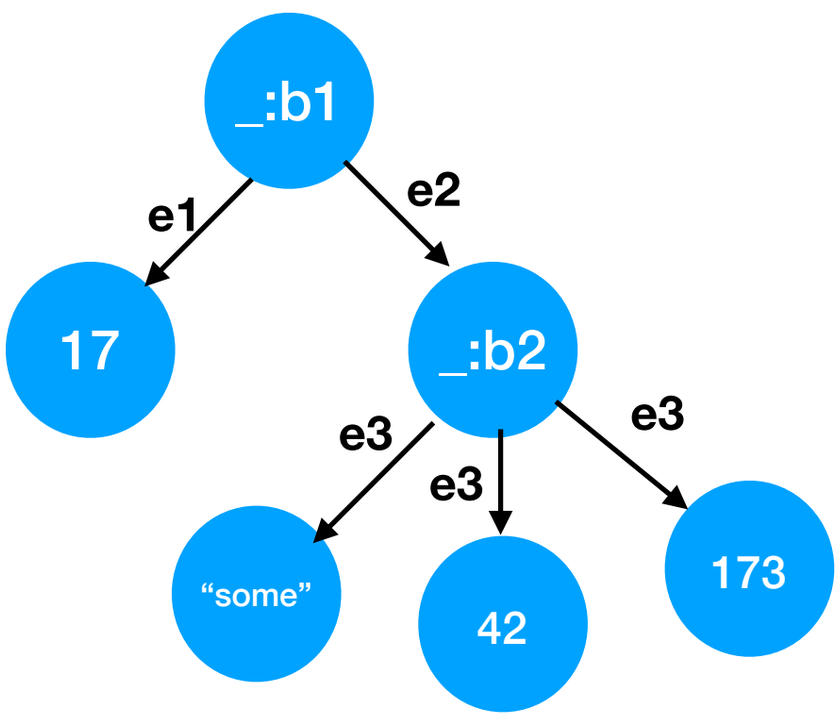
The Graph Object

“@graph”

The Graph Object

The issue: JSON *necessarily* represents a tree structure:

```
{ “e1”: 17,
  “e2”: { “e21”: [“some”, 42, 173] }
}
```

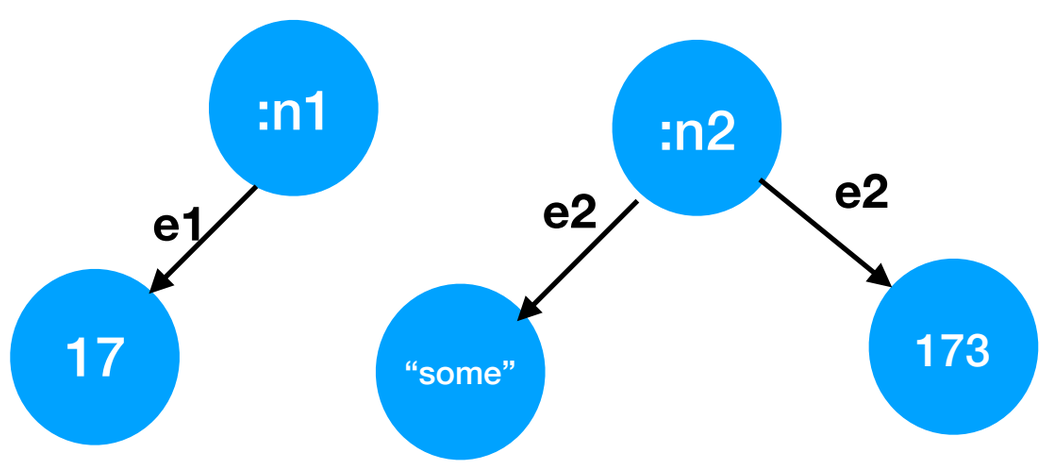


The Graph Object

RDF can represent disconnected graphs as well:

<http://ex.org/n1> <http://ex.org/e1> 17, 42 .

<http://ex.org/n2> <http://ex.org/e2> “some”, 173 .



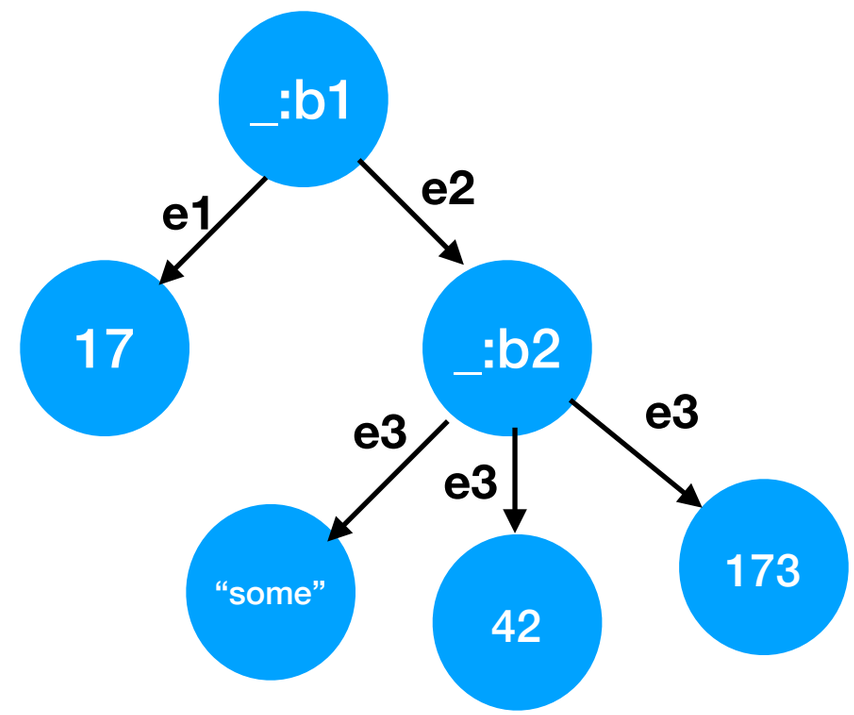
The Graph Object

“@graph” is an “anonymous” intermediate node:

```
{
  "@graph": {
    "http://ex.org/e1": 17,
    "http://ex.org/e1e2": {
      "http://e21": ["some", 42, 173]
    }
  }
}
```

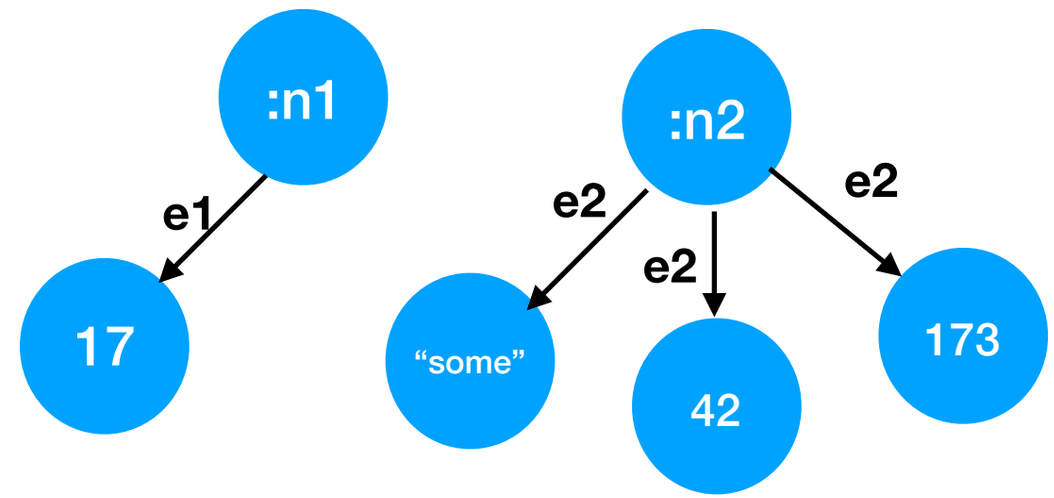
```
{
  "http://ex.org/e1": 17,
  "http://ex.org/e1e2": {
    "http://e21": ["some", 42, 173]
  }
}
```

Identical Semantics



The Graph Object

```
{
  "@graph": [
    {
      "@id": ":n1",
      "http://ex.org/e1": 17
    },
    {
      "@id": ":n2",
      "http://e2": ["some", 42, 173]
    }
  ]
}
```




```
{
    {
      "@id": ":n1",
      "http://ex.org/e1": 17
    },
    {
      "@id": ":n2",
      "http://e2": ["some", 42, 173]
    }
}
```

Not Valid JSON but the intent

The Graph Object
Named Graphs

JSON-LD Input | Options

http://tinyurl.com/rlkrpac

Document URL

```
{
  "@id": "http://example.org/mygraph",
  "@graph": [
    {
      "@id": ":n1",
      "http://ex.org/e1": 17
    },
    {
      "@id": "http://example.org/yourgraph",
      "@graph": {
        "@id": ":n2",
        "http://e2": [{"@value": "some", "@language": "en-GB"}, 42, 173]
      }
    }
  ]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA | Signed with Bitcoin

Subject	Predicate	Object	Language	Datatype	Graph
https://json-ld.org/playground/:n1	http://ex.org/e1	17		http://www.w3.org/2001/XMLSchema#integer	http://example.org/mygraph
https://json-ld.org/playground/:n2	http://e2	173		http://www.w3.org/2001/XMLSchema#integer	http://example.org/yourgraph
https://json-ld.org/playground/:n2	http://e2	42		http://www.w3.org/2001/XMLSchema#integer	http://example.org/yourgraph
https://json-ld.org/playground/:n2	http://e2	some	en-gb		http://example.org/yourgraph

JSON and RDF

Start with some JSON…


```
{
  "resourceType": "Observation",
  "id": "f003",
  "identifier": [
    {
      "use": "official",
      "system": "http://www.bmc.nl/zorgportal/identifiers
/observations",
      "value": "6325"
    }
  ],
  "status": "final",
  "code": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "11557-6",
        "display": "Carbon dioxide in blood"
      }
    ]
  },
  "subject": {
    "reference": "Patient/f001",
    "display": "P. van de Heuvel"
  },
  "valueQuantity": {
    "value": 6.2,
    "unit": "kPa",
    "system": "http://unitsofmeasure.org",
    "code": "kPa"
  }
}
```


http://tinyurl.com/tvlklb9

JSON and RDF
Convert Predicates to URIs


```
{
  "http://hl7.org/fhir/resourceType": "Observation",
  "http://hl7.org/fhir/id": "f003",
  "http://hl7.org/fhir/identifier": [
    {
      "http://hl7.org/fhir/use": "official",
      "http://hl7.org/fhir/system": "http://www.bmc.nl/zorgportal/identifiers/observations",
      "http://hl7.org/fhir/value": "6325"
    }
  ],
  "http://hl7.org/fhir/status": "final",
  "http://hl7.org/fhir/code": {
    "http://hl7.org/fhir/coding": [
      {
        "http://hl7.org/fhir/system": "http://loinc.org",
        "http://hl7.org/fhir/code": "11557-6",
        "http://hl7.org/fhir/display": "Carbon dioxide in blood"
      }
    ]
  },
  "http://hl7.org/fhir/subject": {
    "http://hl7.org/fhir/reference": "Patient/f001",
    "http://hl7.org/fhir/display": "P. van de Heuvel"
  },
  "http://hl7.org/fhir/valueQuantity": {
    "http://hl7.org/fhir/value": 6.2,
    "http://hl7.org/fhir/unit": "kPa",
    "http://hl7.org/fhir/system": "http://unitsofmeasure.org",
    "http://hl7.org/fhir/code": "kPa"
  }
}
```


http://tinyurl.com/uudge52

Quick Aside
JSON-LD formats

1. Expanded Document Form
2. Compacted Document Form
3. Flattened Document Form
4. Framed Document Form

Expanded Document Form

Expanded | Compacted | Flattened | Framed | N-Quads

```
[
  {
    "@id": "http://hl7.org/patient/1173418",
    "http://example.org/person/active": [
      {
        "@value": true
      }
    ],
    "http://example.org/person/birthdate": [
      {
        "@type": "http://www.w3.org/2001/XMLSchema#date",
        "@value": "2002-07-21"
      }
    ],
    "http://example.org/person/children": [
      {
        "@value": "Tom"
      },
      {
        "@value": "Dick"
      },
```

- **@context completely expanded (covered later)**
- **objects in expanded form: (“pred”: [{“@value”: …}]**
- **@graph - present if needed (option for expansion API)**

http://tinyurl.com/vtpe2ho

Compacted Document Form

Expanded | Compacted | Flattened | Framed | N-C

```
{
  "@id": "http://hl7.org/patient/1173418",
  "http://example.org/person/active": true,
  "http://example.org/person/birthdate": {
    "@type": "http://www.w3.org/2001/XMLSchema#date",
    "@value": "2002-07-21"
  },
  "http://example.org/person/children": [
    "Tom",
    "Dick",
    "Harry"
  ],
  "http://example.org/person/id": "1173418",
  "http://example.org/person/name": {
    "@id": "http://hl7.org/patient/1173418/name",
    "http://example.org/person/first": "Seymour",
    "http://example.org/person/last": "Snodgrass",
    "http://example.org/person/middle": "P"
  }
}
```

- **@context fully applied (covered later)**
- **objects in simplest form possible: ("pred": …)**

http://tinyurl.com/rbtqn5l

Flattened Document Form

```
{
  "@graph": [
    {
      "@id": "http://hl7.org/patient/1173418",
      "http://example.org/person/active": true,
      "http://example.org/person/birthdate": {
        "@type": "http://www.w3.org/2001/XMLSchema#date",
        "@value": "2002-07-21"
      },
      "http://example.org/person/children": [
        "Tom",
        "Dick",
        "Harry"
      ],
      "http://example.org/person/id": "1173418",
      "http://example.org/person/name": {
        "@id": "http://hl7.org/patient/1173418/name"
      }
    },
    {
      "@id": "http://hl7.org/patient/1173418/name",
      "http://example.org/person/first": "Seymour",
      "http://example.org/person/last": "Snodgrass",
      "http://example.org/person/middle": "P"
    }
  ]
}
```

- **compacted form with a separate @graph entry for *every* subject URI**

```
@prefix ex: <http://example.org/person/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://hl7.org/patient/1173418> ex:active true ;
    ex:birthdate "2002-07-21"^^xsd:date ;
    ex:children "Dick",
        "Harry",
        "Tom" ;
    ex:id "1173418" ;
    ex:name <http://hl7.org/patient/1173418/name> .

<http://hl7.org/patient/1173418/name> ex:first "Seymour" ;
    ex:last "Snodgrass" ;
    ex:middle "P" .
```

Turtle Equivalent

http://tinyurl.com/wdankok

Framed Document Form

```
@prefix ns1: <http://schema.org/> .
@prefix ns2: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://biguniversity.org/sam> a ns1:Person ;
    ns2:knows ( <http://biguniversity.org/john> <http://anotheruniversity.org/melissa> ) .

<http://biguniversity.org/john> a ns1:Person ;
    ns2:age 39 ;
    ns2:knows <http://anotheruniversity.org/melissa> .

<http://anotheruniversity.org/melissa> a ns1:Person ;
    ns1:member <http://anotheruniversity.org/melissa> .
```

How do we turn this into JSON?

Framed Document Form


```
{
  "@graph": [
    {
      "@id": "http://anotheruniversity.org/melissa",
      "@type": "http://schema.org/Person",
      "http://schema.org/member": {
        "@id": "http://anotheruniversity.org/melissa"
      }
    },
    {
      "@id": "http://biguniversity.org/john",
      "@type": "http://schema.org/Person",
      "http://xmlns.com/foaf/0.1/age": 39,
      "http://xmlns.com/foaf/0.1/knows": {
        "@id": "http://anotheruniversity.org/melissa",
        "@type": "http://schema.org/Person",
        "http://schema.org/member": {
          "@id": "http://anotheruniversity.org/melissa"
        }
      }
    },
    {
      "@id": "http://biguniversity.org/sam",
      "@type": "http://schema.org/Person",
      "http://xmlns.com/foaf/0.1/knows": {
        "@list": [
          {
            "@id": "http://biguniversity.org/john",
            "@type": "http://schema.org/Person",
            "http://xmlns.com/foaf/0.1/age": 39,
            "http://xmlns.com/foaf/0.1/knows": {
              "@id": "http://anotheruniversity.org/melissa"
            }
          },
          {
            "@id": "http://anotheruniversity.org/melissa",
            "@type": "http://schema.org/Person",
            "http://schema.org/member": {
              "@id": "http://anotheruniversity.org/melissa"
            }
          }
        ]
      }
    }
  ]
}
```


One Option - *always* expand inline

http://tinyurl.com/sr9wb5u

Framed Document Form

How do we turn this into JSON?

```
@prefix ex: <http://example.org/person/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://hl7.org/patient/1173418> ex:active true ;
    ex:birthdate "2002-07-21"^^xsd:date ;
    ex:children "Dick",
        "Harry",
        "Tom" ;
    ex:id "1173418" ;
    ex:name <http://hl7.org/patient/1173418/name> .

<http://hl7.org/patient/1173418/name> ex:first "Seymour" ;
    ex:last "Snodgrass" ;
    ex:middle "P" .
```

One Graph starting here?

```
{
  "@graph": [
    {
      "@id": "http://hl7.org/patient/1173418",
      "http://example.org/person/active": true,
      "http://example.org/person/birthdate": {
        "@type": "http://www.w3.org/2001/XMLSchema#date",
        "@value": "2002-07-21"
      },
      "http://example.org/person/children": [
        "Tom",
        "Dick",
        "Harry"
      ],
      "http://example.org/person/id": "1173418",
      "http://example.org/person/name": {
        "@id": "http://hl7.org/patient/1173418/name",
        "http://example.org/person/first": "Seymour",
        "http://example.org/person/last": "Snodgrass",
        "http://example.org/person/middle": "P"
      }
    }
  ]
}
```

http://tinyurl.com/r6xhkkk

Framed Document Form

How do we turn this into JSON?

```
@prefix ex: <http://example.org/person/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://hl7.org/patient/1173418> ex:active true ;
    ex:birthdate "2002-07-21"^^xsd:date ;
    ex:children "Dick",
        "Harry",
        "Tom" ;
    ex:id "1173418" ;
    ex:name <http://hl7.org/patient/1173418/name> .

<http://hl7.org/patient/1173418/name> ex:first "Seymour" ;
    ex:last "Snodgrass" ;
    ex:middle "P" .
```

Two Graphs w/ ref from first to second?

Framed Document Form

How do we turn this into JSON?

```
@prefix ex: <http://example.org/person/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://hl7.org/patient/1173418> ex:active true ;
    ex:birthdate "2002-07-21"^^xsd:date ;
    ex:children "Dick",
        "Harry",
        "Tom" ;
    ex:id "1173418" ;
    ex:name <http://hl7.org/patient/1173418/name> .

<http://hl7.org/patient/1173418/name> ex:first "Seymour" ;
    ex:last "Snodgrass" ;
    ex:middle "P" .
```

```
{
  "@graph": [
    {
      "@id": "http://hl7.org/patient/1173418",
      "http://example.org/person/active": true,
      "http://example.org/person/birthdate": {
        "@type": "http://www.w3.org/2001/XMLSchema#date",
        "@value": "2002-07-21"
      },
      "http://example.org/person/children": [
        "Tom",
        "Dick",
        "Harry"
      ],
      "http://example.org/person/id": "1173418",
      "http://example.org/person/name": {
        "@id": "http://hl7.org/patient/1173418/name",
        "http://example.org/person/first": "Seymour",
        "http://example.org/person/last": "Snodgrass",
        "http://example.org/person/middle": "P"
      }
    },
    {
      "@id": "http://hl7.org/patient/1173418/name",
      "http://example.org/person/first": "Seymour",
      "http://example.org/person/last": "Snodgrass",
      "http://example.org/person/middle": "P"
    }
  ]
}
```

Two Graphs, duplicating the information?

End Quick Aside

Expanded Format Value Objects

Allows the specification of:

- RDF data types (@type)

  “2002-07-22”^^"http://www.w3.org/2001/XMLSchema#date"

- RDF language tags (@language)

  “midwife”@en
  “sage-femme”@fr

- Text direction (@direction) - (“rtl”, “ltr”) (Not covered here)

- Content addressability (@index)

Language Tags

JSON-LD Input | Options | Document URL

```
"http://example.org/person/active": true,
"http://example.org/person/name": {
  "@id": "http://hl7.org/patient/1173418/name",
  "http://example.org/person/first": "Seymour",
  "http://example.org/person/middle": "P",
  "http://example.org/person/last": "Snodgrass"
},
"http://example.org/person/birthdate": {
  "@type": "http://www.w3.org/2001/XMLSchema#date",
  "@value": "2002-07-21"
},
"http://example.org/occupation": [
  {"@value": "midwife", "@language": "en"},
  {"@value": "sage-femme", "@language": "fr"}
],
"http://example.org/person/children": ["Tom", "Dick", "Harry"]
}
```

Language Tags

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA | Signed with Bit

```
<http://hl7.org/patient/1173418/name> <http://example.org/person/first> "Seymour" .
<http://hl7.org/patient/1173418/name> <http://example.org/person/last> "Snodgrass" .
<http://hl7.org/patient/1173418/name> <http://example.org/person/middle> "P" .
<http://hl7.org/patient/1173418> <http://example.org/occupation> "midwife"@en .
<http://hl7.org/patient/1173418> <http://example.org/occupation> "sage-femme"@fr .
<http://hl7.org/patient/1173418> <http://example.org/person/active> "true"^^<http://www.w3.org/2001/XMLSchema#boolean> .
<http://hl7.org/patient/1173418> <http://example.org/person/birthdate> "2002-07-21"^^<http://www.w3.org/2001/XMLSchema#date> .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Dick" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Harry" .
<http://hl7.org/patient/1173418> <http://example.org/person/children> "Tom" .
<http://hl7.org/patient/1173418> <http://example.org/person/id> "1173418" .
<http://hl7.org/patient/1173418> <http://example.org/person/name> <http://hl7.org/patient/1173418/name> .
```

http://tinyurl.com/sxashbg

JSON in JSON

The Issue: we need to include JSON as an RDF Literal:

_:b1 :representation ‘{“item”: 17.2, “count”: 103}’ .

Solution: “@type”: “@json”:

```
{
  "http://example.org/inv/entry": {
    "@type": "@json",
    "@value": {
      "item": 17.2,
      "count": 103
    }
  }
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA | Signed

```
_:b0 <http://example.org/inv/entry> "{\"count\":103,\"item\":17.2}"^^<http://www.w3.org/1999/02/22-rdf-syntax-ns#JSON> .
```

JSON to RDF Conversion Steps

1. Convert predicates to URI’s
2. Add subject URI’s where appropriate
3. Add data types, language tags and/or directions
4. Add list ordering

Lists in JSON-LD

JSON Lists are ordered:

“children”: [“First”, “Second”, “Third”]

JSON-LD Lists are not (!)

JSON-LD Input | Options

http://tinyurl.com/rymk6ao

```
{
  "http://example.org/person/children": [
    "Charlie",
    "Bravo",
    "Alpha"
  ]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Ta

```
_:b0 <http://example.org/person/children> "Alpha" .
_:b0 <http://example.org/person/children> "Bravo" .
_:b0 <http://example.org/person/children> "Charlie" .
```

These are equivalent

JSON-LD Input | Options

```
{
  "http://example.org/person/children": [
    "Charlie",
    "Bravo",
    "Alpha"
  ]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads

```
_:b0 <http://example.org/person/children> "Alpha" .
_:b0 <http://example.org/person/children> "Bravo" .
_:b0 <http://example.org/person/children> "Charlie" .
```

JSON-LD Input | Options

```
{
  "http://example.org/person/children": {"@set": [
    "Charlie",
    "Bravo",
    "Alpha"
  ]}
}
```

Expanded | Compacted | Flattened | Framed | N-Quads

```
_:b0 <http://example.org/person/children> "Alpha" .
_:b0 <http://example.org/person/children> "Bravo" .
_:b0 <http://example.org/person/children> "Charlie" .
```

Ordered Lists

JSON-LD Input | Options

```
{
  "http://example.org/person/children": {"@list": [
    "Charlie",
    "Bravo",
    "Alpha"
  ]}
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed

```
_:b0 <http://example.org/person/children> _:b1 .
_:b1 <http://www.w3.org/1999/02/22-rdf-syntax-ns#first> "Charlie" .
_:b1 <http://www.w3.org/1999/02/22-rdf-syntax-ns#rest> _:b2 .
_:b2 <http://www.w3.org/1999/02/22-rdf-syntax-ns#first> "Bravo" .
_:b2 <http://www.w3.org/1999/02/22-rdf-syntax-ns#rest> _:b3 .
_:b3 <http://www.w3.org/1999/02/22-rdf-syntax-ns#first> "Alpha" .
_:b3 <http://www.w3.org/1999/02/22-rdf-syntax-ns#rest> <http://www.w3.org/1999/02/22-rdf-syntax-ns#nil> .
```

Multiple Ordering Approaches

JSON-LD Input | Options

```
{
  "http://example.org/person/children": [
    {"@value": "Charlie"},
    {"@value": "Bravo"},
    {"@value": "Alpha"}
  ]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads

```
_:b0 <http://example.org/person/children> "Alpha" .
_:b0 <http://example.org/person/children> "Bravo" .
_:b0 <http://example.org/person/children> "Charlie" .
```

http://infolab.stanford.edu/~stefan/daml/order.html

http://smiy.sourceforge.net/olo/spec/orderedlistontology.html

Ordering Alternatives
Explicitly Insert Ordering Variable

JSON-LD Input | Options

```
{
  "http://example.org/person/children": [
    {
      "http://example.org/person/name": "Charlie",
      "http://hl7.org/fhir/index": 1
    },
    {
      "http://example.org/person/name": "Bravo",
      "http://hl7.org/fhir/index": 2
    },
    {
      "http://example.org/person/name": "Alpha",
      "http://hl7.org/fhir/index": 3
    }
  ]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table

```
_:b0 <http://example.org/person/children> _:b1 .
_:b0 <http://example.org/person/children> _:b2 .
_:b0 <http://example.org/person/children> _:b3 .
_:b1 <http://example.org/person/name> "Charlie" .
_:b1 <http://hl7.org/fhir/index> "1"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b2 <http://example.org/person/name> "Bravo" .
_:b2 <http://hl7.org/fhir/index> "2"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b3 <http://example.org/person/name> "Alpha" .
_:b3 <http://hl7.org/fhir/index> "3"^^<http://www.w3.org/2001/XMLSchema#integer> .
```

http://tinyurl.com/s7lm9tq

Ordering Alternatives
Ordering Variable with explicit identifiers

JSON-LD Input | Options

```
{
  "http://example.org/person/children": [
    {
      "@id": "http://someidentifier.org/charlie",
      "http://example.org/person/name": "Charlie",
      "http://hl7.org/fhir/index": 1
    },
    {
      "@id": "http://someidentifier.org/bravo",
      "http://example.org/person/name": "Bravo",
      "http://hl7.org/fhir/index": 2
    },
    {
      "@id": "http://someidentifier.org/alpha",
      "http://example.org/person/name": "Alpha",
      "http://hl7.org/fhir/index": 3
    }
  ]
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RSA

```
<http://someidentifier.org/alpha> <http://example.org/person/name> "Alpha" .
<http://someidentifier.org/alpha> <http://hl7.org/fhir/index> "3"^^<http://www.w3.org/2001/XMLSchema#integer> .
<http://someidentifier.org/bravo> <http://example.org/person/name> "Bravo" .
<http://someidentifier.org/bravo> <http://hl7.org/fhir/index> "2"^^<http://www.w3.org/2001/XMLSchema#integer> .
<http://someidentifier.org/charlie> <http://example.org/person/name> "Charlie" .
<http://someidentifier.org/charlie> <http://hl7.org/fhir/index> "1"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b0 <http://example.org/person/children> <http://someidentifier.org/alpha> .
_:b0 <http://example.org/person/children> <http://someidentifier.org/bravo> .
_:b0 <http://example.org/person/children> <http://someidentifier.org/charlie> .
```

Ordering Alternatives
Ordered List Ontology

http://tinyurl.com/qtuor5y

JSON-LD Input | Options

```
{
  "http://example.org/person/children": {
    "@type": "http://purl.org/ontology/olo/core#OrderedList",
    "http://purl.org/ontology/olo/core#length": 3,
    "http://purl.org/ontology/olo/core#slot": [
    {
      "http://purl.org/ontology/olo/core#index": 1,
      "http://purl.org/ontology/olo/core#item": "Charlie"
    },
    {
      "http://purl.org/ontology/olo/core#index": 2,
      "http://purl.org/ontology/olo/core#item": "Bravo"
    },
    {
      "http://purl.org/ontology/olo/core#index": 3,
      "http://purl.org/ontology/olo/core#item": "Alpha"
    }
      ]
  }
}
```

Expanded | Compacted | Flattened | Framed | N-Quads | Normalized | Table | Visualized | Signed with RS

```
_:b0 <http://example.org/person/children> _:b1 .
_:b1 <http://purl.org/ontology/olo/core#length> "3"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b1 <http://purl.org/ontology/olo/core#slot> _:b2 .
_:b1 <http://purl.org/ontology/olo/core#slot> _:b3 .
_:b1 <http://purl.org/ontology/olo/core#slot> _:b4 .
_:b1 <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/ontology/olo/core#OrderedList> .
_:b2 <http://purl.org/ontology/olo/core#index> "1"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b2 <http://purl.org/ontology/olo/core#item> "Charlie" .
_:b3 <http://purl.org/ontology/olo/core#index> "2"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b3 <http://purl.org/ontology/olo/core#item> "Bravo" .
_:b4 <http://purl.org/ontology/olo/core#index> "3"^^<http://www.w3.org/2001/XMLSchema#integer> .
_:b4 <http://purl.org/ontology/olo/core#item> "Alpha" .
```

JSON-LD Context

Predicates to URIs
JSON to

```
{ “@context”: {
      “@vocab”: “http://example.org/person/"
  },
  “id”: “1173418”,
  “active” : true,
  “name” : {
      “first” : “Seymour”,
      “middle” : “P”,
      “last”: “Snodgrass”
  },
  “birthdate” : “2002-07-21”,
  “children” : [“Tom”, “Dick”, “Harry”]
}
```

JSON and RDF

An arbitrary JSON structure (link):

```
{ “http://example.org/person/id”: “1173418”,
  “http://example.org/person/active” : true,
  “http://xmlns.com/foaf/0.1/name” : {
      “http://xmlns.com/foaf/0.1/givenName” : “Seymour”,
      “http://example.org/person/middle” : “P”,
      “http://xmlns.com/foaf/0.1/familyName”: “Snodgrass”
  },
  “http://example.org/person/birthdate” : “2002-07-21”,
  “http://example.org/person/children” : [“Tom”, “Dick”,
“Harry”]
}
```

@context

An arbitrary JSON structure (link):

```
{ "@context": {
      "@vocab": "http://example.org/person/"
  },
  "id": "1173418",
  "active" : true,
  "name" : {
      "first" : "Seymour",
      "middle" : "P",
      "last": "Snodgrass"
   },
   "age" : 42,
   "children" : ["Tom", "Dick", "Harry"]
}
```

Decision 1: URL Context

- http://tinyurl.com/yenkhzrm